**A**

**Seminar Report**

On

**“Smart Voice System”**

Submitted

In partial fulfilment

For the award of the Degree of

**BACHELOR OF TECHNOLOGY**

In

** COMPUTER ENGINEERING**

|  |  |
| --- | --- |
| **Submitted to:**  Ms. Apoorva Sharma  Assistant Professor | **Submitted By:**  Aman Panchal  Aakash Yadav  Ashish Meena  Dheeren Divya  Ajay Joshi  CS(7th Sem) |
| **Department of Computer Science &Engineering**  **Arya Institute of Engineering & Technology, Jaipur**  **Rajasthan Technical University, Kota [2020]** |  |
|  |  |
|  |  |

***Candidate’s Declaration***

I hereby declare that the work, which is being presented in the Seminar Reportin partial fulfilment for the award of Degree of “Bachelor of Technology” in Computer Engineering, and submitted to the **Department of Computer Science & Engineering,** ARYA Institute of Engineering & Technology, Affiliated to Rajasthan Technical University is a record of my own work carried out under the Guidance of **Ms.** Apoorva Sharma, Assistant Professor, Department of Computer Science & Engineering.

**Student Name**

Aman Panchal(17EAICS019)

Aakash Yadav(17EAICS001)

Ashish Meena(17EAICS033)

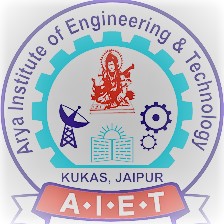
Dheeren Divya(17EAICS013)

Ajay Joshi(17EAICS010)

**ARYA INSTITUTE OF ENGINEERING & TECHNOLOGY**

**SP-40, RIICO Industrial Area, Jaipur (Raj)-302022**

***Department of Computer Science & Engineering***



***Certificate***

*This is to certify that the work, which is being presented in the project Stage-I entitled“IOT TEMPERATURE AND FACE MASK SCANENTRY SYSTEM FOR COVID” submitted by Mr. Shiv kumarshrivastav, Mr. Gaurav Totla, Mr. Yashvardhan and Mr. Yuvraj Singh a student of fourth year (VIII Sem) B.Tech. in Computer Engineering in partial fulfilment for the award of degree of Bachelor of Technology is a record of student’s work carried out and found satisfactory for submission.*

**Ms. Apoorva Sharma**  **Dr. Manish Kumar**

**Project Coordinator Head of department**

***Acknowledgement***

We wish to express our deep sense of gratitude to our Project Guide & Coordinator **Ms.** Apoorva Sharma for guiding me from the inception till the completion of the project. We sincerely acknowledge her for giving her valuable guidance, support for literature survey, critical reviews and comments for our Project.

We would like to first of all express our thanks to **Dr. Arvind Agarwal**, Chairman of Arya Main Campus, for providing us such a great infrastructure and environment for our overall development.

We express sincere thanks tothe Principal of AIET, for his kind cooperation and extendible support towards the completion of our project. Words are inadequate in offering our thanks to **Er. Manish Mukhija**, Head of CSE Department, for consistent encouragement and support for shaping our project in the presentable form.

We also like to express our thanks to all supporting CSE faculty members who have been a constant source of encouragement for successful completion of the project.

Also, our warm thanks to **Arya Institute of Engineering & Technology**, who provided us this opportunity to carryout, this prestigious Project and enhance our learning in various technical fields.

**Student Name**

Aman Panchal

Aakash Yadav

Ashish Meena

Dheeren Divya

Ajay Joshi

**ABSTRACT**

In this project, we were experiment with a self created audio dataset, and to explore how machine learning algorithms can be used to predict user voice. We were expected to gain experience using a common data-mining and machine learning library, and were expected to submit a report about the dataset and the algorithms used. After performing the required tasks on a dataset of my choice, herein lies in this report.

**Keyword :** Machine Learning, Numpy, Pandas, Matplotlib, Logistic Regression, K-nearest neighbour, Support Vector Machine, Decision Tree Classifier.

