

NOT MFCCS?!

Abstract of  
a thesis presented to the Faculty  
of the University at Albany, State University of New York  
in partial fulfillment of the requirements  
for the degree of  
Master of Arts  
College of Arts & Sciences  
Department of Mathematics & Statistics

Nicholas W. Jarrett

2010

## ABSTRACT

Mel-Frequency Cepstral Coefficients (MFCCs) are widely used for a variety of problems in signal processing. There can be limitations to this approach when you attempt to model something which cannot be easily discriminated with the human ear. This thesis investigates the results that can be obtained with a more objective frequency scale function, which will be called NMFCC. Techniques utilized include: Support Vector Machines (SVMs), autocorrelation, NMFCC, binary classification, and regression analysis. All programming for this thesis was done for the R statistical package.

Results in the discrimination problem between four computer keys were used as a baseline to assess the success of the NMFCCs. The discrimination algorithm's accuracy was increased by approximately 30% with the use of the NMFCCs.

Each problem has a different family of characteristic frequencies. The scale imposed can either obfuscate or illuminate them. While these results do not prove that objective frequency scale functions are superior to the traditional MFCCs, they provide an example where the NMFCCs reveal more important characteristics than the MFCCs.

NOT MFCCS?!

A thesis presented to the Faculty  
of the University at Albany, State University of New York  
in partial fulfillment of the requirements  
for the degree of

Master of Arts  
College of Arts & Sciences  
Department of Mathematics & Statistics

Nicholas W. Jarrett

2010

## ACKNOWLEDGEMENTS

“The world we have created is a product of our thinking;  
it cannot be changed without changing our thinking.”

- Albert Einstein

Thank you to all of the inspirational people who have opened my understanding of the world.

My thesis advisor, Dr. Carlos Rodriguez truly redefined my perception of statistics. Its my lifes passion, and his enthusiasm for Bayesian statistics has truly inspired me. The best thing thats happened to me here at the University at Albany, was being lucky enough to be in his classroom.

Dr. Karin Reinhold is my graduate advisor. Professor Reinhold has always been someone I could go to with my troubles from choosing the classes which will most benefit my subject area to just needing someone to talk to. She is truly an amazing individual.

I have given several presentations for Dr. Martin Hildebrand masters seminar. His advice has helped me to come to where I am now, particularly in my abilities as an instructor. He taught me something invaluable which I know no textbook, only his life experience, could have taught me.

I would especially like to thank my good friend, Matthew Lester. We spent countless nights working on problems together with little to no sleep. A lot of this thesis would be radically different without his input.

## TABLE OF CONTENTS

ABSTRACT.....	ii
ACKNOWLEDGEMENTS.....	iv
TABLE OF CONTENTS.....	v
CHAPTER	
1 ORIGIN OF THE MFCCS.....	1
2 CONSTRUCTION OF THE MFCCS.....	3
3 AUDITORY FEATURE EXTRACTION ON A COMPUTER....	5
4 MFCCS IN ACTION.....	7
5 THE FOUR KEY CLASSIFICATION PROBLEM.....	8
6 STRUCTURE IMPOSED BY MFCCS.....	10
7 CONSTRUCTION OF THE NMFCCS.....	12
8 CONCLUSION.....	18
REFERENCES.....	19



# 1 Origin of the MFCCs

One of the challenges in audio and visual analysis is simply to find a method to deal with the sheer size of the dataset. Humans have been very successful in performing discrimination for these kinds of problems: both in accuracy and speed. Consequentially, many techniques directed at understanding such phenomena have been developed to mimic human physiology.

At first, it may not be clear that characteristic feature extraction from the raw data is necessary. Humans don't record a list of amplitudes when we hear a sound, so we're not aware of the volume of data encoded in sound. For standard sampling rates, the text file of raw amplitudes associated with 30 seconds of sound can be over a gigabyte in size. Not only does our ear filter the sound significantly, but our brain also passes the information through filters as well. When standard speech is passed through even simple audio filters, it can become unrecognizable. However, when the listener is told what is being said before hand, they often become able to hear it. This is an example of how the brain's expectations place filters on auditory data. More than this though, the information arriving at the brain has already been cut down by the ear. This is supported by both anatomical and physiological reasons as well as experimental evidence relating to the perceptibility of slight differences in sound frequencies.

