University of Groningen
Short Programming Project

Gionanidis Emmanouil

March 2, 2018

Introduction

In our society which is characterized by the huge progress of technology, we can elaborate in the part of how this evolution interact with us and help us especially in scientific fields such as Sociology , Biology , Health and Medicine, Robotics etc.

We reached a point that technology is an indivisible part of our daily life particularly in this project the main aspect of technology concetrates in the domain of Health and Medicine.Plenty of techniques are used in this field to automate procedures such as surgeries , diagnosis etc or to help the people who are working on this aera to make better decisions based on their advises. Furthermore to upgrade the services which are provided to the patients.

Main idea

The initial step of the project constitute from the extraction of special features of patients' voice signals with vary techniques. Based on the evidence that these features can provide us , a classifier has been made to categorize people , as a first step , between Male , Female .

In the end and after evaluating every procedure and trying different techniques as concern the extraction and the classification stage , the result is a classifier that distinct people between four categories : Healthy-Male , Healthy-Female , Not-Healthy-Male , Not-Healthy-Female.

Scope and delimitation of the project

Digital signal processing (DSP) is the use of digital processing,such as by computers,to perform a variety of signal processing operations. Digital signal processing and analog signal processing are subfields of signal processing[1-3]. DSP applications include audio and speech processing, sonar, radar and other sensor array processing, spectral density estimation, statistical signal processing, digital image processing, signal processing for telecommunications, control systems, biomedical engineering, seismology, among others. Using the Mel-frequency cepstral coefficients (MFCCs) which is a procedure based on DSP the feature extraction of patient's voice signals takes place[5-6]. This procedure is provided by a library in Python.
The documentation of the procedure and the implementation are the following:

- 1. $http://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html$
- 2. $https://github.com/jameslyons/python_speech_features$

The other DSP technique that used is the Linear predictive coding (LPC) , the procedure in particular :

- 1) The basic idea of LPC says that the speech sample can be approximated as linear combination of past scpeech samples. The initial step denoted as pre-emphasis is to boost the amount of energy in the high frequencies.Because the high frequencies are weak in speech signal. This drop between the high-low frequencies called spectral tilt. The previous procedure takes place with a low-pass filter $y[n] = x[n] - a \times x[n-1]$, $a = 0.97, a > 0$ for low-pass filters, most common value of a=0.95.
- 2) We dont want to extract our spectral features from an entire uttrance or conversiation, because the spectrum changes very quickly. We can say that speech is a non-stationary signal , meaning that it's properties are not constant across time. Instead we want to extract spectral features from a small window , for which we can make the assumption that it is stationary. Because of this we split the signal into N frames constitute from samples.
- 3) This phase is called windowing and it is using a window which is non-zero in the specific region, this window is going though the singal at each frame. One of the most commonly used windows is Hamming window : $W[n] = 0.54 - 0.46 \times cos((2 \times \pi \times n)/(frameSize - 1))$ which shrinks the values of the signal toward zero at the window boundaries avoiding discontinuities.
- 4) After we have the signal pre-emphasised and every frame windowed we can start the autocorrelation procedure , autocorrelate every windowed frame by itself. The resulting autocorrelation matrix is Toeplitz and can be readily solved using standard matrix solutions.
- 5) The last step called LPC analysis . Convert each frame of autocorrelation matrix into LPC parameter set using Levinson-Durbin's method. This method is implemented in Python from SciPy library:
  - a) $https://docs.scipy.org/doc/scipy-0.17.0/reference/generated/scipy.linalg.solve_toeplitz.html$

  * For both of the feature extraction techniques the KALDI tool used to check the results of the procedures[4].

Classifier implementation:

Reading an input file with the following format

feature1,feature2,feature3.feature3,feature4.feature5.feature6,feature7.feature8,feature9,feature10,feature11,feature12

The first 13 features are from the voice signal feature extraction , and the state is at first step to denote Male,Female and then health situation of the individual(Healthy, not-Healthy).The data provided from the data base : The Massachussets Eye and Ear Infirmary Dataset(MEEI-Dataset)[7].

Initial approach :

Six machine learning algorithms used in this stage,

- 1. Logistic Regression
- 3. Linear Discriminant Analysis
- 2. K-Nearest neighbors Classifier
- 4. Decision Tree Classifier
- 5. Gaussian Naive Bayes
- 6. Support vector machine