**Table of contents**

|  |  |  |
| --- | --- | --- |
|  | Candidate’s Declaration | ii |
|  | Certificate | iii |
|  | Acknowledgement | iv |
|  | Abstract | v |
|  | Table of contents | Vi |
|  | List of Figures | vii |
|  |  |  |
| 1. | Introduction | 1 |
| 2. | The Project | 1 |
|  | 2.1 Data | 1 |
|  | 2.2 Machine Learning | 4 |
| 3. | Technical Details | 11 |
| 4. | Figure of Project | 11 |
| 5. | Flow Chart | 12 |
| 6. | Results: | 12 |
| 7. | Conclusion and Future Work | 13 |
| 8. | Team Member details | 13 |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| 1. | Correlations between the attributes using heat map.................... | 4 |
| 2. | Sampling over the dataset............................................................ | 5 |
| 3. | Logistic Regression example represented graphically................ | 6 |
| 4. | Understanding K.......................................................................... | 8 |
| 5. | Euclidean Distance……………………………………………... | 9 |
| 6. | Approach of kNN Algorithm…………………………………... | 10 |
| 7. | Example of Decision Tree……………………………………… | 11 |
| 8. | Classifying Grasshopper using Decision Tree………………… | 13 |
| 9. | Random Forest............................................................................. | 14 |
| 10. | Example of How does Random Forest algorithm work……… | 15 |

**1. Introduction:**

This project has been done as part of my course for the B.tech Computer Science in Arya Institute of Engineering and Technology. Supervised by Ms. Dhara Upadhyay. Being interested in everything having a relation with the Machine Learning,the independant project was a great occasion to give me the time to learn and confirm my interest for this field. We can use Machine Learning in Finance, Medicine, almost everywhere. That’swhy I decided to conduct my project around the Machine Learning.

This model will predict that user voice data and used to lock and unlock door using arduino.

**The Objective of Project:-**

The objective of the project is design an IOT based system of smart voice lock with the model of speech recognition using MFCC extraction. This device lock/unlock on user voice.

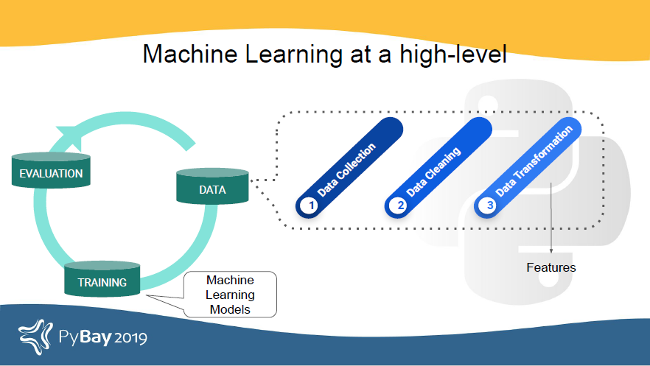
**Description Of the Solution Implemented:-**

The motivation for smart voice lock with speech recognition is simple; It is main principle of communication and is, therefore, a convenient and accessible way of communication with machines.

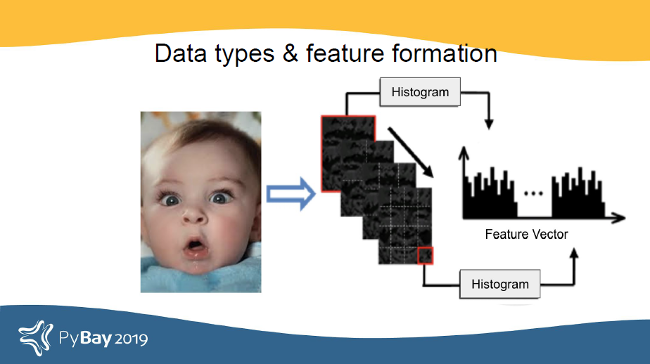
And we can easily use this technology to interact with various electronic devices . And in our project we used it to lock and unlock the lockage’s. We can use use user voice and perform speech processing by using various machine learning and neural networks and train a model which match the voice of user and if get match than he can operate the device/lock .

**THEORETICAL BACKGROUND**

At a high level, any machine learning problem can be divided into three types of tasks: data tasks (data collection, data cleaning, and feature formation), training (building machine learning models using data features), and evaluation (assessing the model). Features, [defined](https://en.wikipedia.org/wiki/Feature_(machine_learning)) as "individual measurable propert[ies] or characteristic[s] of a phenomenon being observed," are very useful because they help a machine understand the data and classify it into categories or predict a value.