The human ear generally consists of three sections: outer, middle, and inner. See Figure 1 . For the purpose at hand, the inner is the most important. Sound waves are collected in the outer ear and channeled in through the middle ear. They enter the inner ear via the oval window. The cochlea, located in the inner ear, is essentially the auditory feature extractor for the human body. Figure 2 gives a frequency-location breakdown along the basilar membrane. The amplitude of a wave increases as it

moves along the tube until it hits its maximum. At this point it decays rapidly.

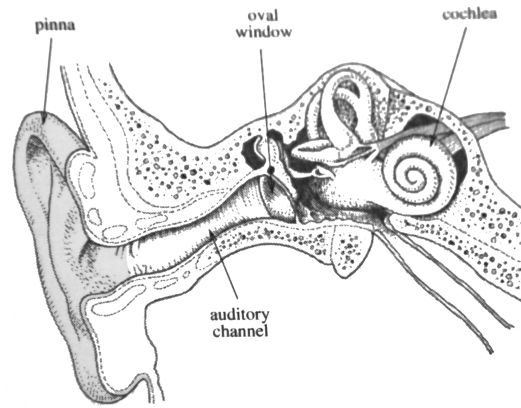


Figure 1: The Human Ear

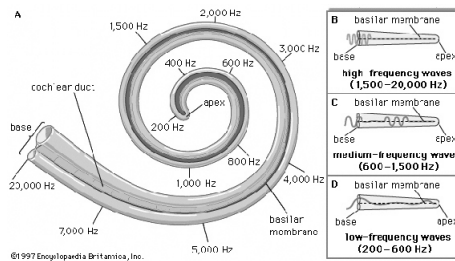


Figure 2: Basilar Membrane

We are able to perceive sounds through the use of nerves located throughout the basilar membrane sense this location and the intensity at it. The idea is basically that there are holes between our ability to determine precise location. Just as a mosquito can land on your arm without you feeling it, we may be unable to precisely determine the exact location at which a wave begins to decay along the basilar membrane. As a result, our ability to distinguish between sounds in these regions may be diminished or even non-existent. This phenomenon is characterized by considering critical bands (centered at critical frequencies) over which we are unable to tell one frequency from another.

## 2 Construction of the MFCCs

The MFCCs are based on the aforementioned properties of the human auditory feature extraction process. The intensity function for the Mel scale is:

$$M(f) = 1125 \cdot \ln \left( 1 + \frac{f}{700} \right)$$

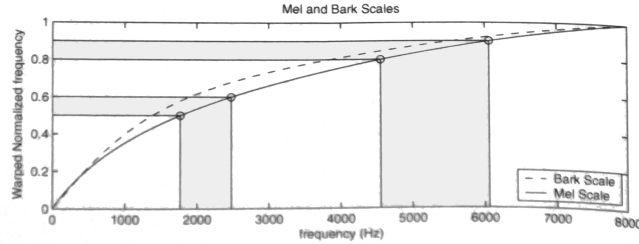


Figure 3: Mel Intensity & Critical Bands

Sampling uniformly from the Mel scale and mapping down to the axis yields the critical bands and frequencies. MFCCs attempt to mimic the filtering process of the ear by combining the energy within critical bands. Naturally, this filtering process works well for the purpose of speech processing; however, MFCCs are sometimes applied in more abstract problems.

More explicitly, the MFCCs are extracted from the signal by first converting it into frequencies using Short-time Fourier Transforms (STFTs). STFTs are FTs over incremental windows which cover the whole signal. They are of the form  $e^{-i2\pi k}$ . These results are stored in a matrix which will be denoted  $[STFTs]$ . The intensity function, desired number of coefficients, and frequency overlapping are used to define the  $[Scaling]$  matrix.  $[STFTs]$  and  $[Scaling]$  are composed by matrix multiplication to combine the energy over respective critical bands. The result is then logged and negated to extract the exponents, and composed with the inverse discrete cosine



transform  $[IDCT]$ . The IDCT can be interpreted as an inverse FT. It then becomes clear that this step essentially undoes the transformation of the signal into frequencies using FTs from the beginning of the process.

