Voice recognition Algorithm

R Rohith Reddy

26 June 2019

1 Objective

To know the math running behind the code of Voice recognition algorithm.

2 Introduction

Voice recognition, in this case classification is done using machine learning.Machine learning is an application where we train the systems to learn and improve the things from experience rather than explicitly programming it.
In voice recognition we give a voice input to the system where we will get an text output from it.For this we have to train the algorithm with many different voices for obtaining accurate results.
The algorithm that is used to recognize voice in the given code is Linear regression algorithm.

3 Steps involved

1. Conversion of given sample to mfcc format
2. Dividing the dataset into training and test sets
3. Training the Algorithm

4 Math behind the steps involved

4.1 Conversion of given sample to MFCC format

Mel Frequency Cepstral Coefficents (MFCCs) are a feature widely used in automatic speech and speaker recognition.
The Mel scale relates perceived frequency, or pitch, of a pure tone to its actual measured frequency.
The formula for converting from frequency to Mel scale is:

$$M(f) = 1125 * ln(1 + f/700)$$

To go from Mels back to frequency:

$$M^{-1}(m) = 700(exp(m/1125) - 1)$$

Steps involved to convert are:

1. Frame the signal into short frames.
2. For each frame calculate the periodogram estimate of the power spectrum.
3. Apply the mel filterbank to the power spectra, sum the energy in each filter.
4. Take the logarithm of all filterbank energies.
5. Take the DCT of the log filterbank energies.
6. Keep DCT coefficients 2-13, discard the rest.

4.2 Dividing the set

Just because a learning algorithm fits a training set well, that does not mean it is a good hypothesis. It could over fit and as a result your predictions on the test set would be poor. Hence we divide the given dataset into training set and test set.

4.3 Algorithm

The algorithm that is used for voice recognition in this code is Linear Regression.
Linear regression is a linear approach in modelling the relationship between a scalar response and one or more explanatory variables.
After conversion of voice sample into mfcc format 4044 features are generated.These are called inputs or features.
The input matrix is defined as 4044x1 matrix

$$X = [0\ 0\ 0.0012\ upto\ 4044\ elements]^T$$

Our objective is to determine whether the given voice input is forward,back,left,right or stop. Hence we take output as a 5x1 matrix.
For example,forward is represented as

$$y = [1\ 0\ 0\ 0\ 0]^T$$

Back is represented as

$$y = [0\ 1\ 0\ 0\ 0]^T$$

left is represented as

$$y = [0\ 0\ 1\ 0\ 0]^T$$

Similarly for right and stop.
The multivariable form of the hypothesis function accommodating these multiple features is as follows:

$$y' = sigmoid(W.X + B)$$

$$y'i = sigmoid(\sum_i Wi * xi + b)$$

where W is a 4044x5 weighted matrix, b is a 5x1 bias matrix.
Our goal is to find out the weighted matrix and bias matrix using the various inputs and

outputs.

This hypothesis is then given into sigmoid function to scale the value between 0 and 1 since our final output is in 0's and 1's.Sigmoid function is defined as:

$$sigmoid(x) = 1/(1 + e^{-x})$$

We can measure the accuracy of our hypothesis function by using a cost function. This takes an average difference (actually a fancier version of an average) of all the results of the hypothesis with inputs from x's and the actual output y's. This is represented as

$$J(W, b) = 1/2(\sum (y - y')^2)$$

If we are able to find the minimum value of this cost function then we can find the exact fit to the given data set.This can be done using gradient descent method.
Using gradient descent method we can find the W,b matrices for which we obtain minimum value of J.

4.4 Gradient Descent method

Upto now we have obtained the following equations:
Input:

$$X$$

Orginal Output:

$$y$$

Output:

$$y'$$

Hypothesis:

$$y' = sigmoid(W.X + B)$$

$$y'i = sigmoid(\sum_i Wi * xi + b)$$

Cost function:

$$J(W, b) = 1/2(\sum (y - y')^2)$$

Using gradient descent we have to find the values for W,b which minimizes cost function.The equations are represented as follows:

$$Wi := Wi - \alpha \frac{\partial J}{\partial Wi}$$

$$b := b - \alpha \frac{\partial J}{\partial b}$$

Partial differentiation terms are evaluated as follows:

$$\frac{\partial J}{\partial Wi} = \frac{\partial}{\partial Wi} 0.5 * (\sum (y - y')^2)$$

$$\frac{\partial J}{\partial Wi} = \frac{\partial}{\partial Wi} 0.5 * (\sum (y - sigmoid(\sum_i Wi * xi + b))^2)$$

$$\frac{\partial J}{\partial Wi} = \frac{\partial}{\partial Wi} 0.5 * (\sum (y - sigmoid(\sum_i Wi * xi + b))^2)$$

$$\frac{\partial J}{\partial Wi} = \sum (y - y') * x_i * (-sigmoidprime(y'))$$

where

$$sigmoidprime(x) = sigmoid(x) * (1 - sigmoid(x))$$

Similarly for b matrix coefficients:

$$\frac{\partial J}{\partial b} = \frac{\partial}{\partial b} 0.5 * (\sum (y - y')^2)$$

$$\frac{\partial J}{\partial b} = \frac{\partial}{\partial b} 0.5 * (\sum (y - sigmoid(\sum_i Wi * xi + b))^2)$$

$$\frac{\partial J}{\partial b} = \frac{\partial}{\partial b} 0.5 * (\sum (y - sigmoid(\sum_i Wi * xi + b))^2)$$

$$\frac{\partial J}{\partial b} = \sum (y - y') * (-sigmoidprime(y'))$$

The vector form for these gradient descent method are:

$$W := W - \alpha X^T \delta$$

$$b := b - \alpha \delta$$

where

$$\delta = \sum (y - y') * (-sigmoidprime(y'))$$

Hence from these equations W,b matrices are found.

4.5 Output

Since W,b is obtained we can find output using

$$y' = sigmoid(W.X + B)$$

One Epoch is when an entire dataset is passed forward and backward through the neural network only once.
In this code a total of 10 epochs are given which means one dataset is trained 10 times.

5 Conclusions

1. The accuracy obtained through this linear regression algorithm is around 80-90%.

2. Increasing the number of training sets and increasing the number of speakers gives higher accuracy than obtained.

3. Using gradient descent method we found the weighted and bias matrices to minimize the cost function.

4. We can also use normal equation method to find out the weighted matrix coefficients but the number of features are large in number therefore the computational time taken is more in this method.

5. Since this is a classification problem ,logistic regression algorithm or neural networks may give more accurate results.

6. Sigmoid function is used to scale the obtained output between 0 and 1.