Different data types use very different processing techniques. Take the example of an image as a data type: it looks like one thing to the human eye, but a machine sees it differently after it is transformed into numerical features derived from the image's pixel values using different filters (depending on the application).



[Word2vec](https://en.wikipedia.org/wiki/Word2vec) works great for processing bodies of text. It represents words as vectors of numbers, and the distance between two word vectors determines how similar the words are. If we try to apply Word2vec to numerical data, the results probably will not make sense.

**What are audio signals?**

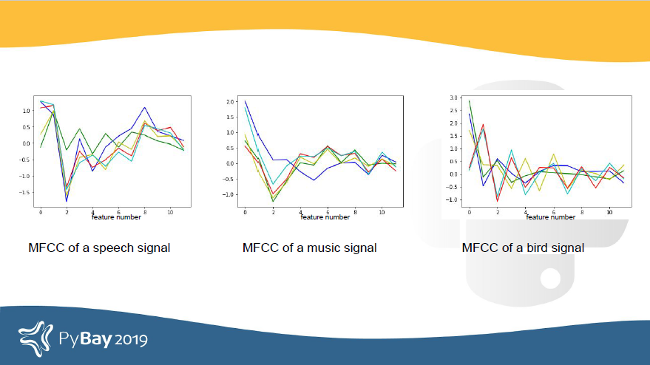
Audio signals are signals that vibrate in the audible frequency range. When someone talks, it generates air pressure signals; the ear takes in these air pressure differences and communicates with the brain. That's how the brain helps a person recognize that the signal is speech and understand what someone is saying.

There are a lot of MATLAB tools to perform audio processing, but not as many exist in Python. Before we get into some of the tools that can be used to process audio signals in Python, let's examine some of the features of audio that apply to audio processing and machine learning.

## Building a classifier

As a quick experiment, let's try building a classifier with spectral features and MFCC, GFCC, and a combination of MFCCs and GFCCs using an open source Python-based library called pyAudioProcessing.

To start, we want pyAudioProcessing to classify audio into three categories: speech, music, or birds.



Some genres do well while others have room for improvement. Some things that can be explored from this data include:

* Data quality check: Is more data needed?
* Features around the beat and other aspects of music audio
* Features other than audio, like transcription and text
* Would a different classifier be better? There has been research on using neural networks to classify music genres.

**2. The Project:**

**2.1 Data**

The crucial element in machine learning task for which a particular attention should be clearly taken is the data. Indeed the results will be highly influenced by the data based on where did we find them, how are they formatted, are they consistent, is there any outlier and so on. At this step, many questions should be answered in order to guarantee that the learning algorithm will be efficient and accurate.

Many sub steps are taken to get, clean and transform the data. I am going to explain each one of them to show how they have been applied on my project why they are useful forthe machine learning part.

**2.1.1 Getting the data:**

The first problem was where I can get the data to build a large enough dataset since I want to be able to predict the voice for a given user according to the audio features.

And we created the dataset artificially of two user and extract features from it.

**●Cleaning:** It is the first module called to clean the item and verify that all the information in it correspond to the pattern used to extract it The cleaning module removes the noise, and check that all the values are not empty, otherwise the item is dropped. This is done for simplicity,

indeed, it could be better to try to inference them later. After the cleaning part done, the item is sent to the formatting module.

**●Formatting:** the second module is used to format the item’s values as we want. A basic example could be for the price, initially got being string type, is converted as float. This is done for every numeric values. The item formatted is then sent to the last module called Integrating.

**●Integrating:** this module, the last one, is basically the one in charges of saving the items in the format that we want. It also checks that there is no redundancies between the tuples . In my case, I decided to save them in an excel sheet for each website

**The six features ​extracted for each estate were the following**

1. File name
2. class

**2.1.2 Attribute Types**

| **file\_name** | **class** | |
| --- | --- | --- |
| Recording (8)-converted.wav | | aman |
| Recording (9)-converted.wav | | aman |
| Recording (6)-converted.wav | | aman |
| Recording (7)-converted.wav | | aman |
| Recording (14)-converted.wav | | aman |

# MFCC (Mel Frequency Cepstral Coefficients) for Audio format

Mel Frequency Cepstral Co-efficients (MFCC) is an internal audio representation format which is easy to work on. This is similar to JPG format for images. We have demonstrated the ideas of MFCC with code examples.