$$-\log ([STFTs] \times [Scaling]) \times [IDCT]$$

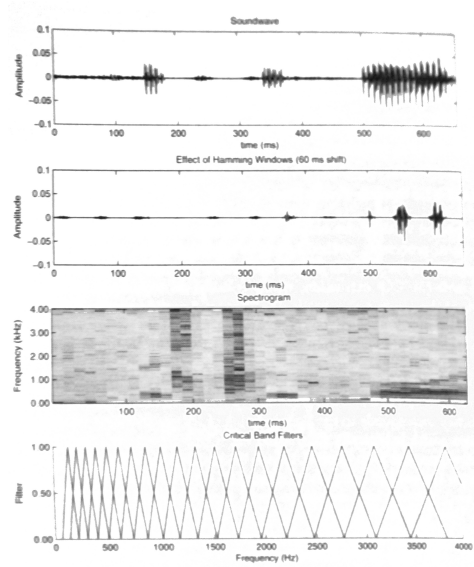


Figure 4: Mel Decomposition of a Signal

### 3 Auditory Feature Extraction on a Computer

Much like the human ear, computers acquire sound information from their environment by evaluating changes in air pressure induced by a sound wave. The mechanical analog to the ear's oval window is the microphone's elastic membrane. It fluctuates back and forth as a sound wave compresses it. By measuring the amount of compression, a computer is able to obtain a list of amplitudes at discrete times which represent the sound. The frequency of the compression is in fact identical to the frequency of the corresponding sound, and the square of the pressure is proportional to the sound's intensity. The result being that a sound wave can be completely characterized simply by measuring fluctuations in the pressure induced along the membrane.

It is important to note that sound waves are continuous, however we are only able to measure them discretely. This can potentially lead to some problems. As stated previously, the FT of the data is typically interpreted as yielding its frequency components, but the discrete nature by which we sample the data can actually obscure the true frequencies. The signal which is reconstructed from these samples need not necessarily be identical to the original wave. This behavior is referred to as aliasing. For an illustration of this effect, see figure 5.

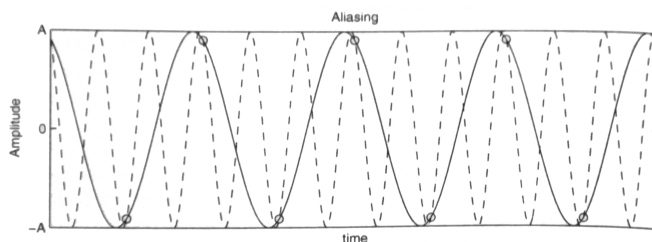



Figure 5: Aliasing



In fact, there are an infinite number of possible sine curves that can generate the same regularly sampled data points. There are several ways in which aliasing can be reduced. Clearly, a larger sampling rate reduces aliasing, since the reconstructed function has to agree with the true function at the end points of the sampled intervals. Additionally, the algorithm by which the frequencies are reconstructed can influence how profound the effects of aliasing are. This issue wasn't investigated in great detail for this thesis; however, keyboard strokes are hardly ordinary signals. It would be interesting to see if improving the characteristic frequencies we attribute to each signal by addressing the issue of aliasing could reveal more unique features for different keys.

## 4 MFCCs in Action

MFCCs have been widely applied to problems everywhere  from speech recognition to analyzing mechanically produced signals. They are often used to address the problem of music genre classification. They were shown to be highly successful for this problem in [4]. MFCCs are also used for problems with more abstract sounds. They were suggested as the best method of characteristic feature extraction for discrimination of computer keyboard emanations by [1] and [2]. They were also used in [3], to discriminate keyboard key strokes with SVMs. For the A, L, Q, and P keys, [3] was able to attain an accuracy of 65.56% using MFCCs. The implications of using MFCCs are explored in reference to this type of problem, and NMFCCs are developed in an attempt to extract features which are more useful for discrimination. The results from [3] will be used in the remainder of this thesis as a baseline which NMFCCs will be compared to.

Results must be interpreted in the context of the algorithm used to obtain them. The following section describes the approach used in [3] for the four key classification problem. It is not necessary for the theory of MFCCs and NMFCCs, and can be temporarily skipped, if the reader wishes to continue developing them from a theoretical stand point.

