DETECTION OF RAIN IN ACOUSTIC RECORDINGS OF THE ENVIRONMENT USING MACHINE LEARNING TECHNIQUES

A THESIS SUBMITTED TO THE FACULTY OF SCIENCE AND ENGINEERING OF QUENSLAND UNIVERSITY OF TECHNOLOGY

MASTER OF INFORMATION TECHNOLOGY (IT60)

Ms. Meriem Ferroudj

School of Electrical Engineering and Computer Science

Science and Engineering Faculty

March 2015

Statement of Original Authorship

The work contained in this thesis has not been previously submitted to meet requirements for an award at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made.

QUT Verified Signature

Signature: Meriem Ferroudj

Date: 26 March 2015

Acknowledgements

I would like to take this opportunity to acknowledge all those who helped me during this thesis work. I would like to thank my supervisors: Michael Towsey, Jinglan Zhang, Paul Roe, and Jasmine Banks for introducing me to the world of Machine Learning and its application to real world problems, their valuable suggestions, their directions, friendship and understanding during difficult times have been inspirational.

My sincere thanks go to the members of the eco-acoustic research group at QUT: Anthony, Mark, Liang, Jason, Xueyan, Phil and Jie for their friendship. Special thanks go to Yvonne Phillips who assisted me with proofreading.

Big thanks go to my husband and my young boys for their love, patience and support throughout my whole degree. Special thanks go to my family in particular my mother. They always wanted me to success in my studies.

My Master degree was supported by the Australian government's Research Training Scheme (RTS). In this regards, I express my appreciation for financial support from the RTS.

Abstract

Environmental monitoring has become increasingly important due to the significant impact of human activities and climate change on biodiversity. Environmental sound sources such as rain and insect vocalizations are a rich and underexploited source of information in environmental audio recordings. Rain is a frequent component of environmental recordings and in some research areas is avoided or removed depending on the application. This thesis is concerned with the detection of rain within acoustic sensor recordings.

Detection of rain will advance the techniques for biodiversity analysis and the proposed method will help and save time for ecologists when they are browsing/navigating long audio recordings, in order to find a particular animal call.

We approached rain detection in acoustic recordings as a classification task using multiple machine learning techniques. We investigated the novel application of a set of features (known as indices) for classifying the content of acoustic recordings: acoustic entropy, the acoustic complexity index, spectral cover, and background noise. In order to improve the performance of the rain classification system we automatically classified segments of environmental recordings into the classes of *heavy* r*ain* or *non-heavy rain*. A Decision Tree classifier is experimentally compared with other classifiers. The experimental results show that our system is effective in classifying segments of environmental audio recordings with an accuracy of 93% for the binary classification of *heavy rain*/*non-heavy rain* (Experiment1). It demonstrates that the features used are promising for classifying acoustic recordings of the environment. Other experiments were conducted; in the multi-class problem (Experiment 2), the confusion matrix showed that the feature set used is capable in distinguishing between multiple classes. In the Experiment 3, different combination of the classification algorithms were tested and it is found that combining different algorithms give a better accuracy rate than using a single classification algorithm. We also conducted another experiment (Experiment 4), which consisted of the prediction of rain in a long recording (24h long).

To test whether the identified feature set for rain classification is useful or not for rain estimation, we applied it to predict rain in a long recording (24h long), then mapped the prediction outputs with weather data for that particular day using different regression techniques. A promising result was achieved with the M5P model at high correlation and low prediction errors.

Keywords

Environmental sound classification

Acoustic event classification

Feature extraction

Audio classification

Machine learning

Data mining

Prediction techniques

Publications

Meriem Ferroudj, Anthony Truskinger, Michael Towsey, Liang Zhang, Jinglan Zhang and Paul Roe. Detection of rain in acoustic recordings of the environment. Paper presented at *the 13th Pacific Rim International Conference on Artificial Intelligence (*PRICAI 2014*)*, 1-5 Dec 2014, Gold Coast, Australia, Lecture Notes in Computer Science Volume 8862, 2014, pp 104-116.

Table of Contents

List of Figures

List of Tables

Abbreviations

Acronym	Meaning
ACI	Acoustic Complexity Index
ANN	Artificial Neural Network
BgN	Background Noise
BP	Band Periodicity
DT	Decision Tree
H	Acoustic Entropy
Hf	Spectral Entropy
HMM	Hidden Markov Model
Ht	Temporal Entropy
kNN	k-Nearest Neighbour
IBk	Instance based learner with fixed neighbourhood k
LPC	Linear Predictive Coding
MFCC	Melfrequency Cepstral Coefficient
MFC	Mel- Frequency Coefficient
MP	Matching Pursuit Algorithm
NFR	Noise Frame Ratio
SF	Spectrum Flux
SNR	Signal to Noise Ratio
SC	Spectral Cover
STFT	Short Time Fourier transform

SVM	Support Vector Machines
WT	Wavelet Transform
ZCR	Zero Crossing Rate

Data Management

The data used in this research have been stored in my own and my supervisor's hard-disk.

I also have one copy stored in my working laptop. All the data related to this research project will be kept for 5 years according to QUT's data management policies.

Chapter 1: **Introduction**

This Chapter outlines the background and motivation (Section 1.1), aims and objectives (Section 1.2) of the research, and its approaches (Section 1.3). Section 1.4 describes the research scope. Section 1.5 describes the contributions and significance. Finally, Section 1.6 includes an outline of the remaining chapters of the thesis.

1.1 BACKGROUND AND MOTIVATION

Environmental sounds are a rich and underexploited source of information in environmental monitoring. They are highly non-stationary and contain much background noise. Hence, it is hard to describe environmental sounds using common audio features. Defining suitable features for environmental sounds is an important problem in an automatic acoustic classification system.

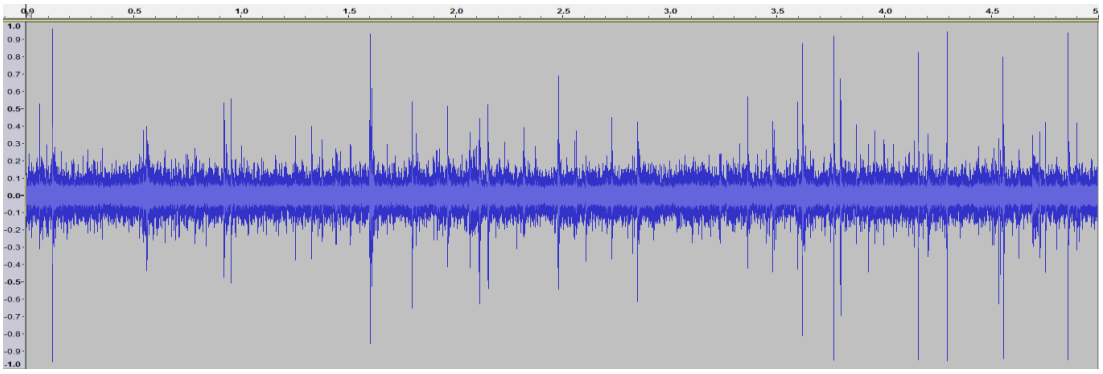
Much of the noise in environmental recordings is of physical origin such as wind, rain, rustling of leaves, etc.; biological origin such as cicadas, bird vocalizations and other animals; and human generated sound such as highway traffic or airplane engine noise. In this work, we define noise as signals with constant acoustic energy which remains constant throughout the duration of the recording. Thus it is possible that the same acoustic source may contribute to both "noise" and specific events "signal". So, assuming that we are interested in birds or cicadas recognition, then rain and wind might be regarded as noise. Finally there is another sense of noise which can be defined as any acoustic event that is not of interest.

In some applications, background noise such as rain is not of interest and often discarded. In our study, background noise, in particular rain, represents our event (signal of interest). For ecologists, rain presents background noise when they are estimating species richness by sampling very long acoustic recordings. Avoiding periods having much background noise will improve the efficiency of audio sampling. For example, when ecologists analyse bird calls in audio recordings, rain makes it harder to annotate bird vocalisations. Therefore, whether birds are calling or not during rain, ecologists may not want to listen to that audio. Masking this

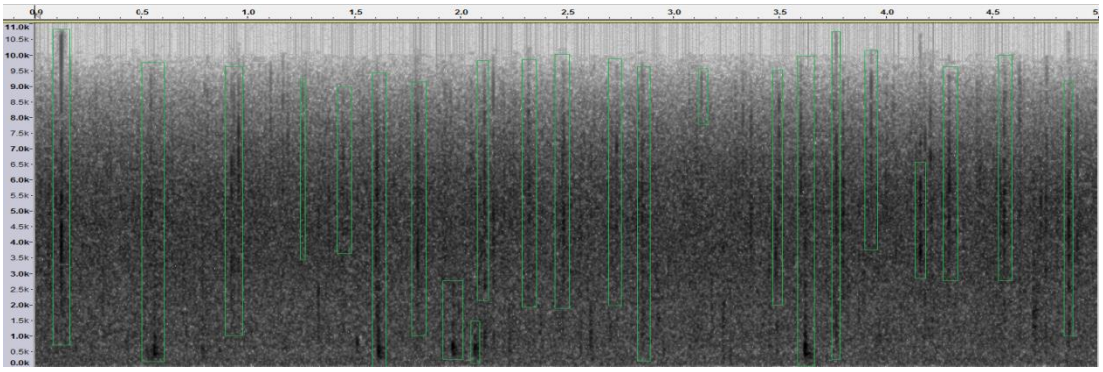
background noise will increase the efficiency and effectiveness of bioacoustics data analysis. This research aims to mark this background noise rather than discarding it as it may be useful in other cases such as frog call analysis.

Figure 1.1 shows the structure of different environmental sounds, such as rain, thunder and bird calls. Figure 1.1(a) is a representation of heavy rain in time domain.

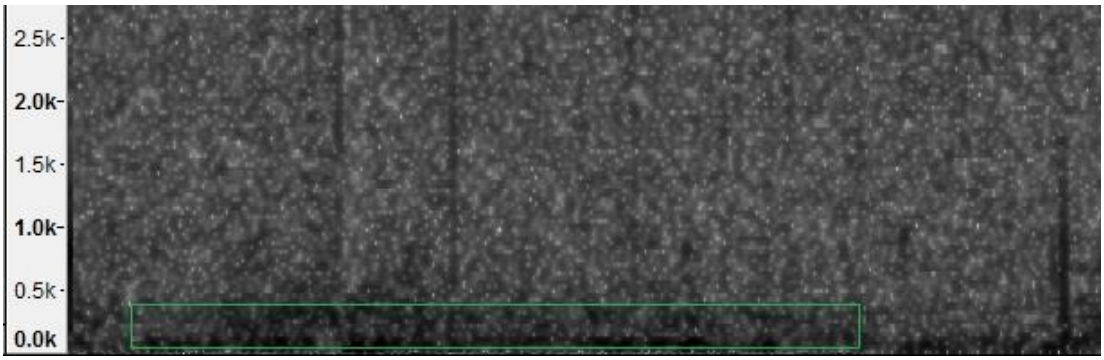
The Figure 1.1(b), (c), and (d) are images/spectrograms in frequency domain; where x-axis represents time, the y-axis represents frequency and the grey scale represents acoustic intensity. The green boxes in this Figure 1.1(b) highlight the rain drops when they hit the surfaces near the microphone. It shows that rain presents vertical lines in a spectrogram and it often occupies the whole frequency band.



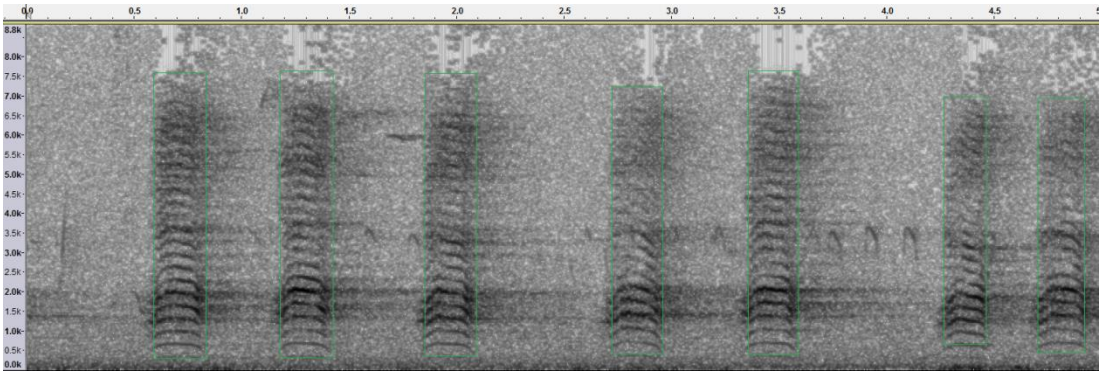
(a) Waveform of heavy rain in time domain



(b) Spectrogram of heavy Rain (vertical lines) in frequency domain



(c) Spectrogram of thunder (the energy is concentrated in the low frequency band)



(d) Spectrogram of a crow call

Figure 1.1 Environmental sounds representation

Our eco-acoustics research group has researched and deployed different types of acoustic sensors (Mason et al., 2008) and collected a large amount of acoustic data (over 24Tb/5 years). Multiple automatic species recognisers have been developed for the ground parrot, male koala, Asian house gecko, whipbird, and other animals. Bioacoustic analysis has become an important field of study when monitoring environment changes. Instead of sending ecologists to the field to record sounds of the environment or making surveys, different types of sensors and audio recorders could instead be deployed in the field environment to help ecologists record any sound. This method has multiple advantages over standard surveys:

- Saves time and effort,
- Provides continuous and persistent recordings,
- Scales over large area and long period.

However, the acoustic data collected is not free from the background noise.

Wind and rain are frequently found in environmental recordings and are generally considered noise because they adversely affect the performance of automatic species recognisers, and mask useful information. Marking this background noise will increase the efficiency and effectiveness of bioacoustics data analysis. For example, if ecologists were interested in a particular bird species, our method will help them by telling them not to look into this part of audio because it has rain (and often when it rains animals and in particular birds are less active), but look into that part of audio (without rain) which is more probable to find birds or whatever animal they are interested in.

1.2 AIMS AND OBJECTIVES

The goal of the proposed research is to automatically classify the content of audio recordings into different classes. The purpose behind this classification is the detection of *heavy rain* in the acoustic recordings of the environment. Environmental sounds comprise all types of sound including speech, music, animal sounds and background noise, etc. There have been many studies on audio classification and segmentation using different machine learning techniques (Karbasi, Ahadi, & Bahmanian, 2011; Ma, Milner, & Smith, 2006; Vavrek, Cizmar, & Juhar, 2012).

In this work we investigate a new set of features previously used in environment monitoring to classify the content of acoustic recordings.

The objectives of the study are:

(1) To explore a set of features originally used in environment monitoring but not evaluated on the classification of the content of acoustic recordings;

(2) To classify environmental sounds into multiple classes including *heavy rain*, *cicada chorus*, *animal sounds* (*bird calls*, *frog-calls*, *koala bellow)*, and others (*light rain* and *night silence*/*low activity);*

(3) To investigate different machine learning techniques on the detection of *rain* in the acoustic recordings.

1.3 RESEARCH PROBLEM AND QUESTIONS

The eco-acoustics research group at Queensland University of Technology has researched and deployed different types of acoustic sensors and collected a large

amount of acoustic data (over 24Tb/5 years) in order to monitor the environment's health. Multiple automatic species recognisers have been developed for the ground parrot, male koala, Asian house gecko, whipbird, and other animals. However, the acoustic data collected is not free from the background noise. Wind and rain are frequently found in environmental acoustics and are generally considered as noise. They adversely affect the performance of automatic species recognisers, and mask useful information. The present research focuses on the detection and prediction of rain in environmental raw data.

We approach rain detection in acoustic recordings as a classification task, where the goal is to avoid listening to audio content that contains rain since birds are less likely to vocalise in rainy condition.

There are three key components in any classification system shown in Figure 1.2:

1) Dataset preparation;

2) Feature extraction; and

3) Sound classification.

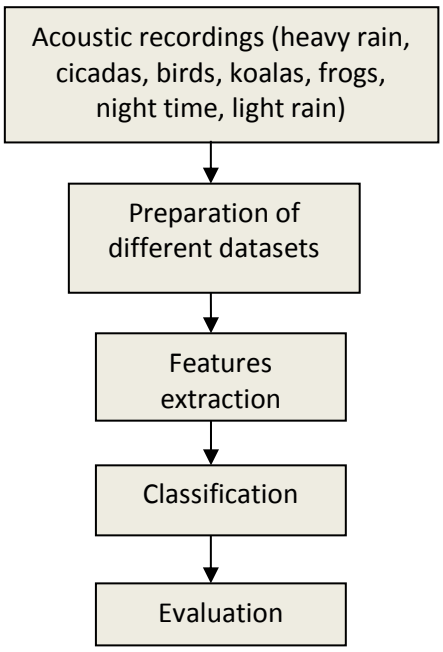


Figure 1.2 Flow chart for the classification process

The research questions for each component are outlined below:

1) Dataset preparation

- How to choose the data of interest from the existing data set?

2) Features extraction

- What features are more suitable for representing environmental sounds and what information do the features carry?
- How to extract these features from the acoustic signals?

3) Environmental sounds classification and regression

- How can environmental sounds be automatically classified using one or a set of classifiers with high accuracy?
- What algorithms yield best results for environmental sound classification?
- How to estimate rainfall in audio recordings?

1.4 RESEARCH SCOPE

This research mainly focuses on the classification of several categories of environmental field data: *heavy rain*, *cicada chorus*, *animal sounds* (*bird calls*, *koala bellow*, *frog calls*), and others (*night time* and *light rain*). Firstly, we have classified the acoustic recordings into heavy-*rain/non-heavy rain* (binary classification) using a dataset recorded from the field and a certain set of features. Secondly, we have performed a multi-class classification using the same dataset and same features as in the binary classification. In addition to the classification tasks, we have extracted the same set of features from a long audio recording, conducted regression analysis and compared with the corresponding weather data (ground truth) recorded by a weather station. The result shows that these features are also good for estimating the degree of rainfall.

1.5 CONTRIBUTIONS AND SIGNIFICANCE

This research focuses on the classification of environmental sounds, such as heavy rain, cicada chorus, bird calls, koala bellow, frog calls, night time and light

rain, using different machine learning algorithms. We used raw audio data automatically recorded by sensors from the field. We explored a novel combination of a set of features, e.g. temporal and spectral entropies, which have been reported useful for the detection of bioacoustic activity and investigated them in the new application of environmental sound classification. We have also investigated the effectiveness of these features in the novel application of rainfall estimation using acoustic recordings.

In particular, the contributions of this research include:

- The effectiveness of a novel combination of different features, namely: acoustic complexity index, acoustic entropy (both spectral and temporal); background noise, and spectral cover.
- The comparison of multiple classifiers in the new application: binary rain classification.
- The further investigation of novel application of rainfall estimation using acoustic recordings with the same feature set used for binary rain classification. We have tested the approach on a 24h-long recording to estimate rainfall and compared the results with the corresponding weather data for the same day and from the same location and the results are very promising.
- The manual and careful preparation of a dataset consisting of 998 different types of audio recordings.