# Steps to convert audio in MFCC :

1) Get your audio in a time domain format.

2) Covert your audio in a periodogram with the help of Fast Fourier Tranform. We do so as it will give us a Nyquist frequency by downsampling of your audio so that we can identufy the sound.

3) After this we convert our periodogram into spectrogram(they are periodograms at different intervals stacked together)

.4) Then we perform Short Fourier Tranform idea behind performing this is that it helps us to study a short interval of audio which is assumed to be steady.

5) Then we perform hamming window (The Hamming window is an extension of the Hann window in the sense that it is a raised cosine window of the form) to prevent spectral leakage(Spectral Leakage is a a phenomenon that takes place due finite windowing of the data. Generally when we take data and pass it to the DFT/FFT algorithm ).

6) We then again perform Fast Fourier Transform to convert amplitude into frequency.

7) Now we convert our frequency into mel scale as they are of better use as discussed above this is done by providing a 26 filter which is defined by macine itself and this help in you machine to learn.

8) Then we perform logarithm of all filterbank energies.

**2.1.3 Basic Statistical Measures**

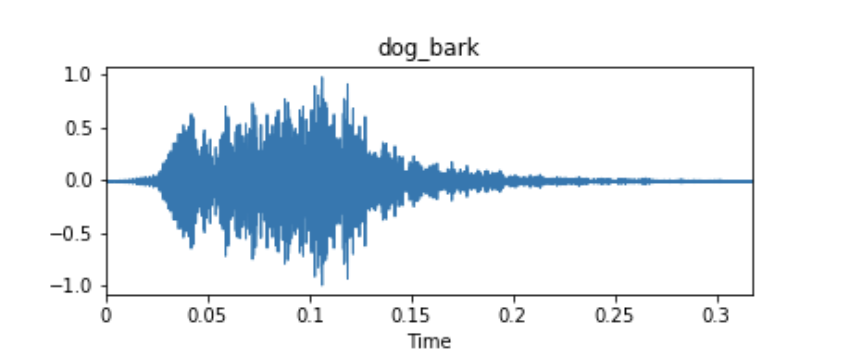
Than features extracted from the audio dataset and few rows in the below table.

| **feature** | **class** |
| --- | --- |
| [-386.7757, 185.17813, -29.811876, 12.254722, ... | akash |
| [-404.7559, 169.52423, -34.098152, 8.080881, 3... | akash |
| [-457.90625, 136.71925, -13.993755, 7.679388, ... | akash |
| [-416.07578, 188.42595, -45.392323, 9.870244, ... | akash |
| [-383.62897, 180.18144, -42.557625, 1.5801047,... | akash |

.

**2.2 Machine Learning:**

The Machine Learning part is about trying to find the best learning algorithm for a given problem even if it is highly conditioned by how well the data has been processed and tune some parameters to improve it. Depending on the problem, if it is supervised (meaning we build a model from labelled training set, the value of the dependent variable is known) or fit unsupervised (the model is built on unstructured and unlabeled data), if it is a regression or classification problem, many learning algorithm exist each with their benefits and drawbacks.

Building machine learning models to classify, describe, or generate audio typically concerns modeling tasks where the input data are audio samples.

Example waveform of an audio dataset sample from UrbanSound8k

These audio samples are usually represented as time series, where the y-axis measurement is the amplitude of the waveform. The amplitude is usually measured as a function of the change in pressure around the microphone or receiver device that originally picked up the audio. Unless there is metadata associated with your audio samples, these time series signals will often be your only input data for fitting a model.

**Basic Terminology**

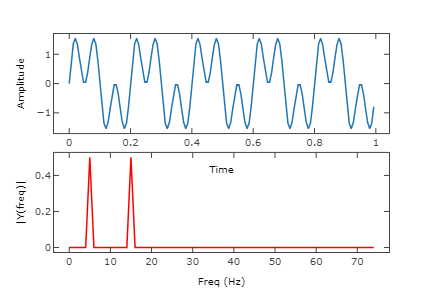
**Amplitude:**

The **amplitude** of a sound wave is a measure of its change over a period (usually of time). Another common definition of amplitude is a function of the magnitude of the difference between a variable’s extreme values.