## 5 The Four Key Classification Problem


The training data set consists of 50 strokes of each of the A, L, Q, and P keys. The key stroke signals are first isolated in the sound file, and then two sets of features are extracted from them: autocorrelation coefficients and MFCCs. For each key, 65 autocorrelation coefficients are taken with lags ranging from 2 to 66. This is augmented by a random number of MFCCs determined by 7 tuning parameters. This matrix of features will be referred to as  $X_{training}$ . Furthermore, the true labels for the training set will be denoted  $Y_{training}$ . The SVM then learns the classification rule for  $\{X_{training}, Y_{training}\}$ . It finds this rule subject to the following constraints:

1. SVM-Type: C-classification
2. SVM-Kernel: radial

The discrimination rule is used on the features of the test data,  $X_{test}$ , to generate predicted labels  $Y_{predicted}$ . The accuracy of the discrimination rule is then evaluated by comparing the true and predicted labels,  $Y_{test}$ , and  $Y_{predicted}$ . More explicitly,  $Y_{test}$  is the following 90x1 matrix of labels:

```
> ytest =c("L", "A", "A", "L", "A", "L", "A", "A", "A", "L",  
+         "L", "L", "L", "L", "L", "A", "A", "A", "A",  
+         "A", "L", "A", "A", "L", "A", "A", "A", "L",  
+         "A", "L", "L", "A", "A", "A", "A", "L", "A",  
+         "L", "L", "A", "L", "A", "L", "A", "L", "P", "Q", "Q", "Q",  
+         "Q", "Q", "Q", "Q", "P", "P", "Q", "Q", "Q",  
+         "Q", "Q", "Q", "P", "P", "P", "P", "P", "P", "Q", "Q", "Q",  
+         "Q", "Q", "Q", "P", "Q", "Q", "Q", "Q", "P", "Q", "P", "Q",  
+         "P", "Q", "P", "P", "P", "P", "P", "Q")
```

The discrimination was done with three binary decisions:

1. A against not A. 
2. L against not L in what was not classified as an A in the first step.
3. Q against not Q in what was not classified as an L in the second step.
4. P is classified as whatever is not classified as Q in the third step.

Each step of the binary classification behaves as described above. More precisely, each binary classification step goes through the below process:

A against not A:

1. Let 0 denote “not A.”
2. Define a new set of labels for the training data,  $Y_{A_{training}}$  which contains only A’s and 0’s.
3. Train the SVM on  $\{X_{training}, Y_{A_{training}}\}$ .
4. Compare the learned decision rule with  $Y_{A_{test}}$  which contains only A’s and 0’s.
5. Remove everything classified as an A in this binary decision step.
6. Move on to the L against not L classification.

The total accuracy of the algorithm is defined as the percentage of total keys correctly predicted. As stated previously, the highest number reached using MFCCs was 65.56%.

## 6 Structure Imposed by MFCCs

Humans aren't very good at distinguishing keyboard strokes. The whole point of MFCCs is to emulate the human auditory filtration process to extract the key features of the data. This only truly makes sense if humans have been shown to be able to do discrimination well for the data set in question. i.e. if the features extracted by humans have been shown to work well for discrimination. MFCCs may be very appropriate for problems involving speech recognition, but it's not immediately clear that they are the best choice for the keyboard classification problem.

On a more fundamental level, the steepness of the intensity function of the Mel scale determines the shape of the bands of frequencies whose energies are summed together. It can thus be seen that this intensity function is actually imposing a structure on the characteristic features you can observe. The intensity function rises sharply over low frequencies and then slowly flattens out. This results in fine critical bands at low frequencies and large bands at high frequencies.

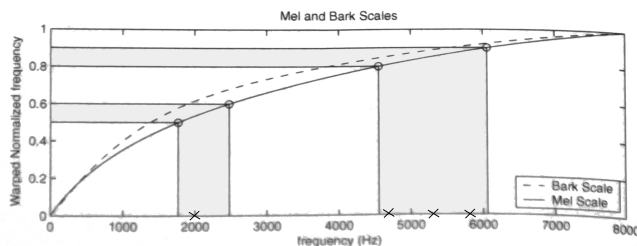


Figure 6: Mel Scale and Incompatible True Characteristics

Suppose that the true characteristic features of your data which allow discrimination are not concentrated at low frequencies, such an algorithm might group them together. See Figure 6 for an illustration. The characteristic features that are useful for discrimination are labeled by x's. Only the joint effects of characteristic frequen-

cies belonging to the same band are maintained. An intensity function which is steep for high frequencies and relatively flat for low frequencies, would yield critical bands which are more compatible with the data in Figure 6. Using the same number of critical bands, it would be possible for such a function to maintain the individual effects of more of these features.

This is just one example of a possible distribution of features. The idea behind NMFCCs is that it is better to allow the data to tell you where its characteristic features are, rather than presuming their density over the frequency domain a priori.