The significance of this research lies in:

- This research result will significantly improve the efficiency and accuracy of automatic species recognisers. Audio recordings include multiple types of acoustic events such as various sounds produced by birds, insects, frogs, human, rain and airplane etc. It is very important to detect automatically these noise-like sounds or events (e.g., rain and wind) and mark them because these sounds can mask content of interest, for example if we are interested in bird calls, then rain becomes an obstacle.
- When combined with other acoustic event recognisers, the research result can also help with correlation analysis between animal behaviour and

weather information. For example, most birds are less active during rain while frogs are more active during or after rain.

1.6 THESIS OUTLINE

This Thesis consists of the following chapters:

Chapter 1 introduces the background and motivation of this thesis.

Chapter 2 reviews the literature related to environmental sounds, audio classification, and the algorithms used for audio classification.

Chapter 3 illustrates the structure of the designed classification system.

Chapter 4 describes a series of experiments to evaluate the classification system.

Chapter 5 conclusion and future work of this research.

Chapter 2: **Literature Review**

In this part we review the most common audio features used in audio classification and explore the related classification techniques and algorithms.

2.1 CONCEPTS

2.1.1 List of definitions

There are several important definitions in the proposed research.

***Acoustic Event*:** is a localised part/region of high intensity in a spectrogram.

Acoustic index: is a statistic that summarizes some aspect of the structure and distribution of acoustic energy and information in a recording.

***Background Noise index*:** Estimated from the wave envelope using the method of Lamel et al. The value is given in decibels (Lamel, Rabiner, Rosenberg, & Wilpon, 1981).

Cross-validation (machine learning): it is a technique for estimating the performance of a predictive model.

***K-fold-Cross-validation*:** the data set is randomly partitioned into k folds (k1, k2, ..., k10) without overlap. Then at the first run, take k1 to k9 as training set and develop a model. Test that model on k10 to get its performance. Next takes k1 to k8 and k10 as training set. Train a model from them and test it on k9. In this way, use all the folds where each fold is used as test set at most one time. The performance from the folds then can be averaged (or combined) to produce a single estimation.

2.1.2 Introduction to acoustics

Acoustics is the interdisciplinary science that deals with the study of all mechanical waves in gases, liquids, and solids including vibration, sound, ultrasound and infrasound. The application of acoustics can be seen in almost all aspects of modern society with the most obvious being the audio and noise control industries.

Because hearing and speech are two of the most important senses of human beings, it is no surprise that the science of acoustic spreads across so many facets of our society – music, architecture, industrial production, warfare and more. Likewise,

animal species such as birds and frogs use sound and hearing as a key element of mating rituals or marking territories.

The following table shows the divisions of acoustics established in the PACS (Physics and Astronomy Classification Scheme) classification system.

Table 2.1 The divisions of acoustics in Physics and Astronomy Classification Scheme (PACS).

Physical acoustics	**Biological acoustics**	**Acoustical engineering**
➢ Aeroacoustics ➢ General linear acoustics ➢ Nonlinear acoustics ➢ Structural acoustics and vibration ➢ Underwater sound	➢ Bioacoustics ➢ Musical acoustics ➢ Physiological acoustics ➢ Psychoacoustics ➢ Speech communication (production; perception; processing and communication systems)	➢ Acoustic measurements and instrumentation ➢ Acoustic signal processing ➢ Architectural acoustics ➢ Environmental acoustics ➢ Transduction ➢ Ultrasonics ➢ Room acoustics

2.1.2.1 Audio frequency

An *audio frequency* or *audible frequency* is characterized as a periodic vibration whose frequency is audible to the average human. It is the property of sound that most determines pitch and is measured in hertz (Hz).

The generally standard range of audible frequencies is 20 to 20,000 Hz although the range of frequencies individuals hear is greatly influenced by environmental factors and by age. Frequencies below 20 Hz are generally felt rather than heard, assuming the amplitude of the vibration is great enough. Frequencies above 20,000 Hz can sometimes be sensed by young people. High frequencies are the first to be affected by hearing loss due to age and/or prolonged exposure to very loud noises.

Frequencies above and below the audible range are called "ultrasonic" and "infrasonic", respectively.

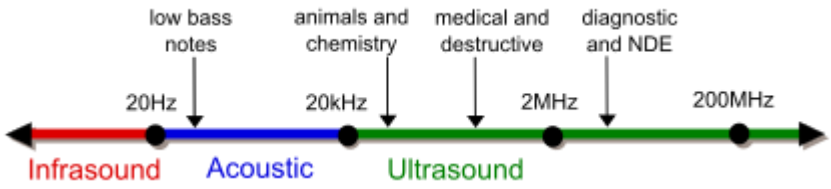


Figure 2.1 Approximate frequency ranges corresponding to ultrasound

The study of acoustics involves the generation, propagation and reception of mechanical waves and vibrations.

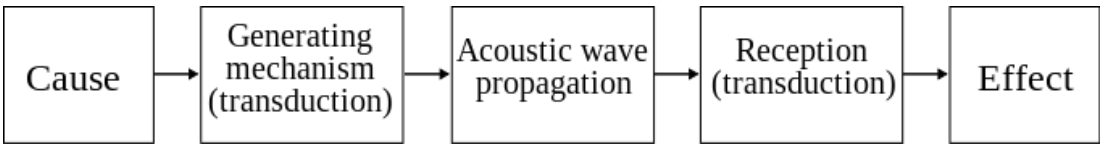


Figure 2.2 Diagram of an acoustical event

The steps shown in the above diagram can be found in any acoustical event or process. There are many kinds of causes, both natural and volitional. There are also many kinds of transduction proves that convert energy from some other form into sonic energy, producing a sound wave. The wave carries energy throughout the propagating medium. Eventually the energy is transduced again to other forms, in ways that again may be natural or volitionally contrived. The final effect may be purely physical or it may reach far into the biological or volitional domains. These five steps are found equally well in seismology, sonar or a band playing in a rock concert.

2.1.3 Environmental audio data

The study site of this research is the QUT Samford Ecological Research Facility (SERF) in the Samford Valley, at 25 minute drive northwest of QUT Gardens Point Campus in Brisbane, Queensland. The dominant vegetation is open-forest to woodland comprised primarily of *Eucalyptus tereticornis*, *E. crebra* (and sometimes *Esiderophloia*) and *Melaleuca quinquenervia* in moist drainage. There are also small areas of gallery rainforest (with *Waterhousea floribunda* predominantly fringing the Samford Creek to the west of the property) and areas of open pasture along the southern boundary.

Regarding the present project, acoustic sensor surveys were conducted at four locations over five days. Sites were located in the eastern corner within open woodland, the northern corner within closed forest along Samford Creek, in the western corner within Melaleuca woodland, and in the southern corner where open forest borders cleared pasture (Figure 2.3). Each site was 100m x 200m and marked with flagging tape. In addition, a weather station was located in the northern section of the property.

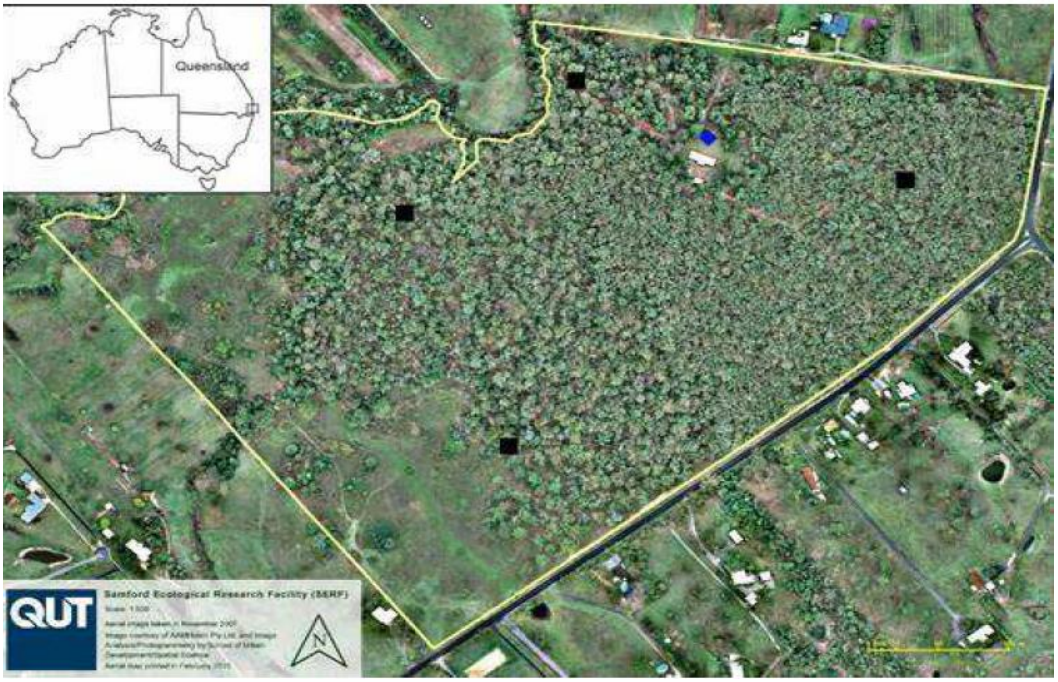


Figure 2.3 Samford Ecological Research Facility (SERF) with survey site positions marked with black squares and weather station position marked with blue diamond

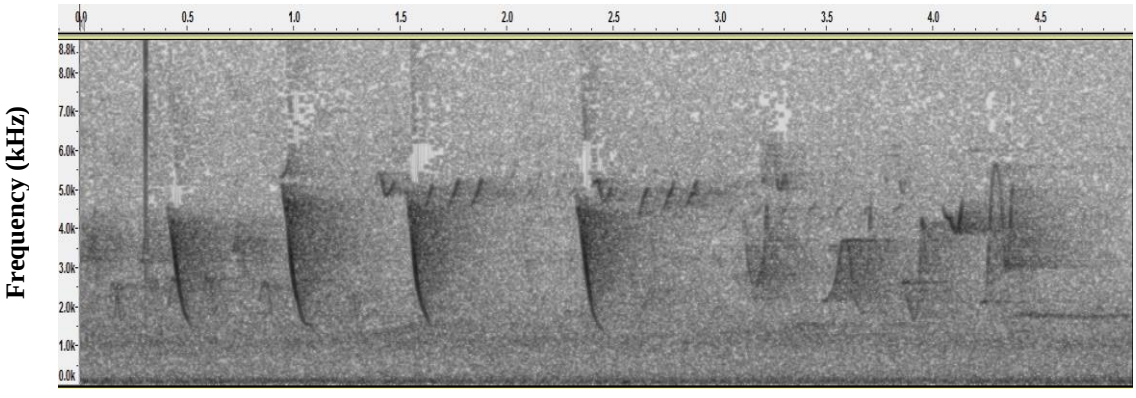
The sensors deployed by the QUT eco-acoustic research group have recorded a large amount of acoustic data at four outdoor sites (see details about these locations (Michael. Towsey & Planitz, 2010)) in Queensland, Australia. Contrasted with audio data collected in the laboratories or quiet environments, environmental audio data also called real-world data is collected in the field, which often records a large number of vocalizations from various sound sources. These sources can be from birds, cicadas, rain, wind, thunder, airplane, and human activities (speech and traffic). The unwanted sounds, in particular rain are viewed as background noise in many applications. However, rain is considered as sound of interest in this research as rain identification can have many applications.

2.1.4 Sound analysis

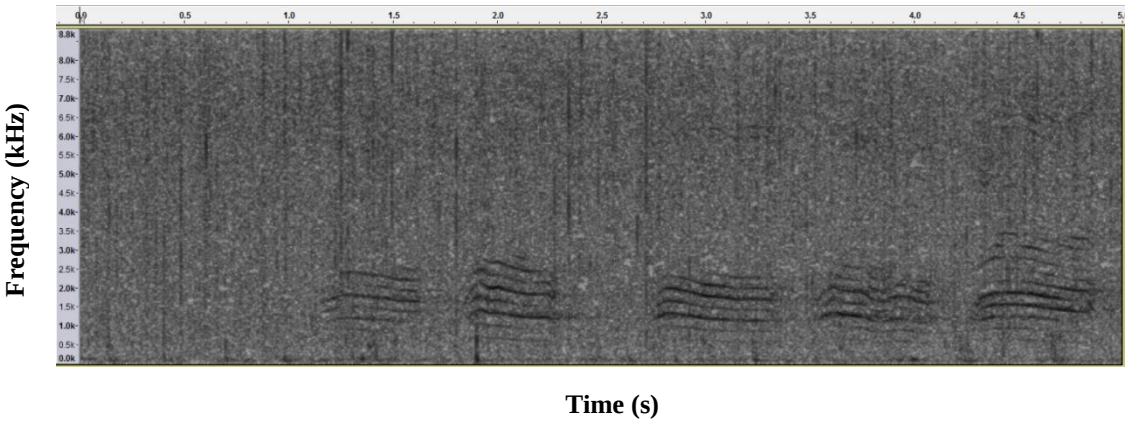
Sounds are time-varying signals in the real world and all of their meaning is related to such time variability. Sound analysis techniques are developed to grasp at least some of the distinguished time-varying features, in order to ease the tasks of understanding comparison, modification and resynthesis for signals (Rocchesso, 2003). The application areas of sound analysis have covered many aspects of acoustic environments: speech processing, video processing, bioacoustics analysis etc.

The most important sound analysis techniques are Short-time Fourier Transform (STFT) and Linear Predictive Coding (LPC). STFT is performed on slices of the time-domain signal and the function of STFT is to transform signals from time-domain to frequency-domain (Griffin & Lim, 1984; Saunders, 1996) . LPC analysis is an efficient and effective mean to achieve synthetic speech and speech signal communication (Brigham & Morrow, 1967).

One of the most useful visual representations of audio signals is the spectrogram. A spectrogram is a grey-scale or colour rendition of the magnitude of the STFT, on a 2D plane where x-axis represents time, y-axis represents frequency and the intensity or colour indicates the amplitude of a particular frequency at a particular time. An example of a spectrogram derived from a field recording can be seen in Figure 2.4 (a) and Figure 2.4 (b).



(a) Spectrogram of whipbird's call



(b) Spectrogram of crow's call

Figure 2.4 Spectrograms of field recordings

2.1.5 Acoustic event and background noise

Acoustic event means timestamps in an audio stream (Zhuang et al., 2010). As we can see from Figure 2.4, there can be lots of events in one spectrogram. Some events are calls of interest while some are not. More specifically, calls of interest are called acoustic events in environmental acoustic studies. Whereas, those events that are out of interest are all called background noise. Consequently, the definition of background noise is ambiguous (Planitz & Towsey, 2010). In our study we assume only continuous background noise through a time interval, such as might come from heavy rain sounds, the rustling of leaves or distant traffic.

Background noises in some application are signals of interest in other applications. For example, in one audio recording, the aim of bird call classification is to find calls belonging to a specific species, which will consider other species calls as background noise, like cicadas. Therefore, the definition of noise is dependent on the application area. Generally, background noise is divided into natural and artificial noise. Natural noise might come from the rustling of leaves, wind and rain. Artificial noise comes from human activities: speaking, distant traffic and moving a chair, etc.

2.1.6 Audio features

Acoustic features can be classified into two classes: statistical and non-statistical feature (Cheng, Sun, & Ji, 2010). Statistical features include the mean fundamental frequency, maximum fundamental frequency, minimum fundamental frequency, zero-crossing rate, short-time energy and signal bandwidth. Non-

statistical features include linear prediction coefficients, mel-frequency cepstral coefficients, spectral flux, band energy ratio, etc.

2.1.7 Features in time domain

In the time domain, the short-time energy and average zero-crossing rate in the waveform are measured to classify speech and music events because speech and music have different spectral distribution and temporal changing patterns (Saunders, 1996). However, according to Ye et al.,(2006) calculating the zero crossing rates and energy is not an effective way to detect voice signal in the recordings when the signal to noise ratio (SNR) is quite low. So they proposed a method which combines the geometrically adaptive energy threshold and Least-Square periodicity estimator to analyse the data with a low SNR. Since limited statistical features can be derived directly from the waveform (Wolff, 2008), many researchers turn to spectrograms for obtaining more frequency related features (non-statistical).

- *Zero-crossing rate (ZCR):* has proved to be useful in characterizing different audio signals. It is used in many speech/music classifications algorithms. Zero crossing occurs when the amplitude of successive samples changes from positive to negative or vice versa. The ZCR is the average number of times the signal changes its sign within the short-time window (Srinivasan, Petkovic, & Ponceleon, 1999).
- *The short time energy (STE):* is defined as the total energy in a signal frame (Pohjalainen, 2007).

2.1.8 Features in frequency domain

STFT is usually used for generating spectrogram, and multiple frequency-related features are derived from spectrograms.

MFCCs are one of the most widely used features for audio classification. The idea is to first compute Mel-frequency coefficients (MFCs) which are similar to the magnitude spectrum (the magnitude spectrum represents the intensity of the sound during a frame of signal at different frequencies), but in units of Mels rather than Hertz (Briggs, Raich, & Fern, 2009). Therefore, MFCC features are modelled based on the shape of the overall spectrum, making it more favourable for modelling single sound sources (speech). However, environmental sounds typically contain a large

variety of sounds, including conditions that are characterized by narrow spectral peaks, such as chirping of insects, rain drops, which MFCCs are unable to encode effectively (Chu, Narayanan, & Jay Kuo, 2008) . The filter-banks for MFCC are based on the human auditory system and have been shown to work particularly well for structured sounds, like speech and music, but their performance degrades in the presence of noise (Chu, Narayanan, & Kuo, 2009).

Another commonly used feature is linear prediction cepstral coefficients (LPCCs) (Markel & Gray, 1982). The basic idea behind linear prediction is that the current sample can be predicted, or approximated, as a linear combination of previous samples, which would provide a more robust feature against sudden changes.

In the following paragraph we describe some of other popular spectral features:

- *Band Energy Ratio:* the ratio of the energy in a specific frequency-band to the total energy (Eronen et al., 2006).
- *Spectral flux (SF):* used to measure a spectral amplitude difference between two successive frames (Mitrović, Zeppelzauer, & Breiteneder, 2010).
- *Spectral roll-off:* quantifies the frequency value at which the accumulative value of the frequency response magnitude reaches a certain percentage of the total magnitude. A commonly used threshold is 85% (Pfeiffer & Vincent, 2001).

There are also features that take into account more aspects of human auditory perception and are called also perceptual features, such as pitch, loudness and brightness.

Many researchers have made efforts to improve the accuracy of the classification and recognition of environmental sounds using different type of features, a variety of classifiers and small feature set.