Fourier Transform

https://miro.medium.com/max/585/0*1f8Epv3yoxb6ItBl

The **Fourier Transform** decomposes a function of time (signal) into constituent frequencies. In the same way a musical chord can be expressed by the volumes and frequencies of its constituent notes, a Fourier Transform of a function displays the amplitude (amount) of each frequency present in the underlying function (signal).

Top: a digital signal; Bottom: the Fourier Transform of the signal

**2.2.1 Sampling**

Given my dataset, I applied a sampling technique in order to divide it into differentsubset having each its own utility. It is commonly assumed that more we have data to build amodel more it will have tend to give good results. Usually the dataset is divided as follow withtheir respective utility :



Figure 2: Sampling over the dataset.

**2.2.2 Learning Algorithm**

**Logistic Regression**

**“Logistic Regression is based on this principle: it expresses the multiple logistic regression equation in logarithmic terms(called the logit) and thus overcomes the problem of violating the assumption of Linearity.”**

In order to understand the difference between logistic and linear regression, we need to first understand the difference between a continuous and a categoric variable.

Continuous variables are numeric values. They have an infinite number of values between any two given values. Examples include the length of a video or the time a payment is received or the population of a city.

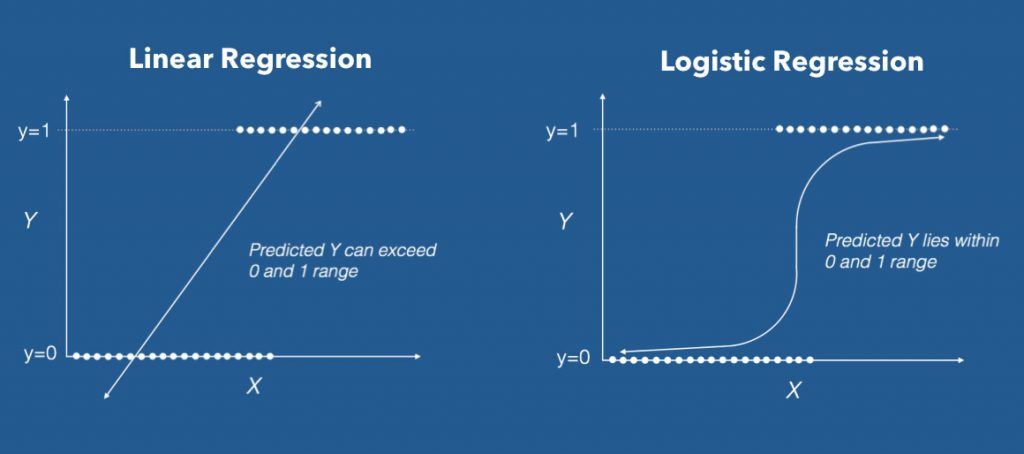
Categorical variables, on the other hand, have distinct groups or categories. They may or may not have a logical order. Examples include gender, payment method, age bracket and so on.

In linear regression, the dependent variable Y is always a continuous variable. If the variable Y is a categorical variable, then linear regression cannot be applied.

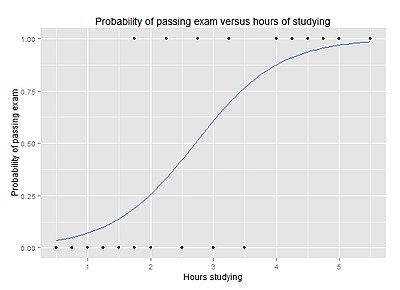
In case Y is a categorical variable that has only 2 classes, logistic regression can be used to overcome this problem. Such problems are also known as binary classification problems.

It’s also important to understand that standard logistic regression can only be used for binary classification problems. If Y has more than 2 classes, it becomes a multi-class classification and standard logistic regression cannot be applied.

One of the biggest advantages of logistic regression analysis is that it can compute a prediction probability score for an event. This makes it an invaluable predictive modeling technique for data analytics.