## 7 Construction of the NMFCCs

NMFCCs are for the most part identical to MFCCs. The key difference is that their intensity function is modeled to fit the types of sounds which are being analyzed, as opposed to being based on the principles of human physiology. As with the traditional MFCCs, the derivation of the NMFCCs begins with the Short-time Fourier Transform of the data. This maps the data from the time-amplitude domain into the time-frequency domain. The goal is to obtain a new intensity function that will represent the structure of the characteristic features better. To do this, the frequencies that were obtained from the FT were sorted from least to greatest. This removes time ordering sensitive data from the intensity function. Recall:



$$-\log ([STFTs] \times [Scaling]) \times [IDCT]$$

The STFT's are composed with the ordered scale thus summing along respective critical bands and recovering at least some of the time ordering.

Since there is variability inherent in the properties of key strokes, the frequency decompositions for many keys were combined to get a better model for an average key hit. As a result, the size of the representation obtained for the new intensity function is large. For the problem of key discrimination, it was characterized by approximately 90,000 entries. In order to construct the new *[Scaling]* matrix, new frequencies, not necessarily in the original 90,000 characteristics need to be mapped by the new intensity function. To achieve this, a parametric model for the intensity function was created.

First the frequency decomposition for a collection of keys is stored in *F*. The data is sorted from least to greatest among each STFT window, and then from least

to greatest for each position over all windows. The result is that the columns contain the  $n^{th}$  smallest entry for each window sorted from least to greatest.

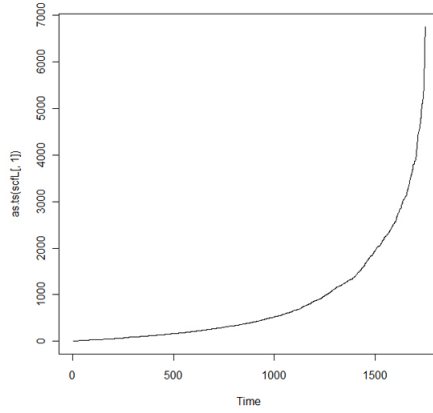


Figure 7: 1<sup>st</sup> smallest

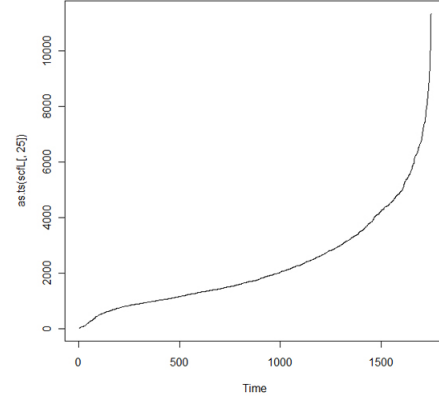


Figure 8: 25<sup>th</sup> smallest

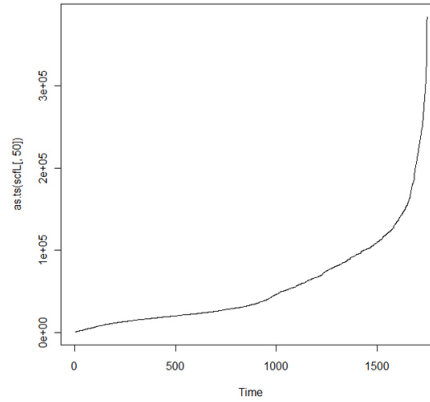


Figure 9: 50<sup>th</sup> smallest

These are initially compared with exponentials centered about the mean of domain. The residuals from this regression are used to help define the next set of parameters which will be introduced. In greater detail:

1. Define  $T$  to be a sequence starting from 1.
2. Find the optimal  $\beta_0$  for  $frequency = \beta_0 \cdot e^{0.01(T-\mu)}$  subject to standard regression assumptions.