In the thesis of Chu et al (2009), they used a new set of time-frequency features and consider the task of recognising environmental sounds for the understanding of a scene or context surrounding an audio sensor. They proposed a novel feature extraction method that uses Matching Pursuit algorithm (MP) to select a small set of

time-frequency features to analyse environment sounds. They adopted a Gaussian mixture model (GMM) classifier for classifying 14 types of environmental sounds. In their results, they have found that using MFCC and MP features separately gives a poor accuracy rate. By combining MFCC and MP features, the average accuracy rate obtained is 83.9% in discriminating fourteen classes.

Li (2010) stated that the matching pursuit algorithm is a good technique for feature extraction, which can clearly describe the environmental sounds. They have also demonstrated that the combination of the features MP and MFCC achieves a high accuracy rate. They use the support vector machine as a classifier for the environmental sound classification system. The accuracy rate of 92% was achieved.

Mitrovic et al. (2009) have employed principal component analysis for the composition of an optimal feature set for environmental sounds and conducted retrieval experiments to evaluate the quality of the feature combinations. The retrieval results show that statistical data analysis gives useful hints for feature selection in environmental sound recognition.

Barkana et al. (2011) explored the classification of a limited number of environmental sounds (engine, restaurant, and rain). They proposed a new feature extraction based on the fundamental frequency (pitch) of the sound. They used two different classifiers, SVM and k-means clustering to classify the different classes. The classifiers used in their research achieved recognition rates of 95.4% and 92.8%, respectively.

Generally, *pitch* is a perceptual feature of sound and its perception plays an important part in human hearing and understanding of different sounds. In an acoustic environment, human listeners are able to recognise the pitch of several real-time sounds and make efficient use of the pitch to acoustically separate a sound in a mixture (Bregman, 1994). However, noise-like non speech audio signals such as street noise, rain, a scream or a gunshot do not have a constant pitch but a range of values.

Uzkent et.al (2012) introduced a new time-frequency feature set combined with a feature extraction method based on the pitch range (PR) of non-speech sounds and the autocorrelation function, to classify non- speech environmental sounds such as:

gunshot, glass breaking, scream, dog barking, rain, engine, and restaurant noise. They have compared the accuracies of the proposed features to MFCCs by using support vector machines and radial basis function neural networks classifiers. The new feature set provided a high accuracy rate when it's used by itself and significantly improved when combined with MFCCs. They made a conclusion that both features methods are complementary.

Most previous works use a combination of some features or even a larger feature set to characterise the audio signals. However adding more features is not always helpful. As the feature dimension increases, data points become sparser and there are potentially irrelevant features that could negatively impact the classification results. The work of Chu et al.(2006) and Colonna et al. (2012) have shown that using high feature set dimension for classification does not always produce good performance for audio classification problems. They used a simple feature selection algorithm to obtain a smaller feature set to reduce the computational cost and running time and achieve an acceptable classification rate.

2.1.9 Other features

Some of recent indices have been used for biodiversity, but not evaluated on rain classification in particular. The following indices are used as our main features and briefly described below.

Acoustic entropy (H) is a measure of the dispersal of acoustic energy within a recording, either through time or frequency bands (Sueur, Pavoine, Hamerlynck, & Duvail, 2008). Sueur et al., (2008) acknowledge the difficulty of building individual species recognizers and therefore turn to indirect measures of biodiversity, making the simple assumption that the number of vocalizing species positively correlates with the acoustic heterogeneity of audio data within a locality. They conclude that acoustic entropy does correlate with acoustic heterogeneity. Their conclusion relies on artificially constructed recordings derived by concatenating a variety of individually isolated bird calls.

Pieretti et al have developed an algorithm: Acoustic Complexity Index (ACI) to produce a direct quantification of the complex bird songs by computing the variability of the intensities registered in audio recordings, despite the presence of human generated noise (Pieretti, Farina, & Morri, 2011). This algorithm is based on

the assumption that bird or cicada songs are characterized by having a great change in the intensity, even in short period of time and in a single frequency bin. However, environmental sounds have an almost constant intensity value, which means the difference in the intensity values between two successive frames t and t+1 is small. In addition, their assumption on that ACI filters well the background noise is not proven. This reason motivated us to investigate this index.

The ACI index measures the absolute difference (d_k) between two adjacent values of intensity (I_k and I_{k+1}) in a single frequency bin (Δf_l):

$$d_k = |I_k - I_{k+1}|$$

$D = \sum_{k=1}^{n} d_k$ is the sum of all the d_k contained in the recording's length (j).

Where $j = \sum_{k=1}^{n} \Delta t_k$ $n =$ number of Δt_k in j

In order to obtain the relative intensity, the result D is divided by the total sum of the intensity values recorded in j:

$$ACI_{bin=\Delta f_l} = \frac{D}{\sum_{k=1}^{n} I_k}$$

The ACI obtained here is calculated in a single frequency bin (Δf_l).

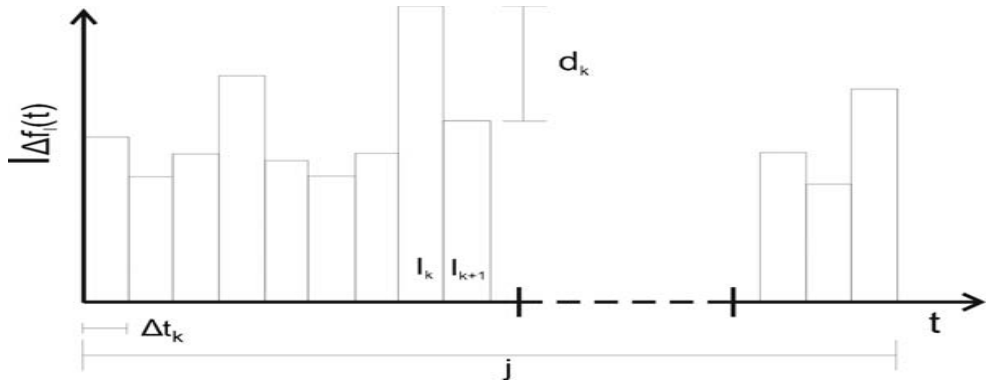


Figure 2.5 Graph Acoustic Complexity Index

(Pieretti, et al., 2011)

Δt_k is a single time fraction; Δf_l is a single frequency bin; $I_{\Delta f_l(t)}$is the intensity registred in Δf_l.

The total ACI for the all frequency bins is calculated:

$ACI_{tot} = \sum_{l=1}^{q} ACI_{bin=(\Delta f_l)}$ Where q is the number of the frequency bins (Δf_l) in the whole recording.

We assume that:

- Birds or cicadas sounds have a significant change in intensity between frames within a single frequency bin producing a high value for ACI.
- For noise like wind, rain, airplane, the change in the intensity is not that much; the variation in the intensity is approximately constant. Therefore, the ACI for these sounds is low.

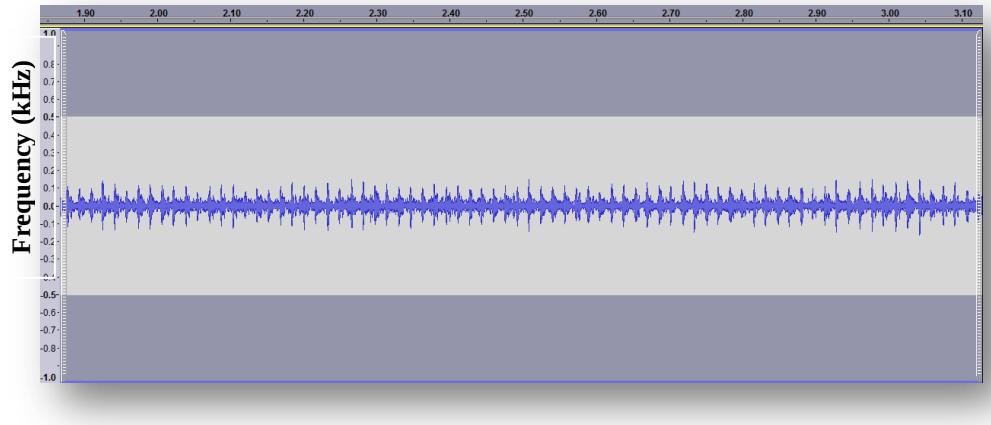
From this hypothesis, ACI might be a good discriminator for *heavy-rain*/*non-heavy rain*.

2.2 PREPROCESSING

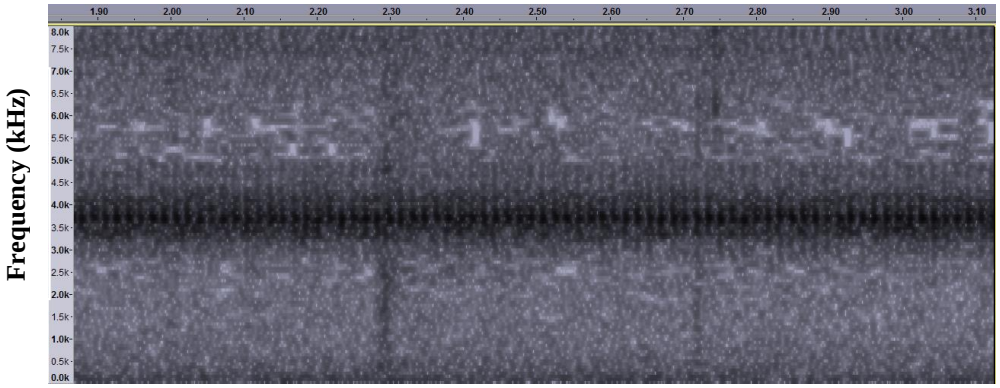
Pre-processing is an important step in a classification process. Its purpose is to form and enhance patterns to be recognized through various processes including signal processing, and noise removal. Because the collected data does not always have good quality, pre-processing is a necessary step for improving the results of subsequent processes.

2.2.1 Signal processing

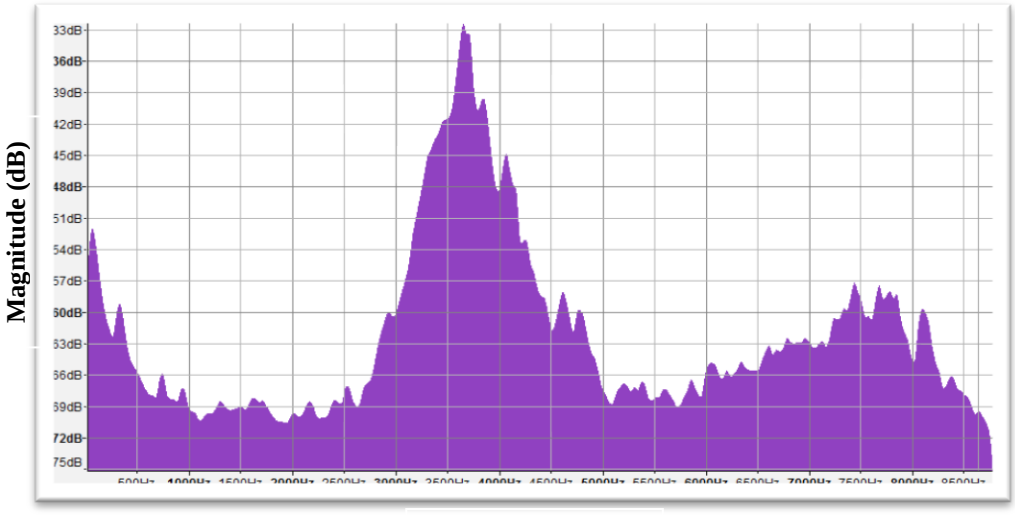
Audio signals are supposed to be processed for constructing an appropriate representation. Generally, Short-Time Fourier Transform (STFT), Fast Fourier Transform (FFT), wavelet transform (WT) and Linear Predictive Coding (LPC) are three main ways to generate the signal representation. Waveform is a basic form of signals in the time domain. To explore more useful visual information of audio signals, Fourier transform is often used to transform time-domain signals into frequency domain signals. Especially, spectrums are formed by Fourier transform and spectrograms are generated by STFT (Cohen, 1995; Griffin & Lim, 1984; Roederer, 2008; Saunders, 1996).



(a) Waveform of a cicada



(b) Spectrogram of a cicada



(c) Spectrum of a cicada

Figure 2.6 Audio signal representation

Figure 2.6 shows a recording of a cicada lasting for three seconds and generated by Audacity software. In Figure 2.6 (a), the x- axis represents time and y-axis is the relative sound pressure level. In Figure 2.6 (b), the x-axis represents time, the y-axis represents frequency and the grey scale represents acoustic intensity. In Figure 2.6 (c), the x-axis represents frequency, and the y-axis represents the dB values of amplitude (power).

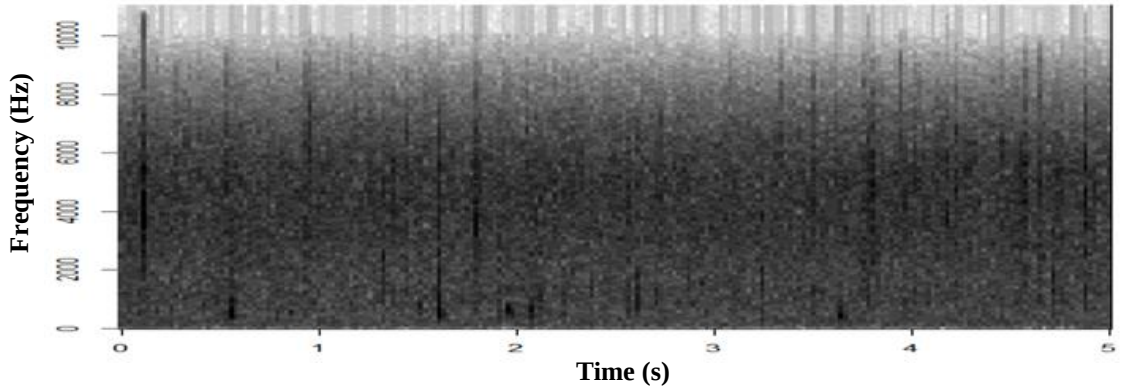
In detail, a waveform figure is plotted in Figure 2.6 (a) through sampling signals derived from a small track of audio file, while a spectrogram in Figure 2.6 (b) and a spectrum in Figure 2.6 (c) are generated through Fourier transform. The spectrogram shows how the frequency values changes over the time, See details of spectrogram generation in the section of signal processing in (Michael. Towsey & Planitz, 2010). The mean spectrum is drawn by computing the average frequency values of an entire signal.

2.2.2 Noise removal

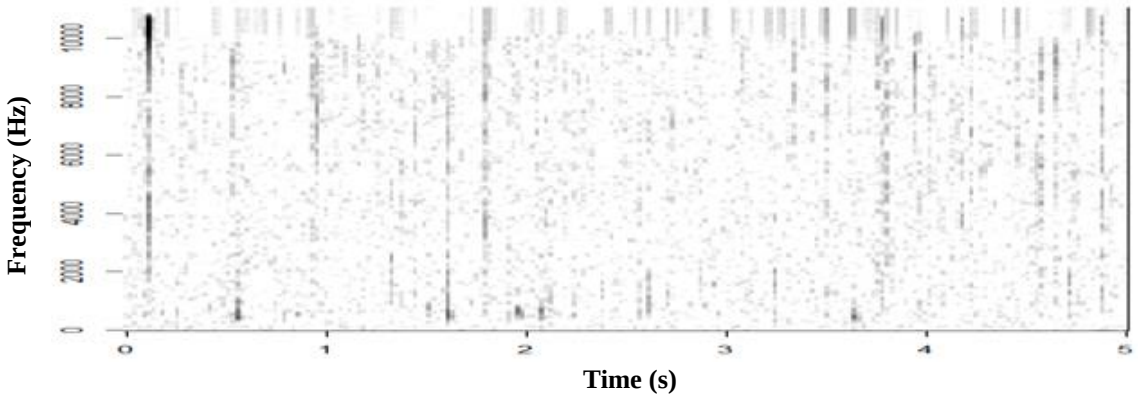
It is important to note that, in the context of audio recordings of the environment, "noise" can have several meanings. Noise does not mean just electronic or microphone noise as engineers understand it. Wind and rain are frequently found in environmental acoustics recordings and are generally considered as noise.

The contribution of noise to recordings of the environment typically declines with increasing frequency. However, we do not assume a standard pink noise model. Rather, we estimate the modal noise power independently for each of the frequency bins in the spectrogram of each one-minute recording. We use a modified version of the same *adaptive level equalisation* algorithm due to Lamel et al (1981). Adaptive level equalisation has the effect of removing continuous background acoustic activity and setting that level to zero amplitude. Thus it becomes possible to define a single absolute threshold for the detection of an acoustic event that spans multiple frequency bins. Note that this modified version can be applied regardless of whether the spectrogram values are converted to decibels or not. Having calculated a threshold intensity value for each frequency bin, we subtract it from each value in that bin (with truncation of negative values to zero), for more details see (M. Towsey, 2013). In our work, spectrograms were not converted to decibels in order to preserve values appropriate for subsequent calculation of ACI and the acoustic

entropy (spectral and temporal entropies). It should be re-emphasised that we performed noise reduction on the two dimensional spectrogram and *not* on the audio recording. The noise removal result can be found in Figure 2.7.



(a) Before noise removal



(b) After noise removal

Figure 2.7 The spectrogram result of rain before and after noise removal

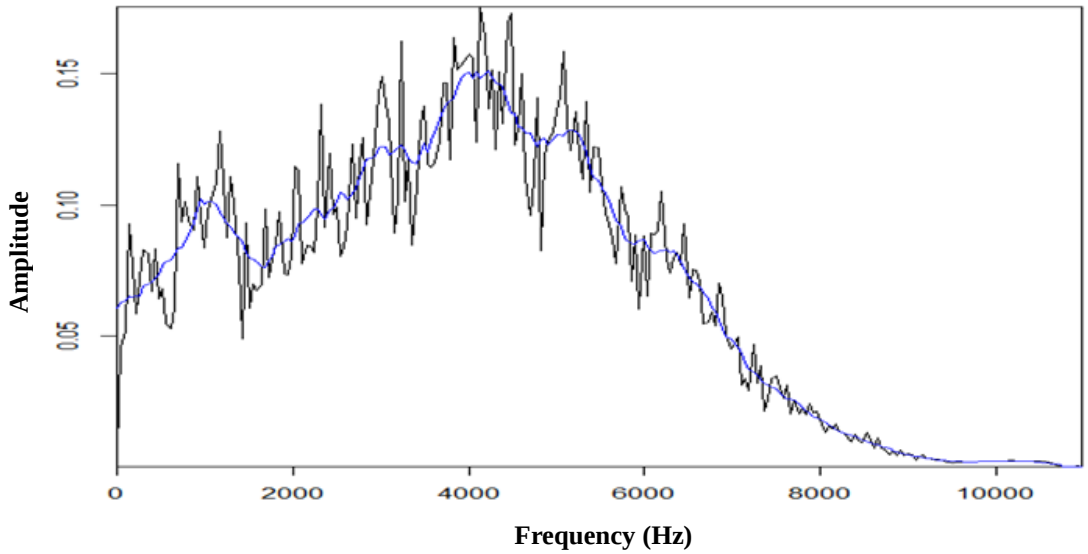


Figure 2.8 Noise intensity versus frequency for a typical spectrogram (rain)

Original and smoothed values are shown

2.3 FEATURE EXTRACTION AND SELECTION

According to Cowling and Sitte (2003) feature extraction can be split into two broad types: stationary (frequency based) feature extraction and non-stationary (time-frequency based) extraction. Stationary feature extraction produces an overall result detailing the frequencies contained on the entire signal. With stationary feature extraction, no distinction is made on where these frequencies occurred in the signal. In contrast, non-stationary feature extraction splits the signal up into discrete time units. This allows frequency to be identified as occurring in a particular area of the signal, aiding understanding of the signal.