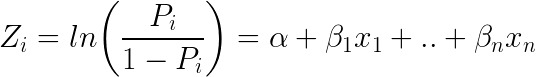


**Figure 3: Logistic Regression example represented graphically**



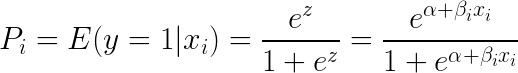
**How does Logistic Regression Work?**

Here’s what the logistic equation looks like:



Logistic Regression Equation

Taking e (exponent) on both sides of the equation results in:



Equation with E Exponent

Here’s how the equation can be implemented in R:

# Template code

# Step 1: Build Logit Model on Training Dataset

logitMod <- glm(Y ~ X1 + X2, family=“binomial”, data = trainingData)

# Step 2: Predict Y on Test Dataset

predictedY <- predict(logitMod, testData, type=“response”)

# Random Forest Algorithm

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of **ensemble learning,**

***"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."*** Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

**The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.**



Figure 9: Random Forest

**Why use Random Forest?**

Below are some points that explain why we should use the Random Forest algorithm:

* It takes less training time as compared to other algorithms.
* It predicts output with high accuracy, even for the large dataset it runs efficiently.
* It can also maintain accuracy when a large proportion of data is missing.

**How does Random Forest algorithm work?**

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

**Step-1:** Select random K data points from the training set.

**Step-2:** Build the decision trees associated with the selected data points (Subsets).

**Step-3:** Choose the number N for decision trees that you want to build.

**Step-4:** Repeat Step 1 & 2.

**Step-5:** For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

The working of the algorithm can be better understood by the below example:



Figure 10: Example of How does Random Forest algorithm work

# Deep Learning

Deep learning can be considered as a subset of [machine learning](https://www.simplilearn.com/tutorials/machine-learning-tutorial/what-is-machine-learning). It is a field that is based on learning and improving on its own by examining computer algorithms. While machine learning uses simpler concepts, deep learning works with artificial neural networks, which are designed to imitate how humans think and learn. Until recently, neural networks were limited by computing power and thus were limited in complexity. However, advancements in [Big Data analytics](https://www.simplilearn.com/what-is-big-data-analytics-article) have permitted larger, sophisticated neural networks, allowing computers to observe, learn, and react to complex situations faster than humans. Deep learning has aided image classification, language translation, speech recognition. It can be used to solve any pattern recognition problem and without human intervention.

[Artificial neural networks](https://www.simplilearn.com/multilayer-artificial-neural-network-tutorial), comprising many layers, drive deep learning. Deep Neural Networks (DNNs) are such types of networks where each layer can perform complex operations such as representation and abstraction that make sense of images, sound, and text. Considered the fastest-growing field in machine learning, deep learning represents a truly disruptive digital technology, and it is being used by increasingly more companies to create new business models.

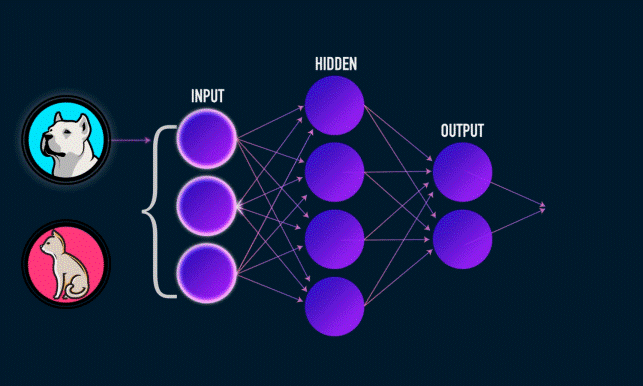


Figure 11: Example of Deep Learning

Activation Function :

Biological neural networks inspired the development of artificial neural networks. The activation function takes the decision of whether or not to pass the signal. In this case, it is a simple step function with a single parameter.