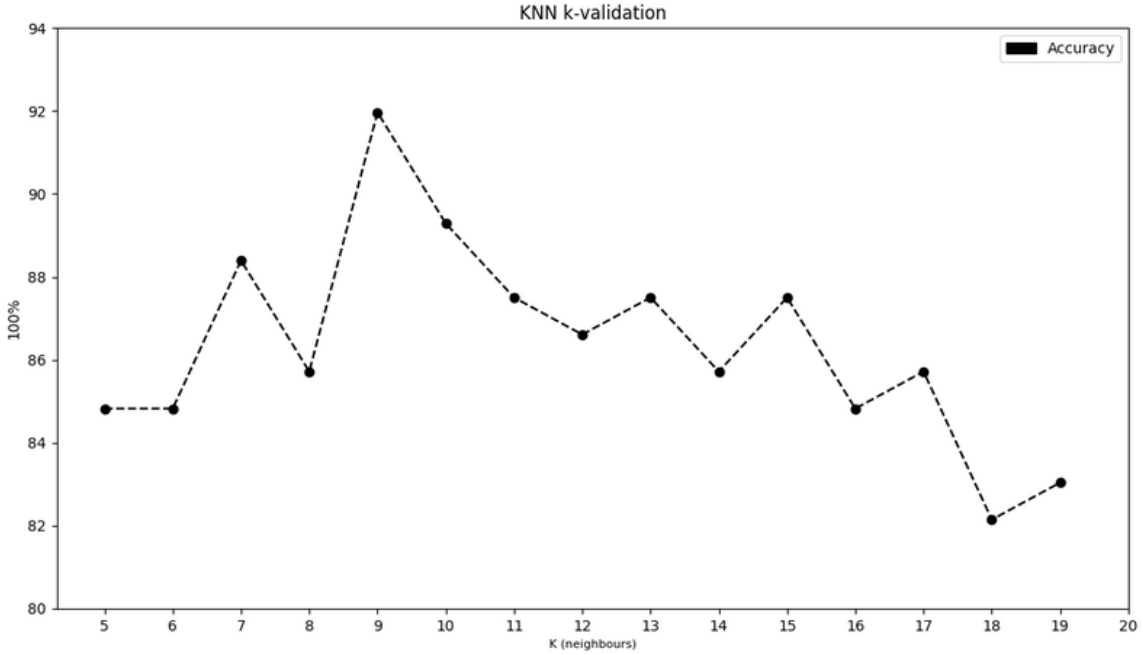
The usage of this stage is not about the algorithms , is about to show the difference between the k-fold cross validation and leave-one-out cross validation. The program ask the user to give as an input the number of the iterations (input: n times). The description concerns one time . After splitting the data and train(training stage: the ratings are 80algorithm , we evaluate the algorithms using 10-fold cross validation and leave-one-out cross validation seperatly and measuring the execution time. This happens n times as the user asked , before every iteration the data is been shuffled . As an output we have :

- 1. The accuracy of every algorithm n times in k-fold and leave-one-out cross validation
- 2. Every algorithm's execution time
- 3. The mean value of all the accuracies and times that every algorithm had

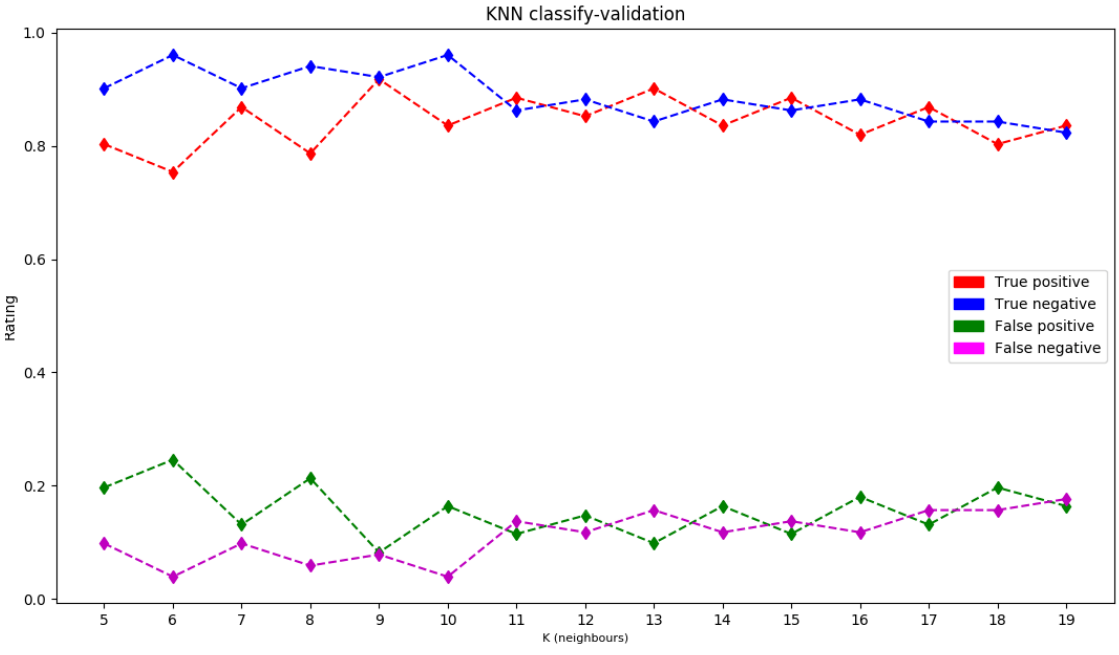
Continuing into , an elaboration in two of the previous algorithms :

K-Nearest Neighbors:

Using a range of neighbors between 5-20 in order to check the optimal number of neighbors measuring the accuracy , and find the true positive , false positive , false negative , true negative rates for every iteration[8].