Non-stationary feature extraction (Cohen, 1995; Hubbard, 1996; Vapnik, 1999; Zhuang, Zhou, Hasegawa-Johnson, & Huang, 2010) includes:

- Short-time Fourier transform (STFT)
- Fast (discrete) wavelet transform (FWT)
- Continuous wavelet transform (CWT)
- Wigner-Ville distribution (WVD)

Stationary features extraction contain eight popular techniques (listed below) fitted for non-speech sounds (Markel & Gray, 1982; Picone, 1993; Rabiner & Juang, 1993):

- Frequency extraction(FE)
- Cepstral coefficients (CCs)
- Mel-frequency cepstral coefficients (MFCCs)
- Linear predictive coding (LPC)
- Linear prediction cepstral coefficients (LPCCs)
- Mel-frequency LPC coefficients (MFLPCCs)
- Bark frequency cepstral coefficients (BFCCs)
- Bark frequency LPC coefficients (BFLPCCs)

Feature selection is an important issue which must be addressed in designing the feature extraction module. It refers to deciding which features to include in the

feature vector representation. Often, the features are selected using a combination of domain knowledge and experimentation (Pohjalainen, 2007).

In machine learning and statistics, feature selection is one of the most important tasks in a classification algorithm. It allows for a low computational load without increasing the misclassification error. The aim is to obtain an efficient and small vector of acoustic features which represent the input pattern for the classification algorithms being trained.

2.4 MACHINE LEARNING

Artificial Intelligence (AI) is a field of computer science whose objective is to build a systemm that exhibits intelligent behaviour in the tasks it performs. A system can be said to be intelligent when it has learned to perform a task related to the process it has been assigned to without any human interference and with high accuracy. Machine Learning (ML) is a sub-field of AI whose concern is the development, understanding and evaluation of algorithms and techniques to allow a computer to learn. ML interlinks with other disciplines such as statistics, finance, human psychology and brain modeling. Since many ML algorithms use analysis of data for building models, statistics plays a major role in this field.

A process or task that a computer is assigned to deal with can be termed the knowledge or task domain (or just the domain). The information that is generated by or obtained from the domain constitutes its knowledge base. The knowledge base can be represented in various ways using Boolean, numerical, and discrete values, relational literals and their combinations. The knowledge base is generally represented in the form of input-output pairs, where the information represented by the input is given by the domain and the result generated by the domain is the output. The information from the knowledge base can be used to depict the data generation process (i.e., output classification for a given input) of the domain. Knowledge of the data generation process does not define the internals of the working of the domain, but can be used to classify new inputs accordingly.

As the knowledge base grows in size or gets complex, inferring new relations about the data generation process (the domain) becomes difficult for humans. ML algorithms try to learn from the domain and the knowledge base to build

computational models that represent the domain in an accurate and efficient way. The model built captures the data generation process of the domain, and by use of this model the algorithm is able to match previously unobserved examples from the domain.

The models built can take on different forms based on the ML algorithm used. Some of the model forms are decision lists, inference networks, concept hierarchies, state transition networks and search-control rules. The concepts and working of various ML algorithms are different but their common goal is to learn from the domain they represent.

ML algorithms need a dataset to build a model of the domain. The dataset is a collection of instances from the domain. Each instance consists of a set of attributes which describe the properties of that example from the domain. An attribute takes in a range of values based on its attribute type, which can be discrete or continuous. Discrete (or nominal) attributes take on distinct values (e.g., *color = brown, weather = sunny*) whereas continuous (or numeric) attributes take on numeric values (e.g., *distance = 3.5meters, rain = 40mm*).

Each instance consists of a set of input attributes and an output attribute. The input attributes are the information given to the learning algorithm and the output attribute contains the feedback of the activity on that information. The value of the output attribute is assumed to depend on the values of the input attributes. The attribute along with the value assigned to it define a feature, which makes an instance a feature vector. The model built by an algorithm can be seen as a function that maps the input attributes in the instance to a value of the output attribute.

Huge amounts of data may look random when observed with the simple eye, but on a closer examination, we may find patterns and relations in it. We also get an insight into the mechanism that generates the data. Witten and Frank (2005) define data mining as a process of discovering patterns in data. It is also referred to as the process of extracting relationships from the given data. In general data mining differs from machine learning in that the issue of the efficiency of learning a model is considered along with the effectiveness of the learning. In data mining problems, we can look at the data generation process as the domain and the data generated by the domain as the knowledge base. Thus, ML algorithms can be used to learn a model

that describes the data generation process based on the dataset given to it. The data given to the algorithm for building the model is called the training data, as the computer is being trained to learn from this data, and the model built is the result of the learning process. This model can now be used to predict or classify previously unseen examples. New examples used to evaluate the model are called a test set. The accuracy of a model can be estimated from the difference between the predicted and actual value of the target attribute in the test set.

WEKA (Witten, et al., 2005), stands for Waikato Environment for Knowledge Analysis. WEKA is a collection of various ML algorithms, implemented in Java that can be used for data mining problems. Apart from applying ML algorithms on datasets and analysing the results generated, WEKA also provides options for pre-processing and visualization of the dataset. It can be extended by the user to implement new algorithms.

There are different ways an algorithm can model a problem based on the interaction with the input data. It is common in machine learning and data mining to consider the learning styles or learning procedures that an algorithm can adopt. These learning styles are defined as follow:

Supervised learning: the input data is called training data and has known label. A model is prepared through a training process where it is required to make predictions and is corrected when those predictions are wrong. The training process continues until the model achieves a desired level of accuracy on the training data. Example problems are classification and regression.

Unsupervised learning: Input data is not labelled and does not have a known result. A model is prepared by deducing structures present in the input data. Example problems are association rule learning and clustering.

Semi supervised learning: Input data is a mixture of labelled and unlabelled examples. There is a desired prediction problem but the model must learn the structures to organize the data as well as make predictions. Example problems are classification and regression.

We describe in detail the classification and regression algorithms that have been used in this thesis in the following sub-sections.

2.4.1 Classification techniques

Algorithms that classify a given instance into a set of discrete categories are called classification algorithms. These algorithms work on a training set to come up with a model or a set of rules that classify a given input into one of a set of discrete output values. Most classification algorithms can take inputs in any form, discrete or continuous although some of the classification algorithms require all of the inputs also to be discrete. The output is always in the form of a discrete value. Decision Trees classifiers; rule based classifiers, support vectors machines and Naives Bayes classifiers are examples of classification algorithms.

Classification examples include the recognition of bird species present in an audio recording (Duan et al., 2011; Somervuo, Harma, & Fagerlund, 2006; Michael Towsey, Planitz, Nantes, Wimmer, & Roe, 2012), categorizing a piece of sound as bird calls, rain, wind (M. W. Towsey & Planitz, 2011) etc.

Many classification techniques have been used in speech and speaker recognition (Cowling & Sitte, 2003; Temko & Nadeu, 2006) such as:

- Dynamic time warping (DTW).
- Hidden Markov models (HMM).
- Learning quantization vector (LVQ).
- Artificial neural networks (ANN).
- K-Nearest neighbour (kNN).
- Gaussian mixture models (GMM).
- Naives Bayes (NB)
- Decision Tree (DT).
- Support vector machines (SVM).

Most of the research studies argue that supervised machine learning algorithms such as Decision Trees , Artificial Neural Networks , Hidden Markov Models, and Support Vector Machines (Acevedo, Corrada-Bravo, Corrada-Bravo,

Villanueva-Rivera, & Aide, 2009; Mitrovic, et al., 2009; Rokach & Maimon, 2005; Vavrek, et al., 2012) are the best choice for audio classification and segmentation because of their high accuracy. In our research, different classification algorithms were used, a comparison of these algorithms was conducted. We describe in detail the classification algorithms that have been used in this thesis in the following sub-sections.

2.5.1.1 Decision Tree

A Decision Tree is a classifier expressed as a recursive partition of the instance space (Connell & Jain, 2001; Kotsiantis, Zaharakis, & Pintelas, 2007; Rokach & Maimon, 2005; Safavian & Landgrebe, 1991).

Decision Tree is a branching that represents a set of rules, distinguishing values in a hierarchical form. This representation can be translated into a set of IF-THEN rules, which are easily understood.

Generally the tree has three types of nodes:

- *A root node* that has no incoming edges and zero or more outgoings edges.
- *Internal nodes*, each which have exactly one incoming edge and two or more outgoing edges.
- *Leaf or terminal nodes,* each which have exactly one incoming edge and no outgoing edges.

In a Decision Tree, each leaf node is assigned a class label. The non-terminal nodes which include the root and other internal nodes, contain attribute test conditions to separate instances that have different characteristics. Instances are classified by navigating them from the root node of the tree down to a leaf node, according to the outcome of the tests along the path.

For example, Figure 2.9 uses the attribute/feature ***temporal entropy*** to separate *heavy-rain* from *non-heavy rain.* The first split is on the *temporal entropy* attribute/feature, and then at the second level, the split is on *temporal entropy again.*

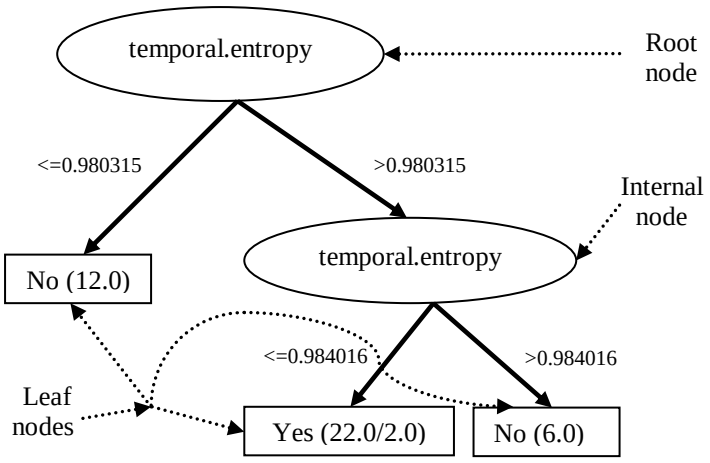


Figure 2.9 A Decision Tree for rain classification problem

This example is performed on a dataset of 40 recordings manually labelled: *heavy rain* and *non-heavy rain* equally, using two attributes/features namely temporal and spectral entropy and using 10 fold-cross validation.

In the tree structure, a colon introduces the class labels that have been assigned to a particular leaf, followed by the number of instances that reach that leaf, expressed as decimal number because of the way the algorithm uses fractional instances to handle missing values, (12.0) means that 12 instances reached that leaf, of which all the instances are classified as *non-heavy rain.*

Note that this Decision Tree incorporates only numeric attributes. Given this classifier, we can predict whether an acoustic recording contains heavy *rain or not* (by sorting it down the tree). Each node is labelled with the attribute it tests, and its branches are labelled with its corresponding values.

Naturally, decision-makers prefer less complex Decision Tree, since they may be considered more comprehensive (Rokach & Maimon, 2005). Furthermore according to (Breiman, Friedman, Stone, & Olshen, 1984) the tree complexity has a crucial effect on its accuracy performance. The tree complexity is explicitly controlled by the stopping criteria used and the pruning method employed. Usually, the *tree complexity* is measured by one of the following metrics:

- The total number of nodes;
- Total number of leaves;

- The depth and ;
- Number of attributes used.

2.5.1.2 Support vector machines

Support vector machines (Fagerlund, 2007; Guo & Li, 2003; Huang, Yang, Yang, & Chen, 2009) are fundamentally binary classifiers, but any number of classes can be accommodated by combining binary SVM classifiers. The principle of SVM classification can be described by first considering linearly separable classes, i.e., two classes which can be perfectly separated using a linear hyperplane as a decision boundary. SVM training is based on the idea of maximizing the margin between any decision boundary and the closest observation at each side of the hyperplane, i.e., the goal is to maximize the distance from the closest class representative points to the decision boundary. These representatives are called support vectors. The optimization problem of designing a maximum margin hyperplane can be solved using Lagrange multipliers.

In the general case in which the classes are not separable even with a nonlinear decision boundary, the nonlinear SVM classifier effectively maps the feature vectors into a higher dimensional space in which linear separation of the training set is possible. The margin is then maximized in the higher-dimensional space during the training procedure. Maximization of the margin in SVM training aims for improved generalization performance of the classifier when presented with previously unseen data.

2.5.1.3 Naive Bayes classifier

Naive Bayes classifier is a statistical classifier. It can predict class membership probabilities, such as the probability that a given sample belongs to a particular class (Good, 1965; Langley & Sage, 1994). Bayesian classification is based on Bayes theorem. Studies comparing classification algorithms have found a simple Bayesian classier known as the *Naïve Bayesian Classifier* to be comparable in performance with Decision Tree and neural network classifiers. The naive Bayes algorithm builds a probabilistic model by learning the conditional probabilities of each input attribute given a possible value taken by the output attribute. This model is then used to predict an output value when we are given a set of inputs. This is done by applying Bayes theorem on the conditional probability of seeing a possible output

value when the attribute values in the given instance are seen together. Before describing the algorithm we first define the Bayes theorem.

Bayes theorem states that:

$$P(A\backslash B) = \frac{P(B\backslash A)P(A)}{P(B)}$$

Where *P(A|B)* is defined as the probability of observing *A* given that *B* occurs. *P(A|B)* is called posterior probability, and *P(B|A)*, *P(A)* and *P(B)* are called prior probabilities. Bayes theorem gives a relationship between the posterior probability and the prior probability. It allows one to find the probability of observing *A* given *B* when the individual probabilities of *A* and *B* are known, and the probability of observing *B* given *A* is also known.

The Naive Bayes algorithm uses a set of training examples to classify a new instance given to it using the Bayesian approach. For an instance, the Bayes theorem is applied to find the probability of observing each output class given the input attributes and the class that has the highest probability is assigned to the instance. The probability values used are obtained from the counts of attribute values seen in the training set.

The Naive Bayes algorithm requires all attributes in the instance to be discrete. Continuous valued attributes have to be discretised before they can be used. Missing values for an attribute are not allowed, as they can lead to difficulties while calculating the probability values for that attribute. A common approach to deal with missing values is to replace them by a default value for that attribute. Bayesian classifiers have also exhibited high accuracy and speed when applied to large datasets (Hall et al., 2009; Kohavi, 1996).

2.5.1.4 Lazy classifier

Lazy learners store the training instances and do no real work until classification time. IB1 is a basic instance-based learner that finds the training instance closest in Euclidean distance to the given test instance and predicts the same class as this training instance. If several instances qualify as the closest, the first one found is used.

Lazy IBk stands for Instance based learner with fixed neighbourhood k (Cufoglu, Lohi, & Madani, 2008). IBk is a k-nearest neighbour method which consists of assigning to the unlabelled feature vector the label of the training vector that is nearest to it in the feature space (Ke, Heng Tao, Kai, & Xuemin, 2006). In kNN, a training set T is used to determine the class of a previously unseen sample x. First, we determine the mean and maximum values in T, and similarly, for the unseen sample x. Then a suitable distance measure in the feature space is used to determine k elements in T closest to x. If most of these k nearest neighbours contain similar values, then x gets classified accordingly. The "Closeness" is defined in terms of Euclidean distance, where the Euclidean distance between two points $X = (x_1, x_2, \cdots, x_n)$ and $Y = (y_1, y_2, \cdots, y_n)$ is $d(X, Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$

This classification scheme clearly defines nonlinear decision boundaries and thus improves the performance. Furthermore, the feature distribution suggests that the number of data-points used in the example set T can be considerably reduced for faster processing; only those examples that are close to the decision boundary are actually required.

Mporas et al. (2012) evaluated six different classification algorithms implemented in WEKA to classify seven species of birds. These classifiers namely are: the k-nearest classifier (IBk), 3-layer Multilayer perceptron (MLP), support vector machines (SMO), the Bayes network learning (Bayes Net). They used two temporal features (the frame intensity (Int) and the zero crossing rate (ZCR)). They used sixteen spectral features (12 first MFCCs, the root mean square energy of the frame (E), the voicing probability (Vp), the harmonics-to-noise ratio (HNR), and the dominant frequency (fd)). The highest recognition accuracy was achieved by bagging and boosting meta- classification algorithm, which used the pruned C4.5 Decision Tree as base classifier.

2.4.2 Regression algorithms

Algorithms that develop a model based on equations or mathematical operations on the values taken by the input attributes to produce a continuous value to represent the output are called of regression algorithms. The input to these algorithms can take both continuous and discrete values depending on the algorithm, whereas the output is a continuous value.

Regression algorithms have been used in many areas such as biodiversity prediction, forest fire detection, stream-flow, finance, health, etc. (Onyari & Ilunga, 2010; Parra Jr & Kiekintveld, 2013). Forest fires are a major environmental issue, creating economical and ecological damage while endangering human lives. Detecting these fires is a key element in controlling such a phenomenon. Cortez et al (2007) have proposed to use real-time and non-costly meteorological data collected by sensors (in Portugal) instead of using satellite images, infrared/smoke scanners, satellite images combined with meteorological data which are costly, to predict the burned area (or size) of forest fires. They have conducted several experiments considering five data mining techniques including (multiple regression, Decision Trees, random forest, neural networks, and support vector machines), and four selection setup (using spatial, temporal, the fire weather index system and meteorological data). In their proposed solution they have used four weather variables (rain, wind, temperature and humidity) combined with SVM to predict the burned area of small fires, their method could be useful in fire management (e.g. resource planning).

The methods we have seen in Decision Tree and rules work most with nominal attributes. They can be extended to numeric attributes either by incorporating numeric-value tests directly into the Decision Tree or rule-induction scheme or by pre-discretising numeric attributes into nominal ones. We describe in detail the prediction algorithms that have been used in this thesis in the following sub-sections.

2.5.2.1 Linear regression

When the outcome or class is numeric, and all attributes are numeric, linear regression is a good technique to consider.

