POLITECNICO DI TORINO

ICT for Health

Report Laboratory 4

Professor: 4/5
Monica Visintin

Student:
Iman Ebrahimi Mehr S250190

A.Y.2019-2020

Contents

1   Introduction

Parkinson's disease (PD) is the most common age-related motory neurodegenerative disease, initially described in the 1800's by James Parkinson as the 'Shaking Palsy'. Motor symptoms in PD are tightly linked to the degeneration of substantia nigra dopaminergic neurons and their projections into the striatum; also it's characterized by resting tremor, rigidity, akinesia and bradykinesia. The sick neurons project to other structures in the basal ganglia causing the loss of function of the latter that is involved in the coordination of the body movement.

Unified Parkinson's Disease Rating Scale (UPDRS) is a worldwide scale used to evaluate the progression of the disease. The UPDRS is composed of four subscales and each of them contain some items (42 in total), which assess impairment related to the PD. One of the features used in the UPDRS is quality of voice; in fact, patients affected PD cannot speak correctly since they cannot control the vocal cords and the vocal tract (note that this isn't always true because PD not always affects voice). More in particular, the speech symptoms related to Parkinson's Disease are: overall loudness level is reduced; rate of speech becomes too slow or too fast; difficulty initiating speech or inappropriate pauses; voice is tremulous and monotonous; hoarse/breathy vocal quality; articulatory effort is reduced or imprecise. For all these reasons, also the pronunciation of a vowel for a long period can be used to understand if the patient is affected by PD or not since the pronunciation of a vowel is obtained by pumping air out of the lungs, while the vocal cords periodically open and close the vocal tract.

The aim of this laboratory is to develop an algorithm by using Hidden Markov Model (HMM) in Matlab that can classify patients as healthy or ill by analysing their vocal samples: one of the most common symptoms of the disease is the inability of keeping the vocal tract opened, due to muscular contraction.

2   The Algorithm

2.1   The data set

The data-set is made of 20 '.wav' recordings of patients pronouncing the "a" vowel for a certain amount of time (according to the patients' muscular possibility). The recordings have been made with a mobile phone at a sampling frequency of 8 kHz.

2.2   Preprocessing

For each audio files, by using the Fourier transformation of the voice signal which is characterized by several peaks, the fundamental frequency of the signal is found f_0 that is the frequency of vibration of the vocal cords. After that, each period $(1/f_0)$ has been interpolated taking N_S-1 samples (note that the analyzed signals are not exactly periodic). In this way, from a time-continuous signal $x(t)$, a discrete time signal $x[n]$ is obtained with period N_s. Then all the signals have been quantized using N_Q values (called representative levels) and, in this way, the signal $x_Q[n]$ is obtained. In order to find the optimum value for N_Q, some signals have been used for training the K-means algorithm (separately for HC and PD voices) which is already implemented in Matlab, that is a clustering algorithm, that has as output the N_Q centroids of the given input (during the quantization, each sample of $x[n]$ is given to the cluster whose centroid is at minimum distance).

meaning!

2.3 Using HMM

A Finite State Machine (FSM) is an abstract machine that is in exactly one of a finite number of states at any given time which could change from one state to another in response to some external inputs. Changing from one state to another is called a transition and an emission/output corresponds to each states. Starting from a sequence of emissions/outputs without knowing the sequence of states, the model went through to generate them, the **Hidden Markov Model (HMM)** can be used to recover the sequence of states from observed data.

In this laboratory two HMMs are used, one trained with HC signals and the other with PD signals. Then, a new signal is given to both the HMMs and as output, the probability that the signal belongs to that HMM is given. In this way, looking at the maximum probability we can classify a signal as being pronounced from an healthy person or from an ill person, so we can classify a voice as pathological or not.

The idea is that, if the signals are exactly periodic, then a signal can be generated by a FSM with NS states such that, from state k, the machine moves to state k+1 and when the machine is in state k it outputs the quantized level k. Since the signals are not exactly periodic, it is necessary to train the FSM that has NS states and NQ output values in order to get the true transition and emission matrices. These matrices fully identify and describe a HMM which are generated as:

- *The emission matrix* **B** is randomly generated with dimensions of $N_s \times N_q$. Data along the rows are then normalized;

- *The transition matrix* **A** is a $N_s \times N_s$ and is generated in two different ways: randomly (normalized rows) and circularly.

Note that, two different initial random transition matrices for each HMM are used, the first one is a circular matrix and the other is generated randomly. By using different initial matrices, different HMM are created during the training phase so that in the end four different HMMs exists, two for PD signals and two for HC signals.

Finally by looking at the final results, specificity and sensitivity for both training dataset and testing dataset can be calculated. To evaluate the specificity, the algorithm compares the probability of the true negative with the probability of the false positive, while the evaluation of the sensitivity is carried out by comparing the probability of the true positive with the probability of the false negative.

2.4 Results

The algorithm has been implemented using three different sets for training and testing, as a sort of 3-fold cross validation:

- samples 1-7 as training and 8-10 as testing

- samples 4-10 as training and 1-3 as testing

- samples 1-3, 7-10 as training and 4-6 as testing

Several trials have been made by changing the parameters and all the following results are shown in table [**1**].

	Train		Test	
Parameters	**Specificity**	**sensitivity**	**Specificity**	**sensitivity**
$N_s = N_Q = 5$	90.45	80.95	66.67	55.56
$N_s = N_Q = 7$	95.24	90.47	66.67	88.89
$N_s = N_Q = 8$	100	100	66.67	84.13
$N_s = N_Q = 9$	100	95.24	77.78	88.89
$N_s = N_Q = 11$	100	100	66.67	77.74
$tolerance = 10^{-2}$	80.95	100	77.78	88.89
$tolerance = 10^{-1}$	90.45	80.95	55.56	66.67
$iterations = 150$	100	100	60.67	77.78
$iterations = 250$	100	100	70.67	100

Table 1: Percentage values of specificity and sensitivity over training and testing sets.
To play with N_s and N_Q we set initial parameters to $tolerance = 10^{-3}$ and number of iterations $= 200$
To play with $tolerance$ and and number of iterations, we kept $N_s = N_Q = 8$ as default value of HMM.

A difference can be seen looking at the specificities and sensitivities for the different values of $N_s = N_q$. A small number of states gives poor results in comparison to the ones obtained with bigger values. At the same time it can be seen that values larger than $N_s = 8$ don't improve significantly the specificity or the sensitivity, actually they start to decrease. In the other hand as the tolerance increases, the specificities and sensitivities obtained tend to decrease slightly.

3 Conclusions

As it can be seen from the results the table [1], good values for both sensitivity and specificity have been obtained. This means that the prediction 'healthy' or 'ill' is often correct; however, the dataset used in this laboratory is really small. The small variations are due to the poor data availability, thus training is not efficient. Moreover, there are some warnings on the used parameters: the number of quantization level N_Q and the number of samples N_s are subject to the maximum number of iterations used in the pre-process phase; sometimes it is not possible to have a given tolerance during the training of a HMM due to constraint of maximum number of iterations.