Abnormal Bat Echolocation Detection

Yang Peng[a,d], Bety Rostandy[a,d], Hadi Soufi[b,d], Kevin Keomalaythong[b,d], Thien Le[b,d], Han Li[c,e], Matina Kalcounis-Rueppell[c], Somya Mohanty[b,e]

[a] *Department of Mathematics and Statistics*
[b] *Department of Computer Science*
[c] *Department of Biology*
[d] *This author contributed equally.*
[e] *This author supervised jointly.*

Abstract

Bats have been known to echolocate for years. Recent research has showed that bats vocalize for a variety of reasons. Scientists have taken an interest in understanding and classifying the purpose of these calls, such as for social interaction, foraging, and feeding. Until now, however, the best classification technique for such purpose was by hand. K-Means clustering was used to cluster the extracted calls into different types, and Random Forest classification was used to classify zero-crossing files into normal or abnormal calls by using machine learning techniques. The preliminary results of K-Means clustering at $k = 3$ shows that abnormal calls are majorly clustered together. From the clustering insight, the six model parameters were utilized for Random Forest classification as well on the source-observation set. The pulses are the observations, and the source refers to zero-crossing files containing the observations.

Keywords: bat echolocation, abnormal bat call, machine learning, classification, clustering

1. Introduction

The first bat ultrasonic study published in 1938 was done by Griffin, a graduate student at Harvard, which confirmed the prior suggestion that bats emit ultrasonic frequency to navigate in the dark [1]. Griffin noted that bats only emit the high frequency sound at the beginning of their flight and the sound cease by the time they are preparing to alight on a surface or avoiding an obstacle. He also noted that this high frequency sound was produced by bats when they were disturbed, therefore he concluded that the sound is a representation of "a call note or alarm signal". He also concluded that the calls were a normal activity and not an accidental by-product [2].

The study of bat call, so far, has been done for conservation purpose, such as to study the bats ecology and their biodiversity [3, 4, 5]. Schnitzler et al. first proposed that bat calls, then only known as echolocation calls, change when used in navigation and foraging. Echolocation during navigating refers to the bat finding its spatial orientation within its surrounding, while echolocation during foraging refers to the bat's detection or interception as well as the localization of itself and its prey [6]. In our dataset, we embraced two other call purposes: social and feeding. Social calls are used by bats to communicate while in roosts and when flying, while feeding calls are used when giving food to bat pups. Usually, one audio recording of bat calls contains many sequential navigation calls, which we regard as normal calls. Sometimes there are some social calls, foraging calls or feeding calls mixed in. We are more interested in these abnormal calls than normal calls as they can tell us a lot more about bat behavior.

According to Schnitzler et al., echolocation calls consists of frequency-modulated (FM) or constant frequency (CF) components, or the combination of FM and CF [6]. Further classification of calls based on the intensity and harmonic composition of the signals can be categorized into; (i) no echolocation, (ii) brief, broadband tongue clicks, (iii) narrowband signals dominated by the fundamental harmonics, (iv) narrowband multiharmonic signals, (v) short, broadband calls with a dominant fundamental harmonic, (vi) short, broadband multiharmonic signals, (vii) long duration broadband calls, and (viii) pure constant frequency signals. Identification of the echolocation call is usually species-related, a family of bats may share a similar call [7].

Bat calls identification was done manually by bat researchers; this qualitative method can be time consuming and lead to human-bias [8]. Therefore, a more reproducible quantitative methodology was needed for scientific study. Two common quantitative methodologies for classification are discriminant function analysis and an artificial neural network. In their study, Parsons and Jones showed that an artificial neural network was more accurate than discriminant function analysis [9].

Britzke et al. compared parametric (linear discriminant analysis, quadratic discriminant analysis, flexible discriminant analysis, two versions of mixture discriminant analysis) and non-parametric (k-nearest neighbor classification and a neural network classification) quantitative methods to classify 12 bat species. Their study showed that one of the mixture discriminant analysis based on adaptive regression spine classification and neural network classification had the highest accuracy rates among the methods; where adaptive regression spine classification performed slightly better than neural network classification [8]. A more recent study of echolocation signal used deep learning algorithm of supervised convolutional neural network to classify bat audio datasets [10].