The linear regression algorithm of WEKA (Wang & Witten, 1997) performs standard least squares regression to identify linear relations in the training data. This algorithm gives the best results when there is linear dependency among the data. It requires the input attributes and target class to be numeric and it does not allow missing attributes values. The algorithm calculates a regression equation to predict the output (x) for a set of input attributes/features a_1, a_2, ..., a_k. The equation to calculate the output is expressed in the form of a linear combination of input attributes with each attribute associated with its respective weight w_0, w_1, ..., w_k, where w_1 is the weight of a_1 and

a_0 is always taken as the constant 1 (Witten, et al., 2005) . The regression equation takes the form:

$$x = w_0 + w_1 a_1 + \cdots\cdots\cdots + w_k a_k$$

For our *rain* example the equation learned would take the form:

$$Rain = w_0 + (w_1 \times H_t) + (w_2 \times H_f) + (w_3 \times ACI) + (w_4 \times BgN) + (w_5 \times SC)$$

Once the math has been accomplished, the result is a set of numeric weights, based on the training data, which can be used to predict the class of new instances.

2.5.2.2 M5P

The M5P or M5Prime algorithm (Wang & Witten, 1997) is a regression-based Decision Tree algorithm, based on the M5 algorithm developed by (Quinlan, 1992). M5P is developed using M5 with some additions made to it. We will first describe the M5 algorithm and then the features added to it in M5P.

M5 builds a tree to predict numeric values for a given instance. The algorithm requires the output attribute to be numeric while the input attributes can be either discrete or continuous. A discrete attribute can be numeric (such as a number of birds) or categorical (such as the gender; male or female). For a given instance the tree is traversed from top to bottom until a leaf node is reached. At each node in the tree a decision is made to follow a particular branch based on a test condition on the attribute associated with that node. Each leaf has a linear regression model associated with it of the form: $w_0 + w_1 a_1 + \cdots\cdots\cdots + w_k a_k$

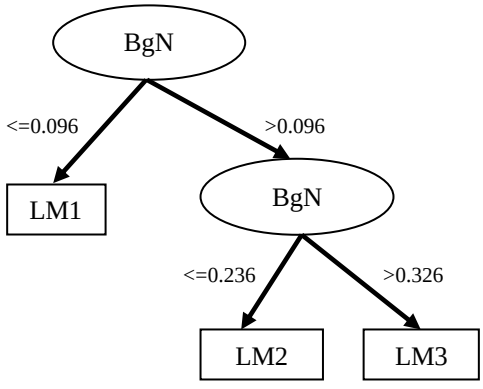


Figure 2.10 A model tree for predicting rain in an audio recording

The decision taken at a node is based on the test of the attributes mentioned at that node. Each model at a leaf takes the form $w_0 + w_1 a_1 + \cdots + w_k a_k$ where k is the number of input attributes.

Based on some of the input attributes $a_1,a_2,.....,a_k$ in the instance and whose respective weights $w_0,w_1,....,w_k$ are calculated using standard regression. As the leaf nodes contain a linear regression model to obtain the predicted output, the tree is called a model tree. When the M5 algorithm is applied on our rain example, the model tree generated will take a form as shown in Figure 2.10.

To build a model tree, using the M5 algorithm, we start with a set of training instances. The tree is built using a *divide-and-conquer* method which works by recursively breaking down a problem into two or more sub-problems of the same (or related) type, until these become simple enough to be solved directly. The solutions to the sub-problems are then combined to give a solution to the original problem.

At a node, starting with the root node, the instance set that reaches it is either associated with a leaf or a test condition is chosen that splits the instances into subsets based on the test outcome. A test is based on an attributes value, which is used to decide which branch to follow. There are many potential tests that can be used at a node. In M5 the test that maximizes the error reduction is used. For a test the expected error reduction is found using:

$$SDR = stdev(S) - \sum_i \frac{|S_i|}{|S|} stdev(S_i)$$

where S is the set of instance passed to the node, *stdev (S)* is its standard deviation, S_i is the subset of S resulting from splitting at the node with the i^{th} outcome for the test. This process of creating new nodes is repeated until there are too few instances to proceed further or the variation in the output values in the instances that reach the node is small.

Once the tree has been built, a linear model is constructed at each node. The linear model is a regression equation. The attributes used in the equation are those that are tested or are used in linear models in the sub-trees below this node. The attributes tested above this node are not used in the equation as their effect on predicting the output has already been captured in the tests done at the above nodes. The linear model built is further simplified by eliminating attributes in it. The attributes whose removal from the linear model leads to a reduction in the error are eliminated. The error is defined as the absolute difference between the output value predicted by the model and the actual output value seen for a given instance.

The tree built can take a complex form. The tree is pruned so as to make it simpler without losing the basic functionality. Starting from the bottom of the tree, the error is calculated for the linear model at each node. If the error for the linear model at a node is less than the model sub-tree below then the sub-tree for this node is pruned. In the case of missing values in training instances, M5P changes the expected error reduction equation to:

$$SDR = \frac{m}{|S|} \times \left[stdv(S) - \sum_{j \in \{L,R\}} \frac{|S_j|}{|S|} stdev(S_j) \right]$$

where *m* is the number of instances without missing values for that attribute, *S* is the set of instances that reach this node. S_L and S_R are sets obtained from splitting on this attribute.

2.5.2.3 RepTree

RepTree builds a decision or regression tree using information gain/variance reduction and prunes it using reduced-error pruning. Optimised for speed, it only sorts values for numeric attributes once. It deals with missing values by splitting instances into pieces, as C4.5 algorithm does.

You can set the minimum number of instances per leaf, the maximum tree depth (useful when boosting trees), the minimum proportion of training set variance for a split (numeric classes only), and number of folds for pruning.

2.5.2.4 Multi-layer-perceptron

A Multilayer Perceptron (MLP) (Bishop, 1995) is a neural network that is trained using back-propagation. Back-propagation is a supervised learning method where the algorithm works towards minimising the error between its output and the target. MLP consists of multiple layers of computational units that are connected in a feed-forward way forming a directed connection from lower units to a unit in a subsequent layer. The basic structure of MLP consists of an input layer, one or more hidden layers and one output layer. Units in the hidden layer are termed *hidden* as their output is used only in the network and is not seen outside the network.

It is explained literature (Witten, et al., 2005) that it appears in practice that the back-propagation method leads to solutions in almost every case, although, the

error back-propagation method does not guarantee convergence to an optimal solution since local minima may exist.

MLP consist of multiple layers of computational units that are connected in a feed-forward way forming a directed connection from lower units to a unit in a subsequent layer. The basic structure of MLP consists of an input layer, one or more hidden layers and one output layer. Units in the hidden layer are termed *hidden* as their output is used only in the network and is not seen outside the network.

2.5.2.5 Decision table

Decision table builds a decision table majority classifier. It evaluates the feature subset using best-first search and can use cross-validation for evaluation (Kohavi, 1995). An option uses the nearest-neighbour method to determine the class for each instance that is not covered by a decision table entry, instead of the table's global majority, based on the same set of features.

2.5 EVALUATION METRICS

2.5.1 Evaluating classification techniques

It is very important to evaluate the performance of the classification algorithms that will be used in this research. The following metrics can be used to evaluate the algorithms: True Positives (TP), True Negatives (TN), False Negatives (FN), False Positives (FP) and F-score are defined followed the definition in the paper of (Picone, 1993):

- TP: correctly recognized positives
- TN: correctly recognized negatives
- FN: positives recognized as negatives
- FP: negatives recognized as positives
- F-score is the mean of the precision and recall

Precision and recall are two widely used statistical criteria. Precision can be seen as a measure of exactness or fidelity, whereas recall is a measure of completeness. They are defined in Table 2.2 (Olson & Delen, 2008).

Table 2.2 Metrics used to evaluate the performance of the classification algorithms.

Performance Measures for Classification Algorithms	
Precision	$\frac{\text{TP}}{\text{TP+FP}}$
Recall	$\frac{\text{TP}}{\text{TP+FN}}$
Accuracy	$\frac{\text{Recall+Precision}}{2}$
F_{score}	$\frac{2*\text{Precision}*\text{Recall}}{\text{Precision+Recall}}$

- *Confusion matrix* is another well-known measure adopted in the literatures (Brandes, Naskrecki, & Figueroa, 2006; Giret, Roy, & Albert, 2011; Vaca-Castaño & Rodriguez, 2010) to measure the confusion among different classes. In machine learning, a confusion matrix is a special table layout that allows the visualisation of the performance of an algorithm, typically supervised one where the class is predefined. Each column of the matrix represents the instance in a predicted class, while each row represents the instance in an actual class.

2.5.2 Evaluating numeric predictions

Several measures are used to evaluate the numeric predictions (Witten, et al., 2005) and some of them are summarised in Table 2.3.

The predicted values on the test instances are: $p_1, p_2, \dots p_n$; the actual values are $a, a_2, \dots a_n$. Notice that p_i refers to the numerical value of the prediction for the test instance i^{th}.

Table 2.3 Measures used for the evaluation of numeric predictions.

Performance Measures for Numeric Predictions	
Mean-squared error	$\frac{(p_1-a_1)^2+\cdots+(p_n-a_n)^2}{n}$
Root mean-squared error (RMSE)	$\sqrt{\frac{(p_1-a_1)^2+\cdots+(p_n-a_n)^2}{n}}$
Mean-absolute error (MAE)	$\frac{\lvert p_1-a_1\rvert+\cdots+\lvert p_n-a_n\rvert}{n}$
Relative-squared error	$\frac{(p_1-a_1)^2+\cdots(p_n-a_n)^2}{(a_1-\bar{a})^2+\cdots+(a_n-\bar{a})^2}$
Root relative-squared error	$\sqrt{\frac{(p_1-a_1)^2+\cdots(p_n-a_n)^2}{(a_1-\bar{a})^2+\cdots+(a_n-\bar{a})^2}}$
Relative-absolute error	$\frac{\lvert p_1-a_1\rvert+\cdots+\lvert p_n-a_n\rvert}{\lvert a_1-\bar{a}\rvert+\cdots+\lvert a_n-\bar{a}\rvert}$
Correlation coefficient (R^2)	$\frac{S_{PA}}{\sqrt{S_P S_A}}$, where $S_{PA}=\frac{\sum_i(p_i-\bar{p})(a_i-\bar{a})}{n-1}$, $S_P=\frac{\sum_i(p_i-\bar{p})^2}{n-1}$, $S_A=\frac{\sum_i(a_i-\bar{a})^2}{n-1}$
Here, $\bar{a}$ is the mean value over the training data. *Here, $\bar{p}$ is the mean value over the test data.*	

2.6 SUMMARY

This Chapter first reviewed the literature on sound classification. The classification process relies on three steps which are pre-processing, feature extraction and classification. Many studies have been conducted on these tasks and researchers are attempting to explore new and effective algorithms or tools for specific applications.

First of all, pre-processing is an important step in the early stage. It mainly covers two main tasks, signal processing and noise removal. Signal processing is a necessary task in this step for almost all classification processes. Thus, we will also pre-process the data used in our study. Feature extraction is a significant task which aims to reduce the original data into a small amount of feature vectors, and these features should be sufficient to discriminate different type of classes. Basically, four types of classification algorithms are summarized in Section 2.4.1, and five types of

prediction algorithms are summarized in Section 2.4.2. Each algorithm has its own advantages and disadvantages depending on the application. We tested all these algorithms in our experiments in order to find out the most suitable ones for the problem considered in this research.

We have summarized many studies on environmental sound classification, using different combination of features, different classes of environmental sounds and different combinations of features. However, there is no existing research that has investigated rain classification or prediction via acoustic analysis. In our work, we explore the new application of set of features used widely in environment monitoring: acoustic complexity index (ACI), temporal entropy (Ht), spectral entropy (Hf), background noise (BgN), and spectral cover (SC) for the detection of rain in acoustic recordings of the environment. The combination of these features hasn't been investigated in previous works for rain classification or prediction.

Chapter 3: **Research Plan**

This Chapter describes the design adopted by this research to achieve the aims and objectives stated in Section 1.3 of Chapter 1. Section 3.1 outlines the methodology used in the study, the stages by which the methodology was implemented, and the research design; Section 3.2 presents the procedure used in the study; Section 3.3 describes the preparation of different datasets; Section 3.4 discusses feature extraction; Section 3.5 gives classifier selection; and finally Section 3.6 describes the software tools used in this research.

3.1 METHODOLOGY AND RESEARCH PLAN

This Section outlines the specific research tasks for this research project in order to address the research questions outlined in Section 1.3. The research will rely on: selecting audio data, extracting useful features, exploring different machine learning techniques in classifying the content of acoustic recording based on the feature set extracted, and evaluating the performance of the classification system using different metrics.

3.1.1 Procedures and approaches

This research aims to detect and classify different types of environmental sounds using audio data directly collected from the field. It is composed of four main tasks: dataset preparation, feature extraction, classification and regression.

Collecting sounds from the wild and analysing these sounds properly is important in environmental monitoring. The first part of this research will focus on the classification of these environmental sounds (*rain, cicadas, silence, animal sounds*) using different machine learning algorithms (classification algorithms). The second part of this research is to explore different machine learning algorithms (regression algorithms) in detecting rain in acoustic recordings of the environment. The proposed research will address the research questions outlined in Section 1.3 and follow an iterative and incremental methodology. The basic classification and regression processes are shown in Figure 3.1.

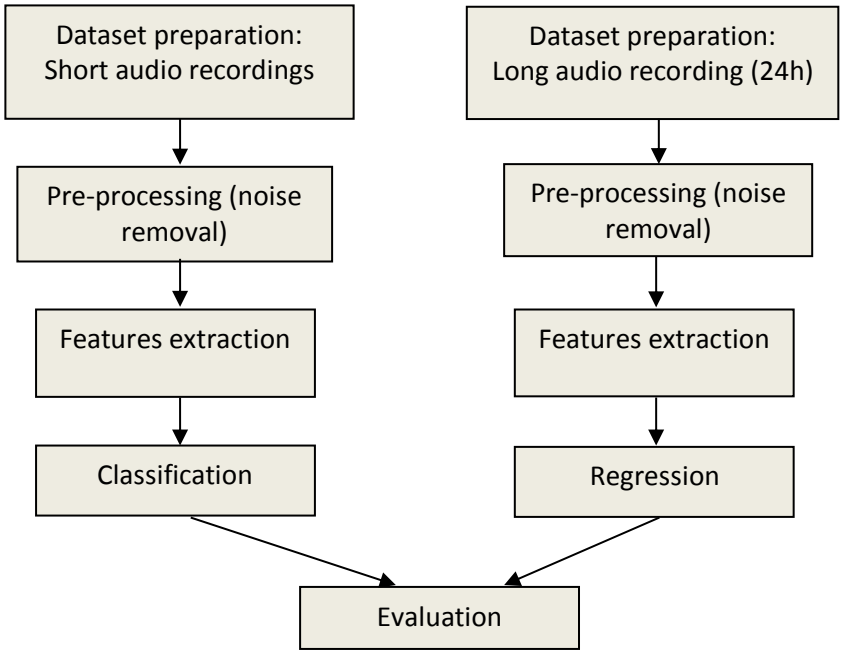


Figure 3.1 Flow chart of the classification and regression processes

This research uses an existing environmental dataset. However, the dataset needs to be pre-processed so that it can be used for evaluating the classification algorithms. In the first phase, the data collection is pre-processed, followed by audio features selection in the second phase. The third phase consisted of choosing a set of classification algorithms which use the extracted features (inputs for the algorithm) to select the class of the sound (outputs of the algorithm). Finally, the developed system is evaluated using some metrics and the results are used to refine and improve the feature selection and classification algorithms.

3.1.2 Hardware for data collection

Recordings were obtained using custom-developed acoustic sensors (Wimmer, Towsey, Planitz, Roe, & Williamson, 2010). The recording equipment consisted of Olympus DM-420 (Olympus, Pennsylvania, USA) digital recorders and external omni-directional electret microphones. Data were stored internally in stereo MP3 format (128 kbits/s, 22.05 kHz) on high capacity 32 GB Secure Digital memory cards. The units were stored in weatherproof cases and powered by four D cell

batteries, providing up to 20 days of continuous recording. Although MP3 is a lossy format, it is designed to reproduce sound accurately for the human ear.

3.2 DATA SETS PREPARATION

3.2.1 Description of heavy rain

Rain in the spectrogram is seen as vertical lines, and often occupies the whole frequency band. When listening to the audio recording, heavy rain can be easily differentiated by human ear from other present sounds.

3.2.2 Signal acquisition

Signals were acquired using an acoustic data logger configured for continuous recording over 24 hours (Wimmer, et al., 2010). All recordings were sampled at 22,050 Hz and a bit rate of 16. Long recordings were subsequently split into one minute segments. The signal is framed using a window of 256 samples (11.6ms) which offers a reasonable compromise between time and frequency resolution. A Hamming window function is applied to each frame prior to performing a Fast Fourier Transform (FFT), which yields amplitude values for 128 frequency bins, each spanning 86.13 Hz. A spectrogram is formed after FFT. Each pixel represents one frame covering 256 samples and one frequency bin spanning 86.13 Hz.

3.3 DATA SETS SELECTION

We have selected two different datasets: *dataset A* and *dataset B*.

3.3. 1 Dataset A (manual segments labelling)

Dataset A is used for the classification problems. Recordings were obtained by use of acoustic sensors from the *Samford Ecological Research Facility (SERF)* in *bush-land on the outskirts of Brisbane city, Queensland, Australia*. To make Dataset A more realistic, the recordings were selected from a wide variety of different sites, different days, and different time in the day (precisely 33 days and four sites in SERF). We used an audio browser which uses acoustic indices developed by Towsey (2012b) to scan through each of the 24 hour recordings to find segments of interest. Interesting segments were examined in *Audacity*, which allowed for aural and visual inspection of the signal. *Dataset A* contains 998 five seconds-long segments. Five seconds were chosen empirically (based on observed patterns of rain starting and stopping) as the

classification resolution for this experiment. Each segment is manually labeled into one of seven classes: *heavy rain*, *cicada chorus*, *bird calls*, *frog calls*, *koala bellow*, *light rain*, and *low-activity (night time/silence)*. Table 1 shows the composition of the Dataset A.

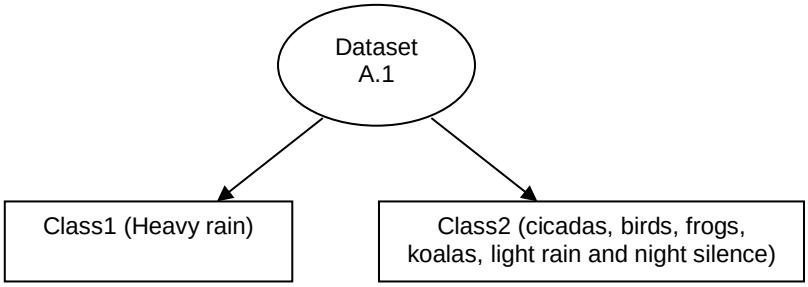
When inspecting the dataset, classes were created to discriminate between the types of acoustic data that were observed. For example most of the recordings include cicada choruses which are continuous (much like rain) but have different acoustic properties in the time-frequency domain. Rain presents as two different visual features in a spectrogram: The first, a general increase in background noise is produced by rain. The second distinct feature seen is vertical broadband lines on the spectrograms – these are percussive drops on the audio senor's housing. Cicadas occupy a certain frequency band between 2kHz and 4kHz. Birds occupy a different frequency band and different species have different call structures. The acoustic Entropy feature can describe this information and constitutes the main feature for classifying these classes. While labeling the training data for rain events, other acoustic classes were also labeled, originally to assist in explaining the classification results. Additionally labeled events include periods of night-time/silence/low activity.