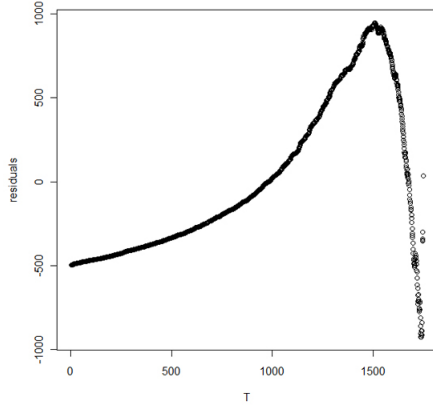


Figure 10: Residuals for 1<sup>st</sup>

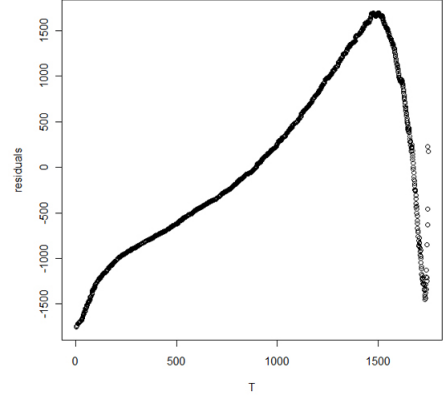


Figure 11: Residuals for 25<sup>th</sup>

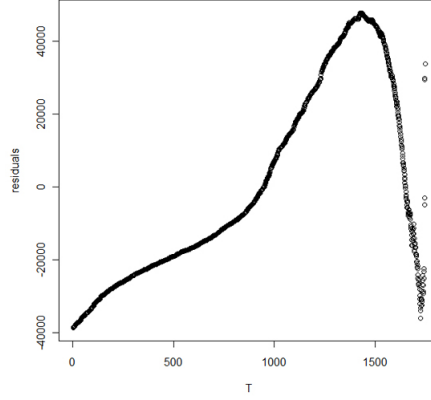


Figure 12: Residuals for 50<sup>th</sup>

The data points after the minimum residual occurs are not considered. It is possible that there could be important information in these few final points, but they do not follow the same form as the rest of the data. Instead of adding more parameters into the model to explain them sufficiently, they were thrown out. Exploring the amount of information which is contained in these points is an area which further research could investigate.

The locations of the maximum and minimum residuals from the above regressions are used to define the cutoffs for a spline model which is exponential over the first

region and the sum of an exponential and a line over the second region. Some examples of the residuals after the separation of the exponential and introduction of the linear component are given below:

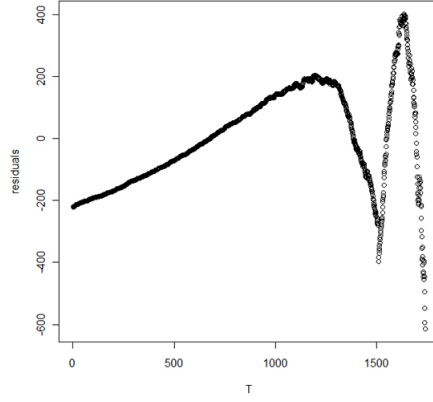


Figure 13: Residuals for 1<sup>st</sup>

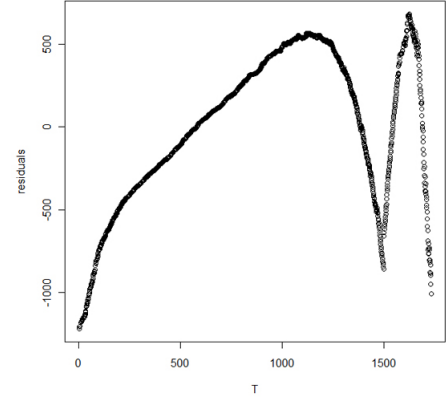


Figure 14: Residuals for 25<sup>th</sup>

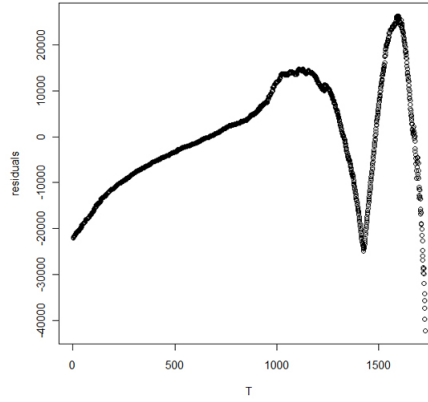


Figure 15: Residuals for 50<sup>th</sup>

The fully sorted data is shown in the following graph. The information gained over the aforementioned regressions will be used to help create a parametric model.