All the methodologies mentioned above had been done mostly to identify bat species. The goal of our work is to identify bat calls that deviates from normal species-based classification. We broadly labeled bat calls or pulses as normal and abnormal in our work. This work is aimed to automate the workflow for bat researchers in determining the files that contain abnormal calls, thus cutting down the time required to process the raw files manually. Throughout this paper, we will use bat calls or pulses interchangeably.

The works presented in this paper includes extraction of meaningful signal from noise, clustering the extracted calls into different types, and classification of zero-crossing files into normal or abnormal calls by using machine learning techniques.

The rest of the paper is organized as follows: Section 2 introduces the data set and statistical analysis; Section 3 gives the methodologies and models we used for noise removal, clustering and classification. Finally We present the conclusion and discussion in Section 4.

2. Data Exploration

2.1. Data Description

The full dataset is made of bat call recordings that were obtained from more than one hundred sites across the state of North Carolina, including UNC Greensboro wetland at Peabody Park. The sample dataset is from the months of June and July for a four-year period; from 2015 to 2018 as summarized in Table 1.

While species-related call metadata was provided by biologists at the UNCG Bat and Mouse Lab, we only consider two levels of labeling for echolocation calls - normal and abnormal. The pie chart below shows that about 98% of the sample data files were labeled as only containing normal calls and 2% as containing a mixture of normal and abnormal calls. A zero-crossing file that contains only normal calls was labeled with bat species. The small number of occurrence and the mixture of abnormal calls within a single file with

Year	Subtotal number of file	Abnormal labeled file
2015	2527	64
2016	2766	32
2017	3124	44
2018	2209	56
Total	10626	196

Table 1: Sample dataset summary

normal calls leads to challenges in data processing that require quantitative and automation methodology.

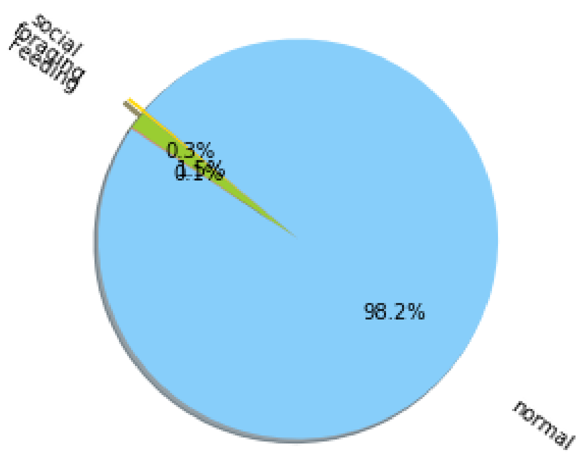

Figure 1: Labeled Sample Dataset Summary

The zero-crossing file format can be viewed with AnalookW software. However, in the process of transforming the raw calls to zero-crossing format we lost important audio data such as information about harmonics and amplitude. This prevents us from identifying species-related calls to a certain degree [5]. So, a zero-crossing file that contains a mixture of normal and abnormal calls was labeled with its related species (if identifiable) and type of abnormal calls: social, foraging or feeding.

2.2. Data Visualization

A segment of one zero-crossing file during 1 to 1.7 seconds is showed in Figure 2. The dominant frequency sweeps of a pulse are easily visible. The normal bat calls have a regular decreasing trend with a slightly smoothing concave up curvature, while abnormal calls have irregular shape such as upward trend, flat trend, U-shape, W-shape and so on. The x-values in the dataset refer to time in *seconds* and the y-values refer bat call frequency in *Hertz*.

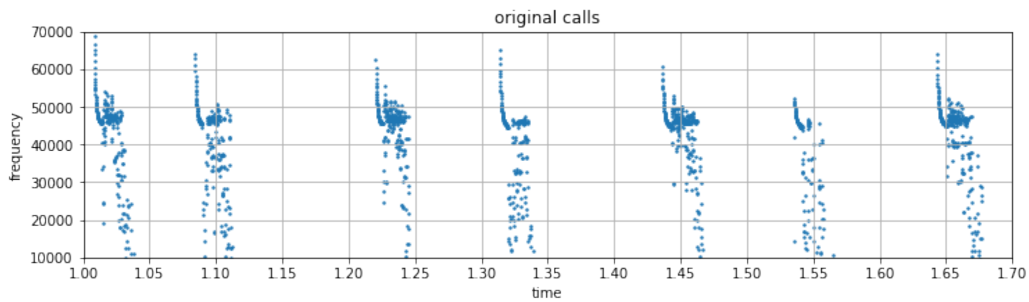

Figure 2: A segment of raw zero-crossing file.