Table 3.1 Composition of Dataset A.

<table>
<tr><th rowspan="2">Classes</th><th rowspan="2">Count</th><th>Dataset A.1</th><th>Dataset A.2</th><th>Dataset A.3</th></tr>
<tr><th>2 Class-problem</th><th>3 Class-problem</th><th>4 Class-problem</th></tr>
<tr><td>Heavy rain</td><td>244</td><td rowspan="7">2</td><td>1</td><td>1</td></tr>
<tr><td>Cicada chorus</td><td>193</td><td rowspan="6">3</td><td>2</td><td>2</td></tr>
<tr><td>Bird calls</td><td>483</td><td rowspan="3">3</td></tr>
<tr><td>Frog calls</td><td>16</td><td rowspan="2">4</td></tr>
<tr><td>Koala bellow</td><td>2</td></tr>
<tr><td>Light rain</td><td>17</td><td rowspan="2">4</td></tr>
<tr><td>Low-activity</td><td>43</td></tr>
<tr><td>Total</td><td>998</td><td>244/754</td><td>244/193/561</td><td>244/193/501/60</td></tr>
</table>

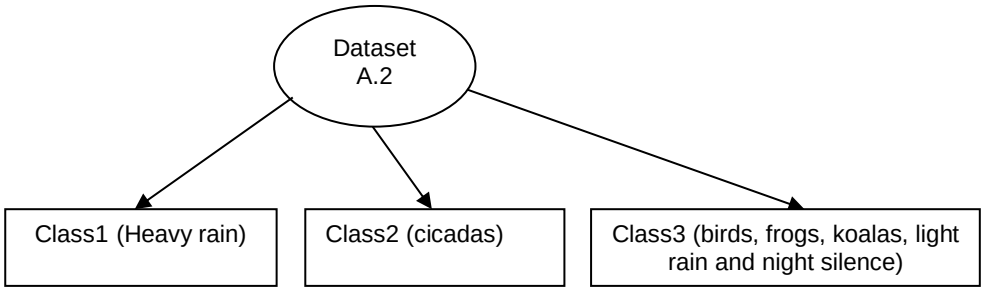
This part describes the construction of the different datasets. In total we constructed three datasets (83minutes each), as shown in Table 3.1.

1) For the binary classification (Dataset A.1), we split the data into two classes: class1 (heavy rain), class2 (cicada chorus, bird calls, frog calls, koala bellow, light rain, night silence).

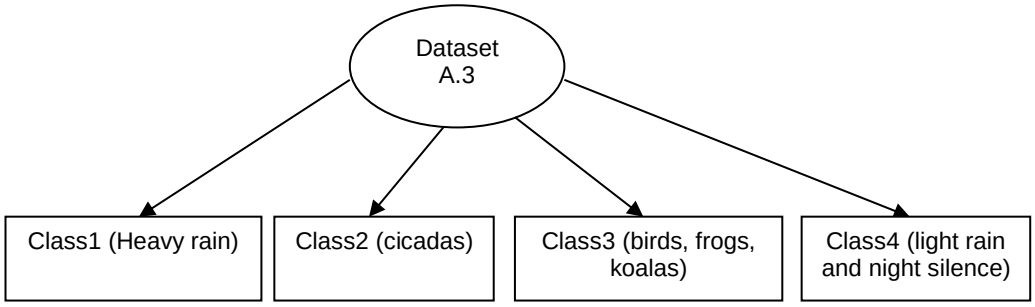


2) For the multi-class problem, we used tow datasets:

- Dataset A.2 contains three classes: class1 (heavy rain), class2 (cicada chorus), and class3 (bird calls, frog calls, koala below, light rain, night silence).



- Dataset A.3 contains four classes: class1 (heavy rain), class2 (cicada chorus), class3/animal sounds (bird calls, frog calls, and koala below), and class4 (light rain, night silence).



3. 3.2 Dataset B (long audio recording)

Dataset B is used for the regression technique quantitative prediction of rainfall.

Dateset B is a 24-hour MP3 recording derived from north east of SERF (core vegetation plot site), on the 13th April 2013. Figure.2 is a false-color spectrogram of a 24-hour recording obtained using the method described by Towsey et al., (2014). The x-axis extends from midnight to midnight. Since the x-axis scale is one pixel-column per minute, a greater than 2000x compression is achieved over the standard spectrogram. Note that the frequency scale is unchanged. The bottom image in Figure 3.2 is a grey-scale representation of the content of the environment of that particular day. The image shows that the source audio does not only contain rain, but also contain *crickets*, as well as other faunal vocalizations.

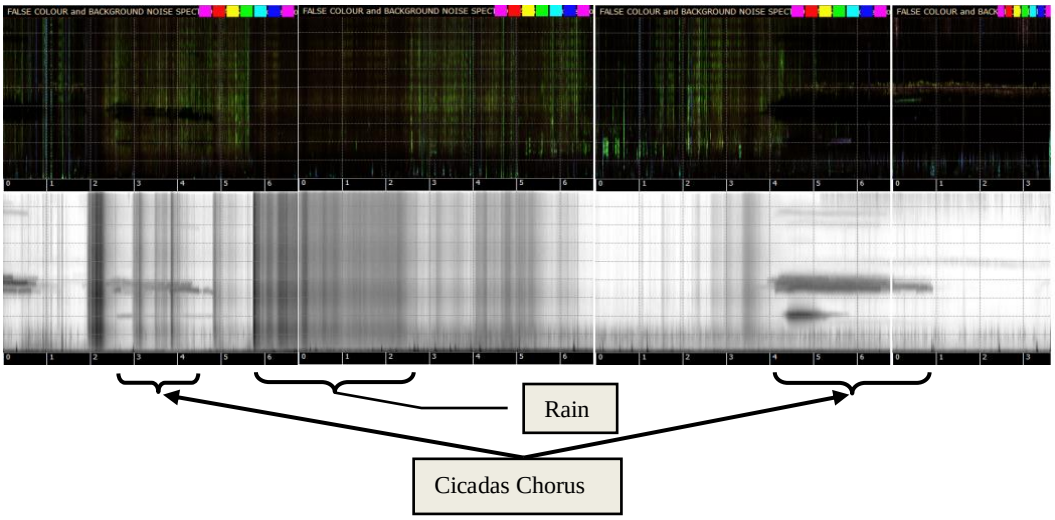


Figure 3.2 Visualization of 24-hour long duration acoustic recordings of the environment.

The steps taken to prepare the *dataset B* are summarized in Figure 3.3:

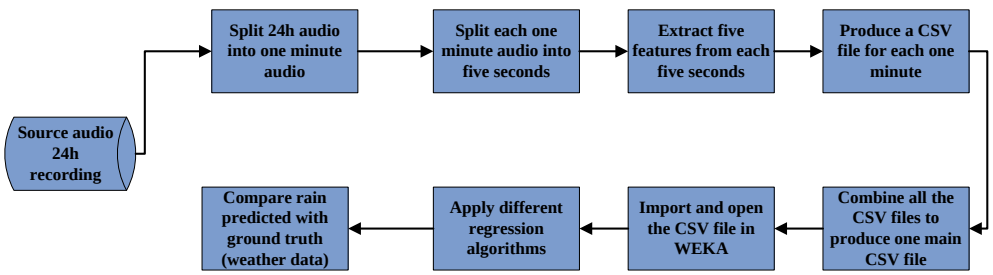


Figure 3.3 Flowchart for 24-hour long data preparation.

3. 3.4 Test dataset

To test the usefulness of trained classifier on real world rainfall data, rainfall values were obtained from the weather station at SERF. Rainfall was measured at mm resolution over intervals of 5min.

3.4 FEATURE EXTRACTION

The general aim of feature extraction is to reduce the original audio data into a compressed amount of feature vectors. Selected features should provide characteristic information about a signal so that similar signals will be grouped together, and the dissimilar ones will be in difference class. A wide range of features have been explored for sounds classification. Basically they can be derived from either time domain or frequency domain.

As discussed in Section 2.1.8, a large amount of features has been used in classifying environmental sounds. For our purpose, we choose and extract five features for environmental sounds: acoustic entropy (H) for which we calculated spectral and temporal entropy (H_f, H_t) respectively, acoustic complexity index (ACI), background noise (BgN) and spectral cover (SC). These features were chosen because previous authors have shown that they are useful for discriminating acoustic activity due to biological sources (Michael Towsey, Parsons, & Sueur, 2014).

The features extracted are described in the following paragraph.

A) ***The Temporal Entropy Index*** $\boldsymbol{H_t}$ and *the* ***Spectral Entropy Index*** $\boldsymbol{H_f}$ are computed following their definitions in (Sueur, et al., 2008) :

$$H_t = -\sum_{t=1}^{n} A(t) \times \log_2(A(t)) \times log_2(n)^{-1}$$

$$H_f = -\sum_{f=1}^{N} S(f) \times log_2\big(S(f)\big) \times log_2(n)^{-1}$$

where n is the length of the signal in number of digitized points; A(t) is the probability mass function of the amplitude envelope; and S(f) is the probability mass function of the mean spectrum calculated using a short term Fourier transform (STFT) along the signal with non-overlapping Hamming window of N=512 points.

The following examples illustrate the temporal and spectral entropies:

- ***Temporal Entropy (Ht)*:** The acoustic energy is spread along the recording. Hence, the temporal entropy is high. Example: the spectrogram of a cicada which can be found in Section 2.2.1, Figure 2.6 (b) is a good illustration ***Spectral Entropy (Hf)***: The acoustic energy is concentrated in a certain frequency band. Example: the spectrum of a cicada which can be found in Section 2.2.1, Figure 2.6 (c) is a good example.

B) *The Acoustic Complexity Index* (ACI) is based on the assumption that bird sounds are characterized by having a great change in the intensity, even in short period of time and in a single frequency bin. However, environmental sounds have smaller changes in intensity values, which means the difference in the intensity values between two successive frames t and t+1 is small. For noise like wind, rain, airplane, the change in the intensity is not that much; the variation in the intensity is approximately constant. Therefore, the ACI for these sounds is low. From this hypothesis, ACI might be a good discriminator for rain/non-heavy rain. Therefore, we choose ACI as one of the main features for rain/non-heavy rain classification.

C) *The Background Noise* (BgN) is estimated from the wave envelope using a modification of the method of (Lamel, et al., 1981) as described by (M. Towsey, 2012b) (the value is expressed in amplitude). Note that the term *background noise* has a technical definition. It is the acoustic energy removed using the method of Lamel.

D) *The Spectral Cover* (SC) calculates the fraction of spectrogram cells where the spectral amplitude exceeds a threshold theta=0.015 (M. Towsey, 2012b). The suitability of this threshold was determined by trial and error.

3.5 CLASSIFIERS SELECTION

In this research we have investigated multiple classification algorithms for the environmental sound classification problem. These algorithms are namely: Decision Tree, Support Vector Machines, Naives Bayes and K-Nearest Neighbour.

We also investigated a variety of regression algorithms for the prediction of rain in a long audio recording. These algorithms are namely: M5P which is a Decision Tree for numeric predictions, Multilayer Perceptron, Linear Regression,

decision Table and RepTree. Each algorithm has its advantages and disadvantages depending on the application. In our case we investigated different classification and regression algorithms to find out which techniques work well for our problem. We selected Decision Tree to be our classifier because DT is a fast learner and it gives explicit rule set so we can see which features are important.

3.6 SOFTWARE TOOLS

This research requires WEKA (Waikato Environment for Knowledge Analysis) software which is a collection of machine learning algorithms written in Java for data mining tasks. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. For most of the tests, the explorer mode of WEKA is used.

We used R programming language to implement the algorithms for features extraction. The R language is widely used in statistics, data analysis and data mining.

Seewave package for sound analysis is used widely in this study. Seewave provides functions for analysing, manipulating, displaying, editing and synthesizing time waves (particularly sound). This package processes time analysis (oscillograms and envelopes), spectral content, resonance quality factor, entropy, cross correlation and autocorrelation, zero-crossing, dominant frequency, analytic signal, frequency coherence, 2D and 3D spectrograms and many other analyses.

Chapter 4: **Experiments and Discussion**

Our work is concerned with the classification and prediction of rain in environmental recordings. We present the novel application of a set of features for environmental acoustics classification: acoustic entropy (H), acoustic complexity index (ACI), spectral cover (SC) and background noise (BgN). In order to improve the performance of the rain classification system, we have investigated different classifiers use the Decision Tree classifier to automatically classify the environmental sounds into different classes and we compare its performance with other classifiers. The experimental results show that our system is effective in classifying the environmental sounds with an accuracy rate of 93% for the two class classification (*heavy-rain*/*non-heavy rain*).

The previous Chapter discussed the construction of the different datasets, followed by the feature extraction and feature selection. The target of this Chapter is to introduce six series of experiments.

4.1 EXPERIMENT 1: BINARY CLASSIFICATION

The *heavy rain* events were classified by C4.5 Decision Tree (DT) classifier (J48 in Weka). The Dataset A.1 contained 244 recordings of *heavy rain* and 754 of *non-heavy rain* (*cicadas, birds, koalas, frogs, low activity*, and *light rain*). We performed 10 fold-cross validation on the data. To measure the classification accuracy, we used three measures: precision, recall and accuracy. Precision is defined as $TP/(TP + FP)$, recall as $TP/(TP + FN)$ and accuracy as $(TP + TN)/total\ samples$, where TP, FP, TN, FN are true positive, false positive, true negative, and false negative respectively. The DT classifier was compared with three other classifiers: Naive Bayes, Lazy IBK ($k = 1$), and SMO. The purpose of this experiment is to find the best algorithm and the best feature set or combination of features. We run experiments that use different combinations of features and different classifiers to classify environmental sounds into two classes as shown in Table 3.1 (Dataset A.1).

Table 4.1 provides a summary of the results that we received from each algorithm for the two classes (*heavy rain*/*non-heavy rain*). It can be observed that the average classification accuracy of the Ht+Hf+ACI+BgN+SC features is the best. We noticed that combining only temporal and spectral entropy produces low classification accuracy in differentiating the classes. It is noticeable that combining more than two features increases the accuracy rate. From Table 4.1, we can see also that DT and lazy IBK perform better than the other algorithms. Despite similar performance between IBK and DT, we conclude that a DT is the best classifier because the classification rules are easily extracted and repurposed. The Ht+Hf+ACI+BgN+SC is the best feature set in our experiment. The classification accuracy achieved is 93%.

Table 4.1 Total accuracy rate of Dataset A.1 using different types of classifiers and features.

Feature Type	**Accuracy Rate (%)**				**Average over 4 classifiers**
	NB	**IBK**	**SMO**	**DT**	
Hf	77	82	76	88	80.75
Ht	76	72	76	76	75
ACI	89	81	88	89	86.75
BgN	77	71	76	78	75.5
SC	83	76	84	84	81.75
Ht+Hf	89	84	76	88	85
ACI+BgN	89	89	90	90	89.5
Hf+Ht+ACI	**91**	90	91	92	91
Hf+Ht+BgN	77	87	78	91	83.25
Hf+Ht+SC	84	85	85	89	85.75
ACI+BgN+SC	**91**	91	**92**	92	91.5
Ht+Hf +ACI+BgN	90	92	91	92	91.25
Ht+Hf +ACI+BgN+SC	**91**	**93**	**92**	**93**	**92.25**

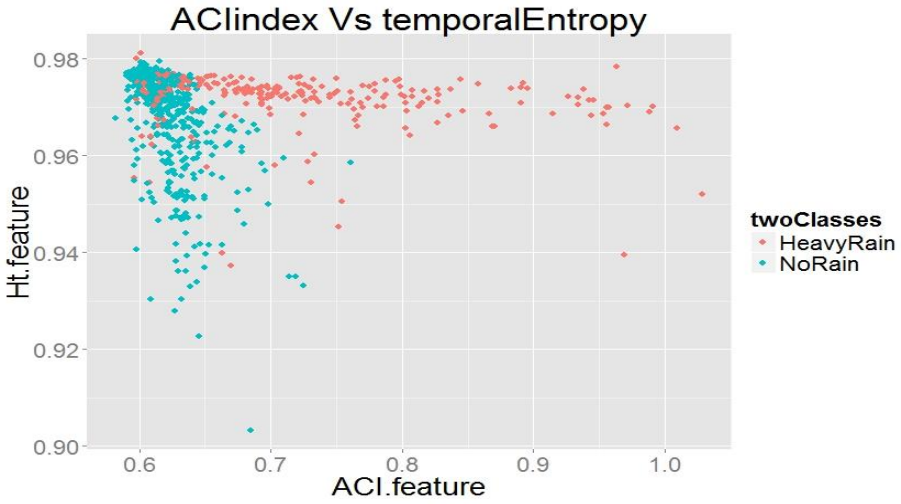


Figure 4.1 The relationship between two features in classifying the Dataset A.1 (two-class-problem) with a Decision Tree classifier.

Figure 4.1 shows the strong relationship between two features namely: acoustic complexity index (ACI) and temporal entropy (Ht) in distinguishing the two class heavy-rain/non-heavy rain (binary classification). It is apparent that a linear function can split the majority of instances into two classes.

For evaluation of our binary classification system, we compared spectral feature set (ACI, Hf, BgN and SC) with the most feature used in audio classification, which is MFCCs feature.

4.1.1. Experiment A: Exploration of spectral features for binary classification (heavy-rain/non-heavy rain)

In Experiment 1 the features (ACI, Ht, Hf, BgN and SC) were calculated for each frequency bin and the average was taken over all the 256 frequency bins. In this experiment we calculate the average of each 16 frequency Bins for the spectral features namely: Hf, ACI, BgN, and SC; which means each feature will have 16 values. The purpose of this experiment is to find the best classifier and the best feature set. We run experiments that use different combinations of features and different classifiers to classify environmental sounds into two classes as shown in Table 4.2.

Table 4.2 Total accuracy rate of Dataset A.1 using different types of classifiers and spectral features (Experiment A).