**Popular types of activation functions**

### 1. Binary Step Function

f(x) = 1, x>=0

= 0, x<0

### 2. Linear Function

f(x)=ax

### 3. Sigmoid

f(x) = 1/(1+e^-x)

### 4. Tanh

tanh(x)=2sigmoid(2x)-1

### 5. ReLU

f(x)=max(0,x)

**3.Technical Details:-**

**Hardware Used:-**

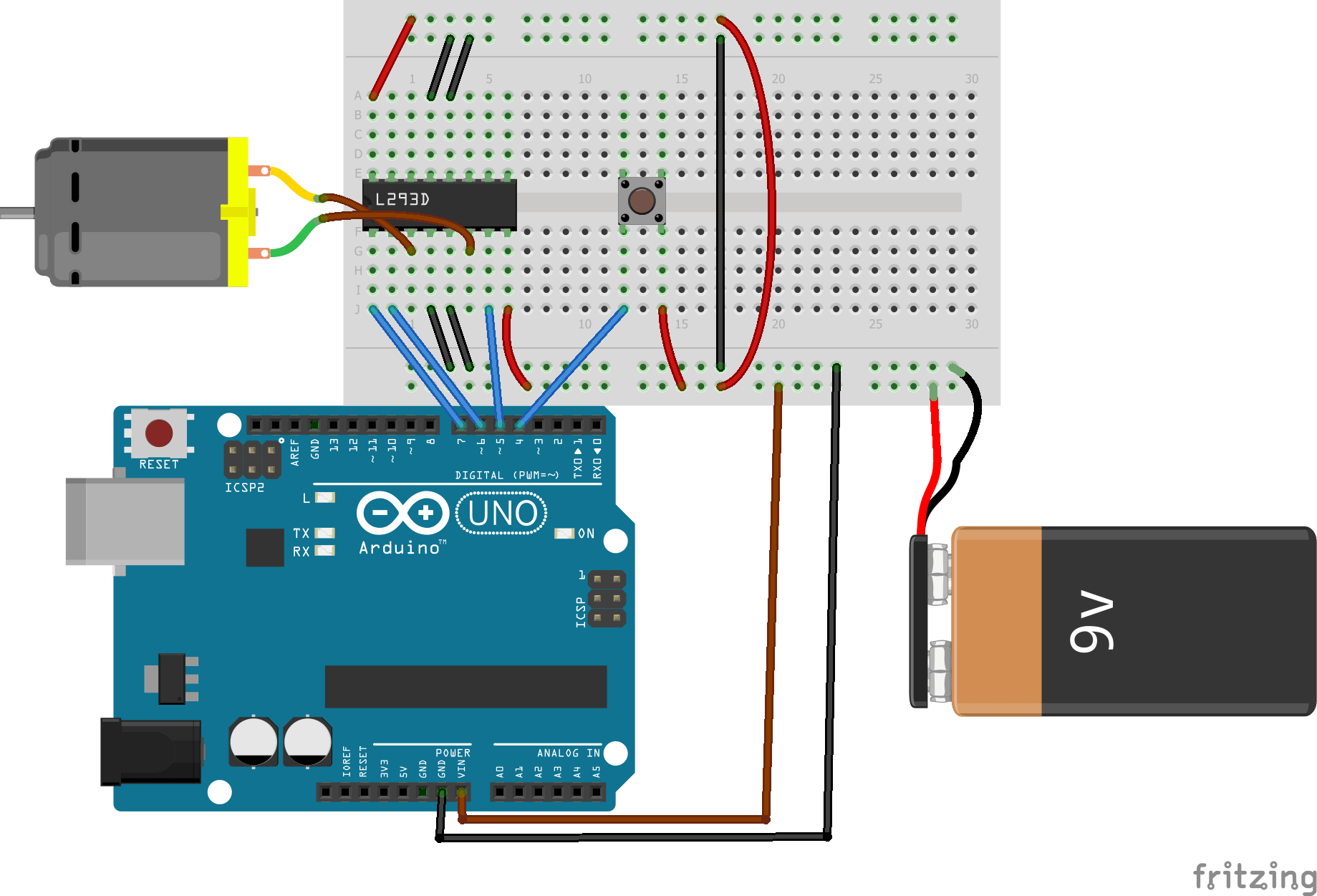
* Arduino-UNO
* Motor Driver – IC L293D
* DC Motors
* 9V Battery
* LED’s
* Battery Caps
* Resistor
* Wires and Connectors

Arduino-UNO : Arduino/Genuino Uno is a microcontroller board based on the ATmega328P ([datasheet](http://www.atmel.com/Images/doc8161.pdf)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.. You can tinker with your UNO without worring too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