The raw files have a high noise ratio and most calls are not easily categorized. Hence, a noise removal technique was devised to overcome these issues.

2.3. Statistical Analysis

Before data was trained using machine learning algorithm, some statistical analysis were done in order to get a better idea of the characteristic or the nature of dataset we have.

A probability modelling of hypergeometric case was performed to detect what is the probability of getting a file containing abnormal calls in a given processed folder. A simulation study was done based on the observed probability of success (θ) by dividing number of abnormal labeled file (N) with total number of file each year (M). The size of iteration is based on total folder in each year, and average number of files (n) is calculated from total number of file divided by total folder in each year. Since, n is way smaller than N, the hypergeometric distribution can be estimated with binomial distribution

The simulation result (shown above only for year 2015) shows that we have a higher frequency of obtaining between none to only one abnormal containing file in four-year period. The probability mass function shows that in at least three out of ten given folder, there will be none to only one abnormal containing file, and the probability of finding two or more is small.

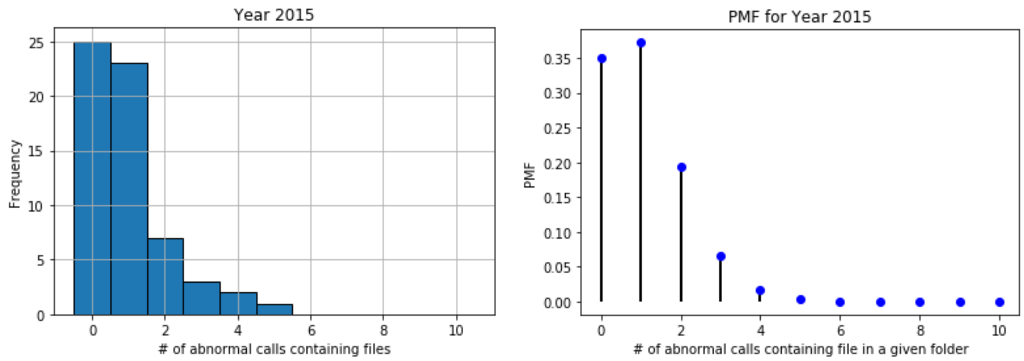

Figure 3: Binomial simulation of year 2015. Left: Histogram for a sample dataset with θ= 0.025, $n = 41$ and $size = 61$. Right: Probability mass function with same θ and n.

Another data exploration was performed on the extracted signals after noise removal. In order to detect the call shape, we use the first order differencing of frequency (dy) to represent the call trend and the second order differencing of frequency ($dy2$) to detect the curvature. The dy and $dy2$ were obtained using the following method:

$$dy_i = y_i - y_{i+1},\ dy2_i = dy_i - dy_{i+1},\ i = 1, 2, 3...$$

To illustrate the frequency differencing above, a folder containing zero-crossing files were randomly chosen and the files were processed for signal extraction. The dy mean and standard deviation were calculated and the histograms were plotted imposing a density curve. The calculation for $dy2$ mean was also performed.

From Figure 4, dy mean centers at -60 Hz with a left-skewed distribution. This indicates that most valid pulses, in the given folder, have a decreasing trend. Moreover, dy standard deviation has a right skewed distribution with a mean of 240; this reveals a handful of pulses have variable and sparse zero-crossing points within a pulse that deviate from majority of pulses. The $dy2$ mean (Figure 5) have a long-tailed but roughly symmetric distribution centered at 10 Hz, this indicates that majority of pulses have a slightly concave up shape and a small portion of pulses have more intense concave up or concave down shapes. These results are consistent with what we discovered in visualization before. This founding of signal characteristics inspires the clustering and classification techniques that we are going to introduce in the Section 3.