Feature Type	**Accuracy Rate (%)**				**Average over 4 classifiers**
	NB	**IBK(k=1)**	**SMO**	**DT**	
Hf feature set	92	97	92	93	93.5
ACI feature set	91	95	92	93	92.75
BgN feature set	85	95	89	92	90.25
SC feature set	84	93	86	91	88.5
Hf+ACI	92	97	93	94	94
Hf+BgN	93	97	95	94	94.75
Hf+SC	91	96	94	94	93.75
(Hf+ACI+BgN) feature set=>F1	**92**	**98**	95	**95**	**95**
(Hf+ACI +SC) feature set	91	97	94	93	93.75
(Hf+BgN+SC) feature set	92	97	96	94	94.75
F1+SC feature set	91	97	96	94	94.5
(Hf+ACI+BgN+SC)feature set=>F2	91	97	**96**	94	94.5

From this table we can see that IB1 classifier gives the best accuracy rate for the classification with 98%, and the Hf+ACI+BgN is the best scoring feature set with

average accuracy rate of 95%. We can conclude that adding SC feature set to feature set F1 didn't change the accuracy rate of the classification (F2 feature set).

We also noticed an important difference between the results shown in Table 4.1 and Table 4.2: that cutting down the spectra from 256 to 16 increases the accuracy of the classification.

4.1.2. Experiment B: Exploration of the combination of spectral features with MFCC features for binary classification (heavy-rain/non-heavy rain)

In this experiment we calculate the average of each 16 frequency Bins for the spectral features namely: Hf, ACI, BgN, and SC. Also we calculated 12 coefficients for MFFCs. The purpose of this experiment is to combine spectral features with MFFCs features to find the best classifier and the best feature set. We run experiments that use different combinations of features and different classifiers to classify environmental sounds into two classes. The results of this experiment are summarised in Table 4.3.

Table 4.3 Total accuracy rate of Dataset A.1 using different types of classifiers, different spectral features, and MFCCs (Experiment B).

Feature Type	**Accuracy Rate (%)**				**Average over 4 classifiers**
	NB	**IBK(k=1)**	**SMO**	**DT**	
MFCCs feature set	90	98	93	95	94
(MFCCs+Hf) feature set	**94**	99	97	95	**96.25**
(MFCCs+ACI) feature set	93	99	96	95	95.75
(MFCCs+BgN) feature set	89	98	94	94	93.75
(MFCCs+SC) feature set	88	98	94	94	93.5
MFCCs+Hf+ACI	93	**99**	97	95	**96**
MFCCs+Hf+BgN	93	98	98	94	95.75
MFCCs+ACI+BgN	93	98	96	95	95.5
MFCCs+ACI+SC	91	99	97	**96**	95.75
MFCCs+BgN+SC	88	98	96	95	94.25
MFCCs+Hf+ACI+BgN	93	98	97	96	96
MFCCs+Hf+ACI+BgN+SC	92	98	**98**	95	95.75

Table 4.3 shows the classification rates for the heavy-rain/non-heavy rain classes using the combination of spectral features (Hf, ACI, BgN and SC) with MFCCs, and obtained by the classifiers: NB, SMO, IBk and DT. Clearly all the classifiers generate good results, we noticed that IBk classifier presents the highest accuracy rate with 99% and MFCCs+Hf+ACI is the best scoring feature set with average accuracy rate of 96%. We can notice that BgN and SC feature set didn't change the

accuracy rate when combined with MFCCs+Hf+ACI feature set. The use of the feature set (Hf+ACI) and MFCCs are complementary.

4.2 EXPERIMENT 2: MULTI-CLASS CLASSIFICATION

The purpose of this experiment is to determine whether the feature set ACI+Ht+ Hf+BgN+SC can be used to distinguish other common sounds in environmental recordings (such as *cicadas*, *animal sounds* in general and *heavy rain*). Note that the features were calculated for each frequency bin and the average was taken over all the 256 frequency bins. To further understand the classification performance, we show results in the form of a confusion matrix, which allows us to observe the degree of confusion among different classes.

Table 4.4 Confusion matrix for Dataset A.2 (3 class-problem) using Decision Tree classifier.

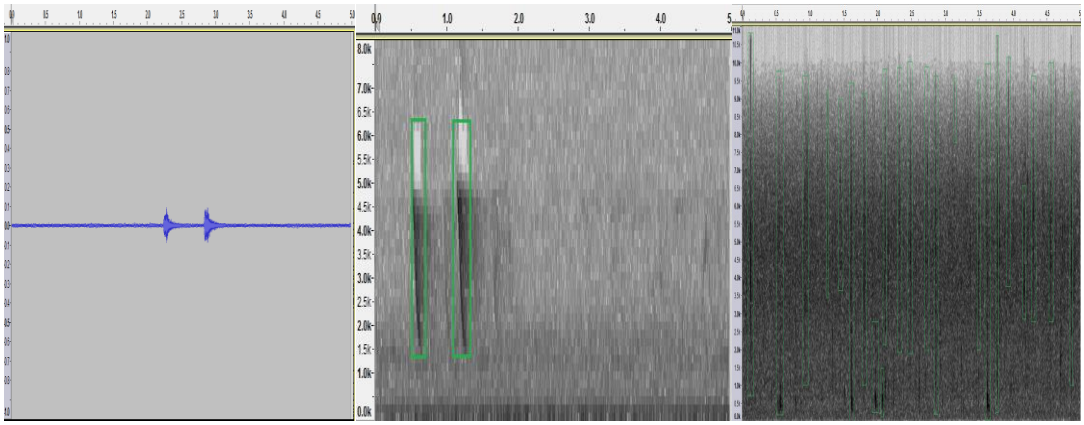
		Predicted as		
		Animal sounds	**Heavy rain**	**Cicadas**
Actual class	**Animal sounds**	**495**	22	**48**
	Heavy rain	**43**	**194**	7
	Cicadas	**40**	5	**144**

Table 4.5 Confusion matrix for Dataset A.3 (4 class-problem) using Decision Tree classifier.

		Predicted as			
		Animal sounds	**others**	**Heavy rain**	**Cicadas**
Actual class	**Animal sounds**	**450**	7	**13**	30
	others	16	**41**	0	4
	Heavy rain	**48**	0	**194**	2
	Cicadas	23	9	5	**156**

The confusion matrix given in Table 4.4 and Table 4.5 are constructed by applying the DT classifier to the Dataset A.2 (3 class-problem) and Dataset A.3 (4 class-problem) respectively; and displaying the number of correctly/incorrectly classified instances.

From Table 4.4 and Table 4.5, we can notice that the major misclassified instances are between “heavy rain”, “birds” and “cicadas”. The reasons lie in the fact that some bird calls have similar call structure as heavy rain (Figure 4.2) and some of the bird calls have similar call structure as cicadas (Figure 4.3).



Waveform of bird call | Spectrogram of bird call | Spectrogram of heavy rain

Figure 4.2 Five seconds audio segment labeled as “bird” but classified as “heavy rain”.

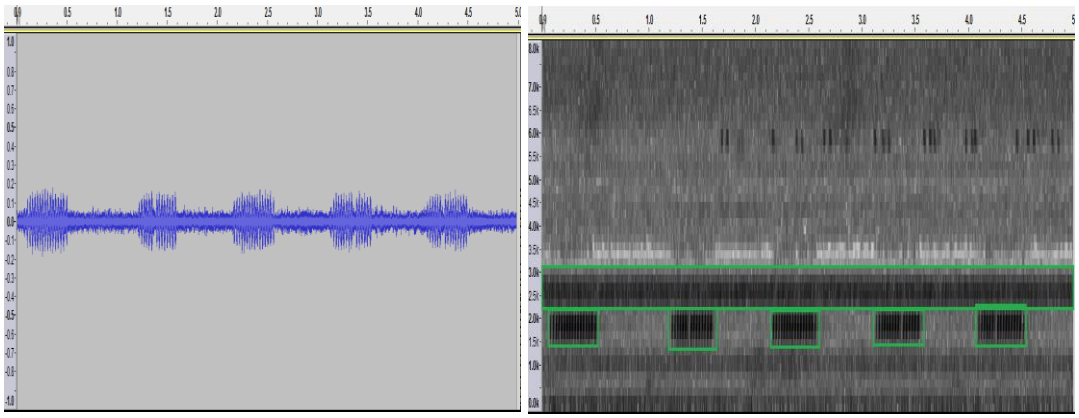


Figure 4.3 Five seconds audio segment labeled as “cicada” but classified as “bird”.

The following figure shows the plot of four different classes using Decision Tree classifier. It is clear that our feature set is capable between the different classes.

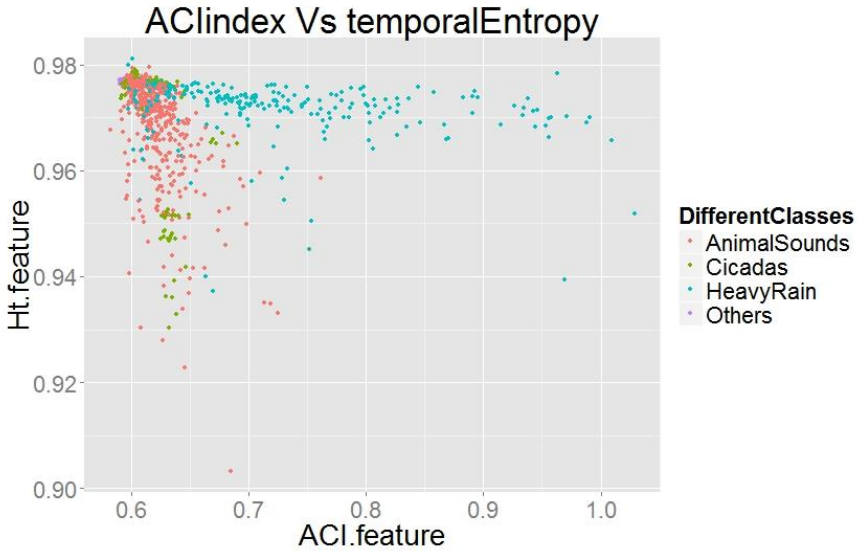


Figure 4.4 The relationship between two features in classifying the Dataset A.3 (multi-class-problem) with a Decision Tree classifier.

From this Figure 4.4, we can clearly see clearly that heavy rain class is easily differentiable from other classes.

4.3 EXPERIMENT 3: COMBINATION OF MULTIPLE CLASSIFIERS FOR DATASET A.3 (FOUR CLASS PROBLEM)

The purpose of this experiment is to combine multiple classifiers instead of one single classifier (as in Experiment 1) and find the best combination. The advantage of using a combination of multiple classifiers is that instead of one single classifier algorithm's power we used three/four classification algorithm's power; so the model induced by combining these multiple classifiers will be more reliable/correct or more sophisticated to identify/classify instances from the cross-fold validation set. We used the "vote classifier" to combine different classifiers.

Table 4.6 Total accuracy rate obtained from the combination of multiple classifiers for Dataset A.3.

	H_t+H_f	H_t+H_f +ACI	H_t+H_f+BgN	ACI+BgN+ SC	H_t+H_f+ACI+ BgN	H_t+H_f+ACI+ BgN+ SC
NB+DT	**75.85**	75.05	76.96	76.36	80.36	83.47
IB1+DT	69.83	75.55	76.45	75.85	83.67	85.27
SMO+DT	73.94	**80.26**	78.46	77.36	83.47	84.77
SMO+ IB1+DT	72.34	79.36	77.86	78.96	84.27	86.38
SMO+DT+ IB1+NB	73.84	79.76	**79.06**	**79.56**	**85.87**	**86.68**

From this table we notice that the combination of the different classifiers: SMO+DT+IB1+NB and the combination of different features: Ht+Hf+ACI+BgN+SC give the best accuracy rate of 86.68%.

4.4 EXPERIMENT 4: DETECTION OF RAIN IN THE 24-HOUR LONG AUDIO RECORDING

The aim of this experiment is to show the ability of regression techniques in predicting rain in the 24-hour long recording. We first split the 24-hour recording into one minute audio which yields to 1440 minutes, we further cut each one minute into five seconds, in total $(1440 \times 12 = 17280)$ of five seconds segments. We extracted five features (the same features used in the Experiment 1) from each five seconds of audio, and then we averaged the feature values to produce five minutes blocks. This is done so the weather data, which has a five-minute resolution (287 instances); can be directly used as ground truth data. We have explored a variety of prediction techniques in Weka, specifically: M5P, linear regression, RepTree, Multi-layer-perecptron, and Decision table.

Weka provides a variety of error measures, which are based on the differences between the actual and estimated values. Three measures were selected for comparison: *correlation coefficients (R^2), mean absolute error* (MAE), and *root mean square error* (RMSE), which can be computed as follow:

MAE and RMSE are regularly used as standard statistical metric to measure the model performance, lower values result in better predictive models.

- The *correlation coefficient* measures the degree of correlation between the actual and estimated values. Table 3 summarizes three different statistical measures (MAE, RMSE and coefficient correlation) for the different algorithms using *10 fold cross-validation*.

M5P proved the best results in our case (Table 4.7) because of the nature of the problem considered as well as the type of data we are using. M5P is a Decision Tree for numeric prediction that stores a linear regression at each leaf to predict the class value of instances that reach that leaf. When the class attributes is numeric, M5P is found to be a good technique to handle such situations. In our case, the class attribute represents the amount of rain in mm over five minute periods; therefore, M5P is more suited for this problem than other techniques.

Table 4.7 Correlation coefficients between actual and predicted rain, MAE and RMSE.

Algorithms	**Correlation coefficients**	**MAE**	**RMSE**
M5P	**0.78**	0.07	0.14
LR	0.75	0.08	0.15
RepTree	0.68	0.08	0.17
MLP	0.67	0.11	0.19
DTB	0.69	0.07	0.17

The M5P tree model developed with *10 fold* cross-validation was realized to be the best model that predicted rain in the 24h-recording with RMSE of 0.14, and a correlation coefficient of the measured and predicted rain of 0.78.

We present an example of the predictor we obtained for *Rain* using M5P; it can be seen that background noise was used by M5P as the main feature.

```
M5 pruned model tree:
(using smoothed linear models)

BgN.mean.vect <= 0.096 : LM1 (114/0%)
BgN.mean.vect >  0.096 :
|   BgN.mean.vect <= 0.236 : LM2 (112/36.832%)
|   BgN.mean.vect >  0.236 : LM3 (61/111.638%)

LM num: 1
Rain =
      -0.2039 * Ht.mean.vect
      + 0.3176 * Hf.mean.vect
      + 0.3064 * aci.mean.vect
      + 0.1899 * BgN.mean.vect
      - 53.8624 * cover.mean.vect
      - 0.2406
LM num: 2
Rain =
      -0.1399 * Ht.mean.vect
      + 2.1356 * Hf.mean.vect
      + 2.7744 * aci.mean.vect
      + 0.3232 * BgN.mean.vect
      - 87.622 * cover.mean.vect
      - 3.0552
LM num: 3
Rain =
      -0.1399 * Ht.mean.vect
      + 13.1186 * Hf.mean.vect
      + 13.7958 * aci.mean.vect
      + 2.0925 * BgN.mean.vect
      - 121.6197 * cover.mean.vect
      - 18.6741
Number of Rules: 3
```

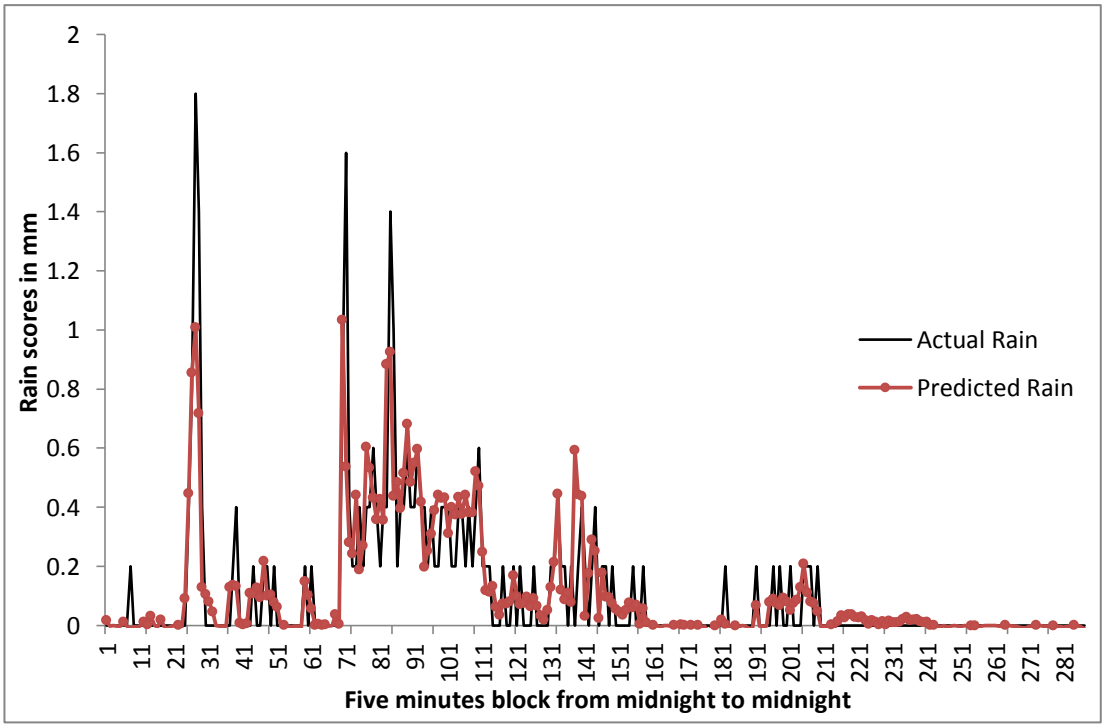


Figure 4.5 An example for rain prediction using M5P.

Figure 4.3 illustrates the power of the M5P algorithm in estimating rain amount in a 24h-long recording. The red series represents the M5P estimates while the black series is the ground truth (actual rain amount from weather station data). It can be seen that the M5P's predictions correspond well with the ground truth data.

We conclude that M5P is the best algorithm for this experiment, whereas MLP algorithm presents the poorest result with correlation coefficient of the measured and predicted rain of 0.67.

Chapter 5: **Conclusions and future work**

This thesis has described the application of a new set of features for environmental sounds classification. In order to get a better accuracy rate, we explored different combination of features and applied different machine learning algorithms to the data.

This chapter summarises the work presented, discusses the significance of the research outcomes and illustrates possible directions for future work.

5.1 SUMMARY OF CONTRIBUTIONS

This thesis has made the following contributions to the environmental sounds classification.

We have presented an environmental sound classification system using five features and the Decision Tree classifier. Our comparison experiments show that the method presented is promising. The combination of five features provides better classification performance than using two features.

Another aim of this study was to show the ability of regression techniques in predicting *rain* in 24-hour long audio recordings collected by sensors in the field. The results showed that M5P has better predictability than the other techniques. Such a prediction tool could prove useful when ecologists are interested in analysing acoustic audio data, especially when the target fauna – such as many Anuran species have a vocalising relationship with rain events.

- The major aim of this work was to classify environmental sounds into different type of classes using recordings directly collected from the field.
- In this work, we have explored different features (Acoustic complexity index, spectral entropy, temporal entropy, background noise and spectral cover) used generally for environmental monitoring but not previously evaluated on rain detection in audio recordings; and propose the application of these features to discriminate different classes of environmental sounds.