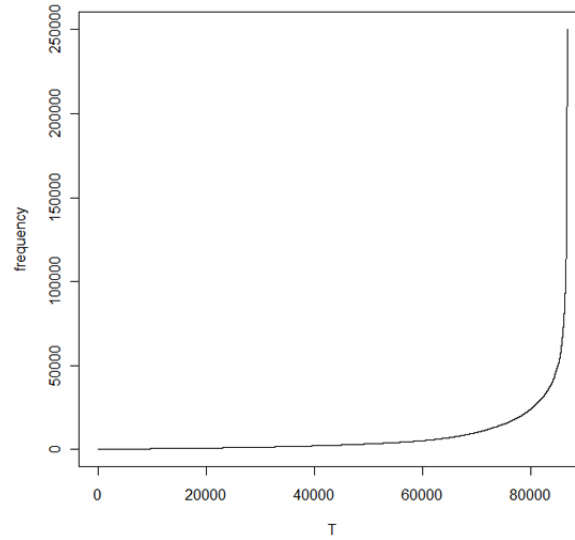


Figure 16: Combined Sorted Frequencies

These regressions are applied on a larger scale to the combined data set. The information gained about the residuals is used to search more efficiently for the optimal cutoff points for the spline. The functions defined on this region are also dependent on the indices at which the maximum and minimum residuals were found in the preliminary steps.

#### Summary of the Regression applied to the Combined Sorted Data

Call:

```
lm(formula = AllscfL[1:K2] ~ y2f + ys2f + linf)
```

Residuals:

Min	1Q	Median	3Q	Max
-87988.3	-531.9	107.1	792.7	44758.7

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	7.488e+02	1.267e+01	59.11	<2e-16 ***
y2f	7.152e+02	7.101e-01	1007.16	<2e-16 ***
ys2f	1.586e-183	0.000e+00	Inf	<2e-16 ***
linf	-7.222e+02	2.454e+00	-294.26	<2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3267 on 86773 degrees of freedom

Multiple R-squared: 0.9473, Adjusted R-squared: 0.9473

F-statistic: 5.199e+05 on 3 and 86773 DF, p-value: < 2.2e-16

This will serve as the NMFCC intensity function for the four key classification problem.

## 8 Performance & Conclusion

The NMFCCs were used in [3]’s algorithm in lieu of the MFCCs. Without increasing the number of coefficients, the accuracy was improved by approximately 30.51% from 65.56% to 85.56%. There are many aspects of NMFCCs which remain to be investigated. The approximation for the intensity function which was used still had very large errors in some spots. NMFCCs are constructed by the prevalence of certain frequencies in a signal, but this may not be truly representative of the characteristic frequencies which allow for discrimination. Intensity functions which are defined on the variability in the prevalence of frequencies in certain areas may perform better.

In this case, it is clear that objective intensity functions yield better features for discrimination. For speech recognition problems, MFCCs are logically very good features to use; however, to make these results more general, future research could look into the classes of problems for which NMFCCs and objective frequency functions provide significant improvement of extracted features.

## References

- [1] Agrawal, Rakesh, and Asonov, Dmitri, “Keyboard Acoustic Emanations,” *IBM Almaden Research Center*, pp. 1-9, 2004.
- [2] Meadows, Catherine, and Paul Syverson, “Keyboard Acoustic Emanations Revisited,” *Proceedings of the 12th ACM Conference on Computer and Communications Security* pp.373-382, New York: ACM, Nov. 2005.
- [3] Lester, Matthew D. “Keyboard Classification” Thesis. University At Albany, State University of New York, 2010. Print.
- [4] Jensen, Jesper Højvang, and Christensen, Mads Græsbøll, “Evaluation Of MFCC Estimation Techniques For Music Similarity,” “<http://kom.aau.dk/~jhj/files/jensen06mfcc.pdf>”
- [5] Camastra, Francesco, and Alessandro, Vinciarelli, “Machine Learning for Audio, Image and Video Analysis: Theory and Applications,” London: Springer, 2008.
- [6] “Fourier Transform,” *Wikipedia, the Free Encyclopedia*, “[http://en.wikipedia.org/wiki/Fourier\\_transform](http://en.wikipedia.org/wiki/Fourier_transform)”.
- [7] “Discrete Cosine Transform,” *Wikipedia, the Free Encyclopedia*, “[http://en.wikipedia.org/wiki/Discrete\\_cosine\\_transform](http://en.wikipedia.org/wiki/Discrete_cosine_transform)”.
- [8] “Mel-frequency cepstrum,” *Wikipedia, the Free Encyclopedia*, “[http://en.wikipedia.org/wiki/Mel-frequency\\_cepstral\\_coefficient](http://en.wikipedia.org/wiki/Mel-frequency_cepstral_coefficient)”
- [9] Encyclopaedia Britannica. Chicago: Encyclopaedia Britannica, 1997. Print.