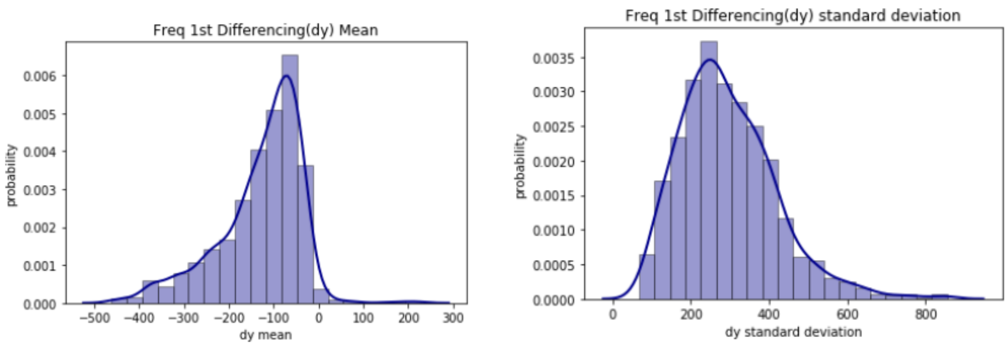

Figure 4: Frequency first differencing (dy) mean and standard deviation for pulses in the files from a given folder.

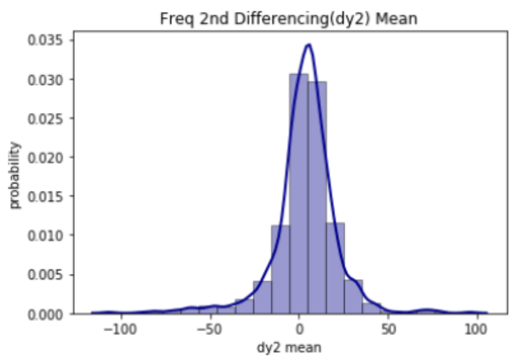

Figure 5: Frequency second differencing ($dy2$) mean

2.4. Database and Graphical User Interface

We incorporated a Python module that connects to a database cloud server. This module allows transferring to and retrieving from the database for users. At this moment, we are relying on a third-party hosting sites that provides a free-tier service. One option for us is GearHost's MySQL with 5 MB of data storage.

The graphical user interface is an interactive and presentable front end website for the users to see the progress of clustering and noise removal algorithm. This platform uses Django, a Python web framework, with the implementation of Dash, the modular dashboard application framework for Django.

3. Methodology and Model

3.1. Data Preprocessing: Noise removal, signal extraction and bulk processing

The time and frequency information were extracted from zero crossing files with Anabat sequence reader modified from Myotisoft ZCANT tool [11]. Each file was then subjected to noise removal and pulse extraction using our in-house algorithm. Our algorithm works sequentially. First, it separates the pulses and fills in holes. Then it subtracts noise, both by analyzing slope for drastic changes and by comparing to a smoothed version of the graph generated by the Savitzky-Golay algorithm.

After noise removal, the pulse information is appended to cleaned time and frequency information. A dataframe containing filename, time, frequency and pulse information for files in a given folder is stored in comma separated value (CSV) format. Another dataframe is also generated to summarize pulses information into six parameters; the duration of call (dur) which also contain the general direction of slope (+/-), the first frequency (f_{start}), the last frequency (f_{end}), median of the pulse frequency (f_{med}), and median absolute deviation of zero-crossing dots for each pulse (mad). A visualization of cleaned pulses then can be displayed as in Figure 6.

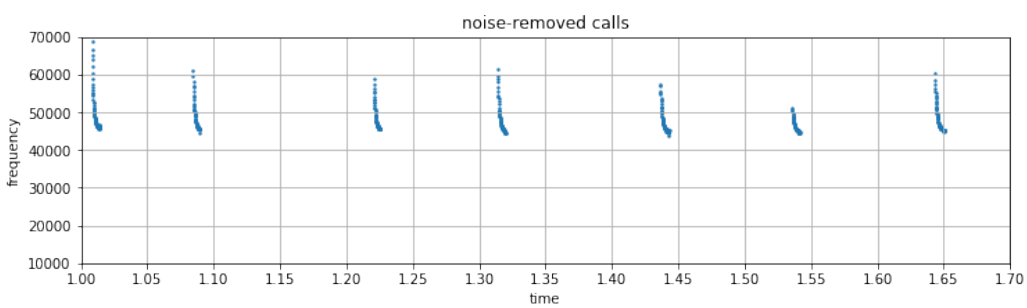

Figure 6: A clean zero-crossing after noise removal

For clustering and classification purpose, one extracted valid pulse was treated as an observation. The bulk processing imports and processes a batch of zero-crossing files or .csv files from a directory and returns all the noiseless pulses. The extracted pulses are displayed in Figure 7 with a random sample.