- We have used a variety of machine learning algorithms such as classification algorithms (e.g., J48 Decision Tree, Support vector machines, Naive Bayes, and Lazy classifier), regression algorithms (e.g., Linear Regression, M5P, RepTree, MultiLayer Perceptron, and Decision table) to predict rain. The effectiveness and accuracy of these algorithms in predicting rain was analysed.
- We have used the same feature set (Ht, Hf, ACI, BgN and SC) as in the binary classification and explored different regression algorithms to predict *rain* in long audio recordings (24h long).
- Our features work well in differentiating *heavy rain* from *non-heavy rain* . The accuracy rate achieved in the two class-problems was 93%. Even more, our feature set is good enough to predict *rain* in long audio recordings.

5.2 LIMITATIONS

Although our classification system including the feature set, the different machine learning techniques (classification and regression) show promising detection ability of *rain* in audio recordings. However, there are some limitations:

- In the classification part, we have used audio recordings collected by sensors, these recordings are from different days, different times of day and from different sites at (SERF) in Queensland. However, the quality of acoustic recordings might be different if they were collected using different type of sensors and collected under different climate conditions. The audio recordings format is MP3 format which is designed to reproduce sound accurately for the human ear and has been found suitable for identifying bird calls (Rempel et al 2005). However we have not investigated the effect that MP3 compression might or might not have on the detection of rain in acoustic recordings.
- The present study has explored a variety of classification and prediction algorithm to detect and classify the content of audio recordings. Although the experiments showed that these classification techniques are good in classifying the content of acoustic recordings based on the

extracted features, some of the used features are suitable to represent a particular class and not for others.

5.3 FUTURE WORK

This research aims to investigate classification techniques that predict rain in large datasets of audio collected by acoustic sensors.

The research can be extended to overcome the identified limitations. Several interesting directions seem promising for improving the current techniques.
First, future research can develop robust noise removal algorithms to enhance the accuracy of the classification/regression techniques. A noise removal algorithm plays an important role in the preprocessing phase of audio recordings.

Second, our technique has been tested on a small dataset, in future work this technique could be applied to much larger datasets weeks, months and years in order to predict rain in audio recordings.

APPENDIX

Using WEKA

WEKA is a collection of machine learning algorithms for Data Mining tasks. It contains tools for data preprocessing, classification, regression, clustering, association rules, and visualization. WEKA has four different modes to work on:

- Simple CLI: it provides a simple command-line interface that allows direct execution of WEKA commands.
- Explorer: it is an environment for exploring data with WEKA.
- Experimenter: it is an environment for performing experiments and conduction of statistical tests between learning schemes.
- Knowledge Flow: it presents a "data-flow" inspired interface to WEKA. The user can select WEKA components from tool bar, place them on a layout canvas and connect them together in order to form a "knowledge flow" for processing and analyzing data.

WEKA requires the data in the train/test file to be in ARFF format. The general format of an ARFF file is given in Table B1. The string @relation is used to mention the name of the dataset, @attribute is used to define the attributes name and type and @data is used to indicate the start of the data, which is in a comma-separated form.

Following are the *classifier_path* for the machine learning algorithms that were used in this thesis along with their default options (*classifier_options*)

Table I: Format of an ARFF file.

```
@relation 2ClassProblem


@attribute temporal.entropy numeric

@attribute spectral.entropy numeric

@attribute ACIndex numeric

@attribute BgNAverage numeric

@attribute CoverAv numeric

@attribute twoClasses (Onyari & Ilunga, 2010)


@data

0.959244,0.908879,0.598785,0.008447,0.147272,NoRain

0.966025,0.910884,0.617509,0.008517,0.054524,NoRain

0.977061,0.900393,0.598504,0.009731,0.031368,NoRain...
```

Table II. Classifier Algorithms in WEKA and their commands

Classifier Algorithms in WEKA			
	Name	**Function**	**Weka Command**
Bayes	NaiveBayes	Standard probabilistic Naïve Bayes classier	`Weka.classifiers.bayes.NaiveBayes`
Rules	DecisionTable	Builds a simple decision table majority classifier	`Weka.classifiers.rules.DecisionTable –X 1 –S "weka.attributeSelection.BestFirst –D 1 –N5"`
Functions	SMO	Sequential minimal optimization algorithm for support vector classification	`Weka.classifiers.functions.SMO –C 1.0 –L 0.001 –P 1.0E-12 –N 0 –V -1 –W 1 –K "weka.classifiers.functions.supportVector.Polykernel –E 1.0 –C 2500027"`
	LinearRegression	Standard multiple linear regression	`Weka.classifiers.functions.LinearRegression -S 0 -R 1.0E-8`
	MultilayerPerceptron	Backpropagation neural network	`Weka.classifiers.functions.MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a`
Lazy	IBk	k-nearest-neighbours classifier	`Weka.classifiers.lazy.IBk –K 1 W 0 –A "weka.core.neighboursearch.LinearNNSearch –A\"weka.core.EuclideanDistance –R first-last\""`
Trees	J48	C4.5 Decision Tree learner	`Weka.classifiers.trees.J48 -C 0.25 –M 2`
	M5P	M5' model tree learner	Weka.classifiers.trees.`M5P –M 4.0`
	RepTree	Fast tree learner that uses reduced-error pruning	Weka.classifiers.trees`–M2 –V 0.001 –N 3 –S 1 –L –1 –I 0.0`

Acevedo, M. A., Corrada-Bravo, C. J., Corrada-Bravo, H., Villanueva-Rivera, L. J., & Aide, T. M. (2009). Automated classification of bird and amphibian calls using machine learning: A comparison of methods. *Ecological Informatics, 4*(4), 206-214.

Barkana, B. D., & Uzkent, B. (2011). Environmental noise classifier using a new set of feature parameters based on pitch range. *Applied Acoustics, 72*(11), 841-848.

Bishop, C. M. (1995). Neural networks for pattern recognition.

Brandes, T. S., Naskrecki, P., & Figueroa, H. K. (2006). Using image processing to detect and classify narrow-band cricket and frog calls. *The Journal of the Acoustical Society of America, 120*, 2950.

Bregman, A. S. (1994). *Auditory scene analysis: The perceptual organization of sound*: MIT press.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*: CRC press.

Briggs, F., Raich, R., & Fern, X. Z. (2009). *Audio classification of bird species: A statistical manifold approach*.

Brigham, E., & Morrow, R. E. (1967). The fast Fourier transform. *Spectrum, IEEE, 4*(12), 63-70. doi: 10.1109/MSPEC.1967.5217220

Cheng, J., Sun, Y., & Ji, L. (2010). A call-independent and automatic acoustic system for the individual recognition of animals: a novel model using four passerines. *Pattern Recognition, 43*(11), 3846-3852.

Chu, S., Narayanan, S., & Jay Kuo, C. C. (2008). *Environmental sound recognition using MP-based features.* Paper presented at the Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on.

Chu, S., Narayanan, S., & Kuo, C. C. J. (2006). *Content analysis for acoustic environment classification in mobile robots.* Paper presented at the AAAI Fall Symposium, Aurally Informed Performance: Integrating Machine Listening and Auditory Presentation in Robotic Systems.

Chu, S., Narayanan, S., & Kuo, C. C. J. (2009). Environmental sound recognition with time–frequency audio features. *Audio, Speech, and Language Processing, IEEE Transactions on, 17*(6), 1142-1158.

Cohen, L. (1995). *Time-frequency analysis* (Vol. 778): Prentice Hall PTR Englewood Cliffs, New Jersey.

Colonna, J. G., Ribas, A. D., dos Santos, E. M., & Nakamura, E. F. (2012). *Feature subset selection for automatically classifying anuran calls using sensor networks.* Paper presented at the Neural Networks (IJCNN), The 2012 International Joint Conference on.

Connell, S. D., & Jain, A. K. (2001). Template-based online character recognition. *Pattern Recognition, 34*(1), 1-14. doi: http://dx.doi.org/10.1016/S0031-3203(99)00197-1

Cortez, P., & Morais, A. d. J. R. (2007). A data mining approach to predict forest fires using meteorological data.

Cowling, M., & Sitte, R. (2003). Comparison of techniques for environmental sound recognition. *Pattern Recognition Letters, 24*(15), 2895-2907.

Cufoglu, A., Lohi, M., & Madani, K. (2008). *Classification accuracy performance of Naive Bayesian (NB), Bayesian Networks (BN), Lazy Learning of Bayesian Rules (LBR) and Instance-Based Learner (IB1)-comparative study*. Paper presented at the Computer Engineering & Systems, 2008. ICCES 2008. International Conference on.

Duan, S., Towsey, M., Zhang, J., Truskinger, A., Wimmer, J., & Roe, P. (2011). *Acoustic component detection for automatic species recognition in environmental monitoring*. Paper presented at the Seventh International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP).

Eronen, A. J., Peltonen, V. T., Tuomi, J. T., Klapuri, A. P., Fagerlund, S., Sorsa, T., . . . Huopaniemi, J. (2006). Audio-based context recognition. *Audio, Speech, and Language Processing, IEEE Transactions on, 14*(1), 321-329.

Fagerlund, S. (2007). Bird species recognition using support vector machines. *EURASIP Journal on Advances in Signal Processing, 2007*.

Giret, N., Roy, P., & Albert, A. (2011). Finding good acoustic features for parrot vocalizations: The feature generation approach. [Article]. *Journal of the Acoustical Society of America, 129*(2), 1089-1099. doi: 10.1121/1.3531953

Good, I. J. (1965). *The estimation of probabilities: An essay on modern Bayesian methods* (Vol. 258): MIT press Cambridge, MA.

Griffin, D., & Lim, J. S. (1984). Signal estimation from modified short-time Fourier transform. *Acoustics, Speech and Signal Processing, IEEE Transactions on, 32*(2), 236-243. doi: 10.1109/TASSP.1984.1164317

Guo, G., & Li, S. Z. (2003). Content-based audio classification and retrieval by support vector machines. *Neural Networks, IEEE Transactions on, 14*(1), 209-215.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter, 11*(1), 10-18.

Huang, C. J., Yang, Y. J., Yang, D. X., & Chen, Y. J. (2009). Frog classification using machine learning techniques. *Expert Systems with Applications, 36*(2), 3737-3743.

Hubbard, B. B. (1996). The world according to wavelets: the story of a mathematical technique in the making.

Karbasi, M., Ahadi, S. M., & Bahmanian, M. (2011). *Environmental sound classification using spectral dynamic features*.

Ke, D., Heng Tao, S., Kai, X., & Xuemin, L. (2006, 03-07 April 2006). *Surface k-NN Query Processing.* Paper presented at the Proceedings of the 22nd International Conference on Data Engineering, 2006. ICDE '06'. .

Kohavi, R. (1995). *A study of cross-validation and bootstrap for accuracy estimation and model selection.* Paper presented at the IJCAI.

Kohavi, R. (1996). *Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid.* Paper presented at the KDD.

Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques.

Lamel, L., Rabiner, L., Rosenberg, A., & Wilpon, J. (1981). An improved endpoint detector for isolated word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing, 29*(4), 777-785.

Langley, P., & Sage, S. (1994). *Induction of selective Bayesian classifiers*. Paper presented at the Proceedings of the Tenth international conference on Uncertainty in artificial intelligence, Seattle, WA.

Li, Y., & Li, Y. (2010). *Eco-environmental sound classification based on matching pursuit and support vector Machine*. Paper presented at the 2nd international conference on Information Engineering and Computer Science (ICIECS).

Ma, L., Milner, B., & Smith, D. (2006). Acoustic environment classification. *ACM Transactions on Speech and Language Processing (TSLP), 3*(2), 1-22.

Markel, J. E., & Gray, A. H. (1982). *Linear prediction of speech*: Springer-Verlag New York, Inc.

Mason, R., Roe, P., Towsey, M., Jinglan, Z., Gibson, J., & Gage, S. (2008, 7-12 Dec. 2008). *Towards an Acoustic Environmental Observatory.* Paper presented at the eScience, 2008. eScience '08. IEEE Fourth International Conference on.

Mitrovic, x, D., Zeppelzauer, M., & Eidenberger, H. (2009, 28-30 Sept. 2009). *On feature selection in environmental sound recognition.* Paper presented at the ELMAR, 2009. ELMAR '09. International Symposium.

Mitrović, D., Zeppelzauer, M., & Breiteneder, C. (2010). Chapter 3 - Features for Content-Based Audio Retrieval. In V. Z. Marvin (Ed.), *Advances in Computers* (Vol. Volume 78, pp. 71-150): Elsevier.

Mporas, I., Ganchev, T., Kocsis, O., Fakotakis, N., Jahn, O., Riede, K., & Schuchmann, K.-L. (2012). *Automated Acoustic Classification of Bird Species from Real-Field Recordings.* Paper presented at the Tools with Artificial Intelligence (ICTAI), 2012 IEEE 24th International Conference on.

Olson, D. L., & Delen, D. (2008). *Advanced data mining techniques*: Springer.

Onyari, E., & Ilunga, F. (2010). *Application of MLP neural network and M5P model tree in predicting streamflow: A case study of Luvuvhu catchment, South Africa.* Paper presented at the International Conference on Information and Multimedia Technology (ICMT 2010), Hong Kong, China.

Parra Jr, J., & Kiekintveld, C. (2013). *Initial Exploration of Machine Learning to Predict Customer Demand in an Energy Market Simulation.* Paper presented at the AAAI Workshop: Trading Agent Design and Analysis.

Pfeiffer, S., & Vincent, T. (2001). Formalisation of MPEG-1 compressed domain audio features. *CSIRO Mathematical and Information Sciences, Australia, Tech. Rep, 1*, 196.

Picone, J. W. (1993). Signal modeling techniques in speech recognition. *Proceedings of the IEEE, 81*(9), 1215-1247.

Pieretti, N., Farina, A., & Morri, D. (2011). A new methodology to infer the singing activity of an avian community: the Acoustic Complexity Index (ACI). *Ecological Indicators, 11*(3), 868-873.

Pohjalainen, J. (2007). *Methods of automatic audio content classification.* Citeseer.

Quinlan, J. R. (1992). *Learning with continuous classes.* Paper presented at the Proceedings of the 5th Australian joint Conference on Artificial Intelligence.

Rabiner, L. R., & Juang, B.-H. (1993). *Fundamentals of speech recognition* (Vol. 14): PTR Prentice Hall Englewood Cliffs.

Rocchesso, D. (2003). *Introduction to sound processing*: Mondo estremo.

Roederer, J. G. (2008). *The physics and psychophysics of music: an introduction*: Springer Science & Business Media.

Rokach, L., & Maimon, O. (2005). Top-down induction of decision trees classifiers-a survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 35*(4), 476-487.

Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *Systems, Man and Cybernetics, IEEE Transactions on, 21*(3), 660-674.

Saunders, J. (1996). *Real-time discrimination of broadcast speech/music.* Paper presented at the Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on.

Somervuo, P., Harma, A., & Fagerlund, S. (2006). Parametric representations of bird sounds for automatic species recognition. *Audio, Speech, and Language Processing, IEEE Transactions on, 14*(6), 2252-2263.

Srinivasan, S., Petkovic, D., & Ponceleon, D. (1999). *Towards robust features for classifying audio in the CueVideo system.* Paper presented at the Proceedings of the seventh ACM international conference on Multimedia (Part 1).

Sueur, J., Pavoine, S., Hamerlynck, O., & Duvail, S. (2008). Rapid Acoustic Survey for Biodiversity Appraisal. *PLoS ONE, 3*(12), e4065. doi: 10.1371/journal.pone.0004065

Temko, A., & Nadeu, C. (2006). Classification of acoustic events using SVM-based clustering schemes. *Pattern Recognition, 39*(4), 682-694.

Towsey, M. (2012b). Technical Report: The calculation of acoustic indices to characterize acoustic recordings of the environment. Brisbane: Queensland University of Technology. Australia.

Towsey, M. (2013). Noise removal from wave-forms and spectrograms derived from natural recordings of the environment. Brisbane: Queensland University of Technology. Australia

Towsey, M., Parsons, S., & Sueur, J. (2014). Ecology and acoustics at a large scale. *Ecological Informatics, 21*, 1-3.

Towsey, M., & Planitz, B. (2010). Acoustic analysis of the natural environment. [Technical report].

Towsey, M., Planitz, B., Nantes, A., Wimmer, J., & Roe, P. (2012). A toolbox for animal call recognition. *Bioacoustics, 21*(2), 107-125. doi: 10.1080/09524622.2011.648753

Towsey, M., Zhang, L., Cottman-Fields, M., Wimmer, J., Zhang, J., & Roe, P. (2014). Visualization of Long-duration Acoustic Recordings of the Environment. *Procedia Computer Science, 29*, 703-712.

Towsey, M. W., & Planitz, B. (2011). Technical report: acoustic analysis of the natural environment.

Uzkent, B., Barkana, B. D., & Cevikalp, H. (2012). NON-SPEECH ENVIRONMENTAL SOUND CLASSIFICATION USING SVMS WITH A NEW SET OF FEATURES. *International Journal of Innovative Computing, Information and Control ICIC International*.

Vaca-Castaño, G., & Rodriguez, D. (2010). *Using syllabic Mel cepstrum features and k-nearest neighbors to identify anurans and birds species.* Paper presented at the Signal Processing Systems (SIPS), 2010 IEEE Workshop on.

Vapnik, V. (1999). The nature of statistical learning theory (Information Science and Statistics).

Vavrek, J., Cizmar, A., & Juhar, J. (2012, 12-14 Sept. 2012). *SVM binary decision tree architecture for multi-class audio classification.* Paper presented at the ELMAR, 2012 Proceedings.

Wang, Y., & Witten, I. H. (1997). *Inducing model trees for continuous classes.* Paper presented at the Proceedings of the Ninth European Conference on Machine Learning.

Wimmer, J., Towsey, M., Planitz, B., Roe, P., & Williamson, I. (2010, 7-10 Dec. 2010). *Scaling Acoustic Data Analysis through Collaboration and Automation.* Paper presented at the e-Science (e-Science), 2010 IEEE Sixth International Conference on.

Witten, I. H., Frank, E., & Hall, M. A. (2005). *Data Mining: Practical machine learning tools and techniques*: Morgan Kaufmann.

Wolff, D. (2008). Detecting Bird Songs via Periodic Structures: A Robust Pattern Recognition Approach to Unsupervised Animal Monitoring. *To be found at< http://www-mmdb. iai. uni-bonn. de/download/Diplomarbeiten/Diplomarbeit_Daniel_Wolff. pdf>[quoted 01.06. 2011]*.

Ye, L., Tong, W., Huijuan, C., & Kun, T. (2006). *Voice activity detection in non-stationary noise.* Paper presented at the Computational Engineering in Systems Applications, IMACS Multiconference on.

Zhuang, X., Zhou, X., Hasegawa-Johnson, M. A., & Huang, T. S. (2010). Real-world acoustic event detection. *Pattern Recognition Letters*, *31*(12), 1543-1551.