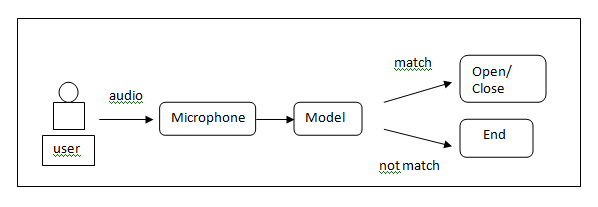
**Software Details:-**

* Python : Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.
* **OS**: Linux/Windows
* **Arduino IDE :** 
  + **Arduino IDE** is an open-source software, designed by Arduino.cc and mainly used for writing, compiling & uploading code to almost all Arduino Modules.
  + It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.
  + It is available for all operating systems i.e. MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role in debugging, editing and compiling the code.
  + A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, [Arduino Micro](https://www.theengineeringprojects.com/2018/09/introduction-to-arduino-micro.html) and many more.
  + Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code.
  + The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
  + The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.
  + This environment supports both C and C++ languages.

**4.Figure of Project :**



**5.Flow Chart:**

****

**6. Results:**

The results we got after using various machine learning algorithm.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No.** | **Model** | **Training Accuracy %** | **Testing Accuracy %** |
| 1 | LogisticRegression | 86.70% | 83.60% |
| 2 | SVC | 100.00% | 71.06% |
| 3 | RandomForestClassifier | 97.66% | 97.43% |
| 4 | Neural Network | 97.10% | 99.04% |

Figure 11: Results

# And the we select the best model which is Neural Network it give accuracy of 97.10% on training data and accuracy is increase to 99.04% on test data.

**6. Conclusion and Future Work:**

So at last we choose model which was built by using K-nearest neighbours after hyper parameter tuning due to its accuracy and giving best result on test data for prediction of health disease in patient.

**Future Road map of the Project:-**

1. Connect it with cloud based server and create database of user .
2. It works on multiple voices at same time.
3. Make it more secure by innovating new features of sound.

**7.Team Member details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S. No. | Name | Roll. No. | Mob. No. | Email ID |
| 1. | Aman panchal | 17EAICS019 | 9783253920 | panchalaman2000@gmail.com |
| 2. | Aakash yadav | 17EAICS001 | 9024361966 | aakashyadav112000@gmail.com |
| 3. | Ashish meena | 17EAICS33 | 9079465747 | ashishmeena479@gmai.com |
| 4. | Ajay joshi | 17EAICS010 | 9001575895 | ajayvkumar1997@gmail.com |
| 5. | Dheeren divya | 17EAICS013 | 9199566196 | Dheerenxd903@gmail.com |

**References**

[1] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe,K. Hansen, and K.-R. Muller, “How to explain individual clas-¨sification decisions,” Journal of Machine Learning Research,vol. 11, no. Jun, pp. 1803–1831, 2010.

[2] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, “Unsupervisedfeature learning for audio classification using convolutional deep belief networks,” in Advances in Neural Information Processing Systems (NIPS), 2009, pp. 1096–1104.

[3] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed,N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath et al. , “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” IEEE  
Signal Processing Magazine, vol. 29, no. 6, pp. 82–97, 2012.

[4]M. Anusuya and S. K. Katti, “Speech recognition by machine;a review,” International Journal of Computer Science and Information Security, vol. 6, no. 3, pp. 181–205, 2009.

[5] M. Abadi et al., “TensorFlow: Large-scale machine learning  
on heterogeneous systems,” 2015, Software available from tensorflow.org.

[6]R. Grosse, R. Raina, H. Kwong, and A.Y. Ng. Shift-invariant sparse coding for audio classifi-  
cation. In UAI, 2007.

[7]H. Lee, C. Ekanadham, and A. Y. Ng. Sparse deep belief network model for visual area V2. In  
NIPS, 2008.

[8]Boureau Y-L, Ponce J, LeCun Y (2010) A theoretical analysis of feature pooling in visual recognition. In: Proceedings of the 27th international conference on machine learning, pp 111–118.

[9]Chen L, Mao X, Xue Y-L, Cheng LL (2012) Speech emotion recognition: features and classification models. Digital Signal Process 22(6):1154–1160.

[10]Goodfellow IJ, Bengio Y, Courville AC (2016) Deep learning (Adaptive Computation and Machine Learning series). MIT Press