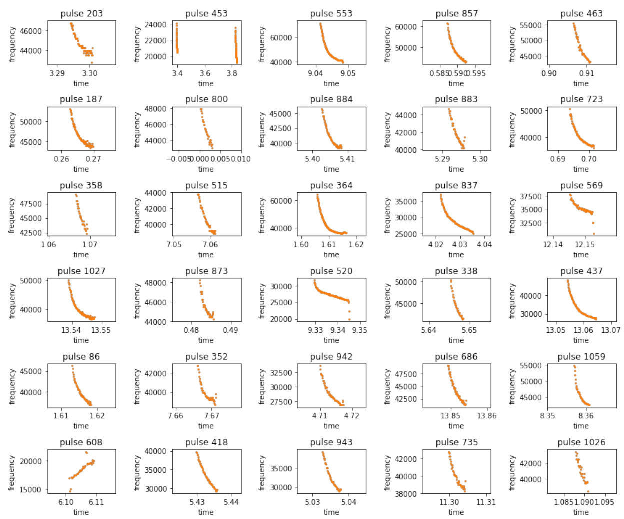

Figure 7: Display of the extracted pulses observation with a random sample

3.2. Clustering

Clustering allows the mining of different pulses. In our initial attempt, dy mean, standard deviation and five-number summary of dy: min, 1st quartile, median, 3rd quartile, max were used as clustering features with K-Means method at $k = 3$. These initial features only work well for simple pulse sets. In order to cluster more complex pulse sets, we used six more robust resistant features: the 1st quartile, median and the 3rd quartile of both dy and $dy2$ to perform K-Means clustering method with parameter $k = 3$. These features allow better detection of the pulses curvature and become the model parameters for clustering. Figure 8 below shows the results for different clusters.

Bat calls in Cluster 0 all have a decreasing trend with some degree of concave up curvature and calls in Cluster 1 also has a decreasing trend with no or slightly concave up shape. These two groups can be deemed as normal. In Cluster 2, there are many irregular shapes ranging from flat to increasing trend, which we could regard as abnormal calls. This result confirmed that the six features for K-Means clustering technique roughly support the cluster of abnormal calls among normal calls, and through this progress, we learned different shapes of abnormal calls.

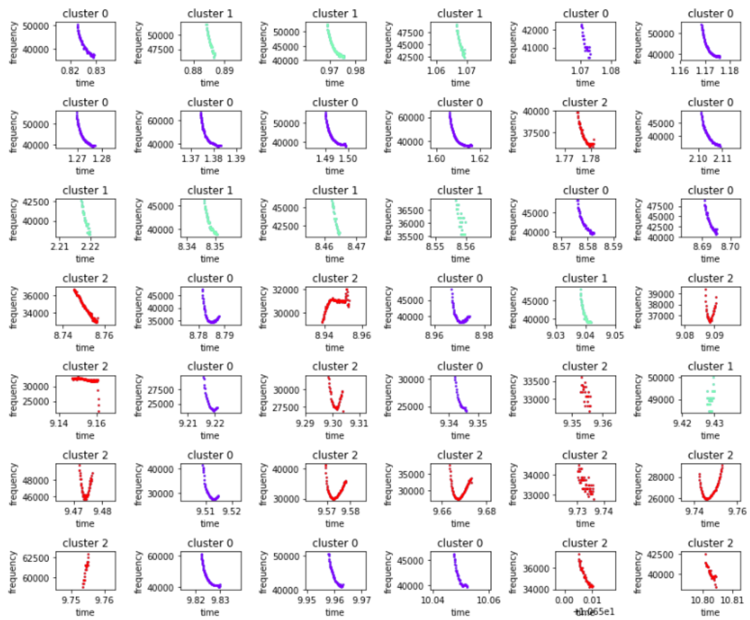

Figure 8: A clustering result via K-Means at $k = 3$ using 6 resistant features

3.3. Classification

Classification is done to look at the variation between files. If a zero-crossing file contains at least one abnormal pulse, it should be classified as abnormal labeled file. The filename-pulse set enabled the source–observation identification.

We gained insight on which clusters contain mostly abnormal calls from the clustering results. We combined that with specialist knowledge to further manually label abnormal pulses. The manual labeling of sample pulses yielded 43 abnormal pulses. We mixed in 161 normal pulses to perform classification. The comparison between normal pulses and abnormal pulses is shown in Figure 9.

A total of 204 sample observation: 161 from normal pulses and 43 from abnormal pulses (after noise removal) were used for classification. Target variable, denoted as t, is a categorical variable: 0 stands for normal and 1 stands for abnormal. We used Random Forest Classifier with 100 trees and train: test splitting ratio of 7:3. The same six features that were used previously for clustering yield the "best" classification accuracy at 90.32%, after trying different parameters combination. Table 2 and Figure 10 are the

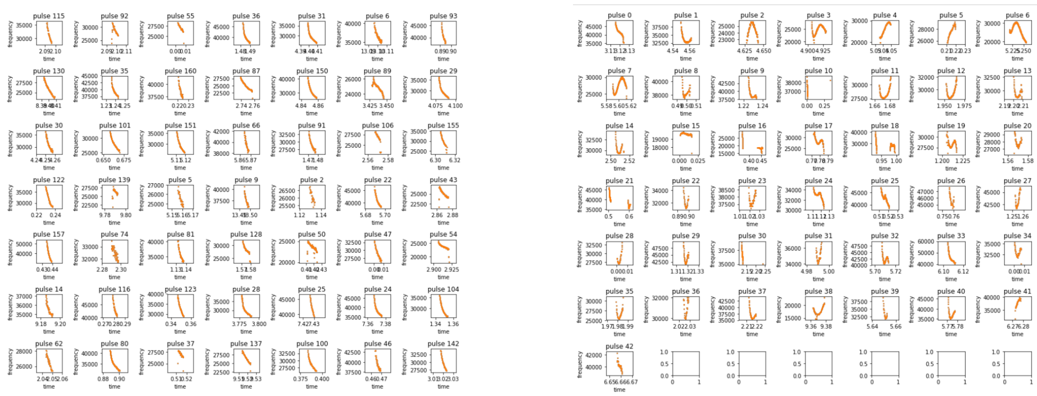

Figure 9: normal pulses vs abnormal pulses sample

		predicted label	
		0(normal)	1(abnormal)
true label	0(normal)	49	5
	1(abnormal)	1	7

Table 2: Confusion Matrix

confusion matrix of classification result.

Since our interest is in the classification of abnormal calls, we regard 1 as positive. The True Positive Rate is 0.88 and the True Negative Rate is 0.91, indicating the classification worked well. From the feature importance bar graph in Figure 11, we found out that dy was the most influential feature. The abnormal calls and normal calls can be differentiated mainly by resistant pulse trend indicator, which is the median of first differencing of frequency in a pulse.

	precision	recall	f1-score	support
normal	0.98	0.91	0.94	54
abnormal	0.58	0.88	0.70	8
avg / total	0.93	0.90	0.91	62

Table 3: Classification Report

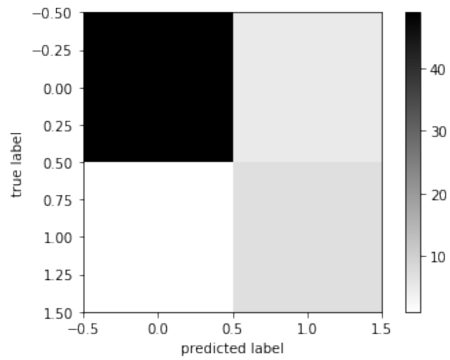

Figure 10: Confusion Matrix Plot

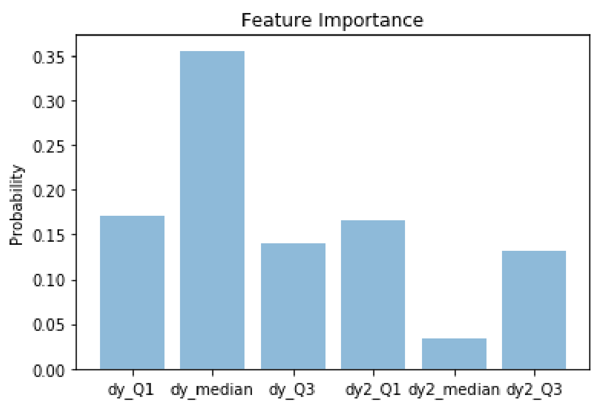

Figure 11: Feature Importance

4. Conclusion and Discussion

In this paper, we proposed a workflow of zero-crossing data processing that also performs clustering and classification analysis to label abnormal calls. The data processing allows noisy frequency dots to be removed, generate arrays and dataframe for quantitative analysis, and bulk processing to allow multiple files analysis.

In performing our work, we primarily use Python 2.7. In our experience of data extraction with modified tool from ZCANT, there was a compatibility issue with Python 3. In Python 3, every encoded data need to be decoded before use. This issue hindered teamwork, making debugging another extra step we have to take. The pre-processing for zero-crossing files and preparing a workable dataset took the majority timeframe in this project.

We had used K-Means for clustering and Random Forest Classifier for classification that yield the desired result. However, we look forward to other machine-learning methodologies to compare to our preliminary result, such as the most common linear discriminant analysis for classification, hierarchical clustering, or even the deep convolutional neural network on pulse images.

In addition to that, more labeled abnormal samples (i.e. social, foraging and feeding) can aid in machine learning technique. Statistical computing techniques such as bootstraping the estimators or parameters distribution can be considered as well. The resistant features had gone through various statistical analysis and had illustrated superiority for the methods of clustering and classification performed in this work.

This project also considers future data storage and interface platform which makes this a well-rounded project of data science.

References

[1] G. Sales, Ultrasonic Communication by animals, Springer Science and Business Media, 2012.

[2] G. W. Pierce, D. R. Griffin, Experimental determination of supersonic notes emitted by bats, Journal of Mammalogy 19(4) (1938) 454–455.

[3] R. A. Krusic, C. D. Neefus, Habitat associations of bat species in the white mountain national forest, In Bats and forest symposium (RMR Barclay and RM Brigham, eds.) (1996) 185–198.

[4] R. F. Lance, B. Bollich, C. L. Callahan, P. L. Leberg, Surveying forest-bat communities with anabat detectors, In Bats and forest symposium (RMR Barclay and RM Brigham, eds.) (1996) 175–184.

[5] D. W. Armitage, H. K. Ober, A comparison of supervised learning techniques in the classification of bat echolocation calls, Ecological Informatics 5(6) (2010) 465–473.

[6] H. U. Schnitzler, C. F. Moss, A. Denzinger, From spatial orientation to food acquisition in echolocating bats, Trends in Ecology and Evolution 18(8) (2003) 386–394.

[7] G. Jones, E. C. Teeling, The evolution of echolocation in bats, Trends in Ecology and Evolution 21(3) (2006) 149–156.

[8] E. R. Britzke, J. E. Duchamp, K. L. Murray, R. K. Swihart, L. W. Robbins, Acoustic identification of bats in the eastern united states: a comparison of parametric and nonparametric methods, The Journal of Wildlife Management 75(3) (2011) 660–667.

[9] S. Parsons, G. Jones, Acoustic identification of twelve species of echolocating bat by discriminant function analysis and artificial neural networks, Journal of Experimental Biology 203(17) (2000) 2641–2656.

[10] O. Mac Aodha, R. Gibb, K. E. Barlow, E. Browning, M. Firman, . Freeman, R., I. Pandourski, Bat detective - Deep learning tools for bat acoustic signal detection, PLoS computational biology 14(3) (2018) e1005995.

[11] D. A. Riggs, Zero-Crossing ANalysis Tool, https://github.com/riggsd/zcant, 2012.