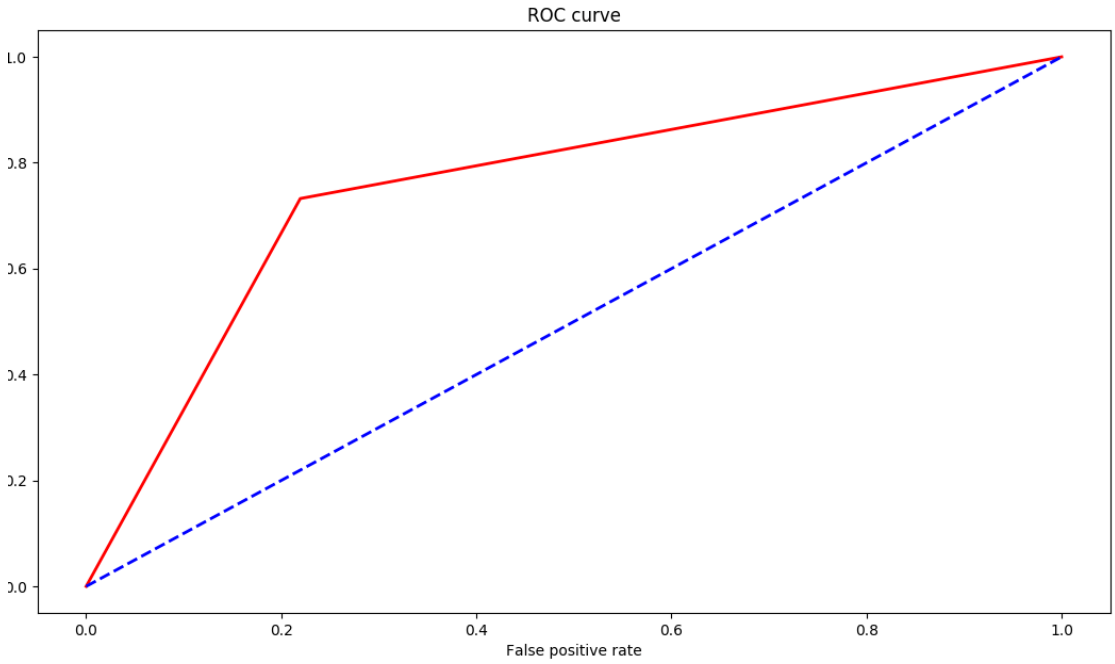


Detailedly true positive is : Assume that our patient is indeed not-healthy and the classifier classifies him as not-healthy . So the test was positive about the patience condition and the result was true because we know that the patient is not-healthy. We have to measure this rates because it is very important to know in a two-class problem which class had bigger rate in miss classification.

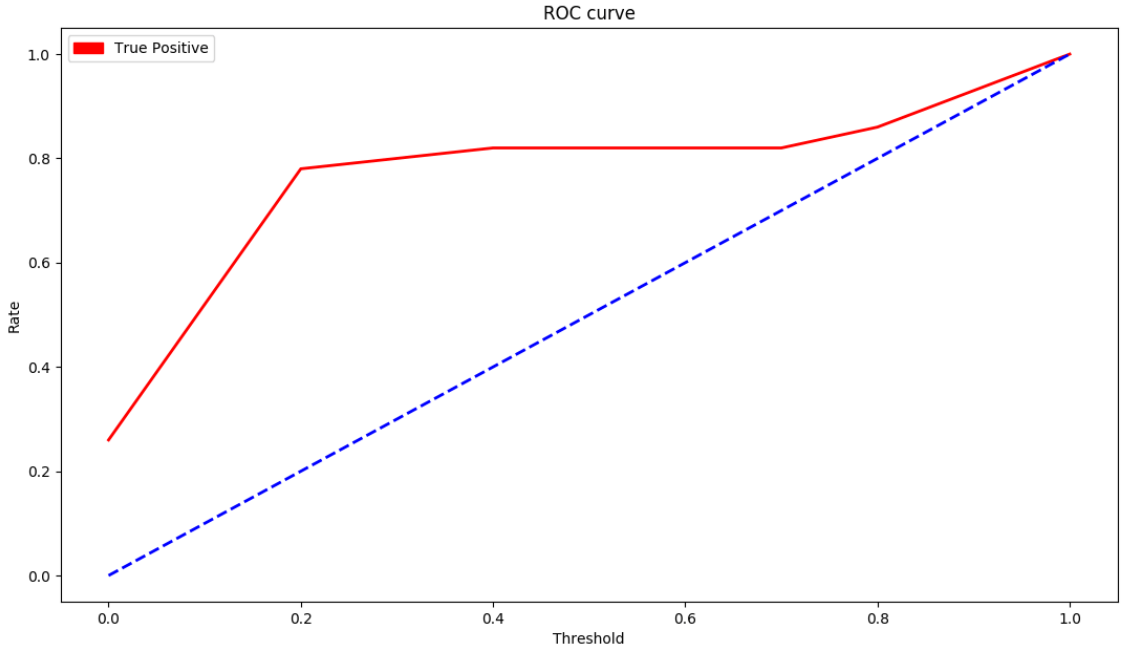


Logistic Regression:

Two implementation for this algorithm . The first is with the build in function that Python has which determines the threshold of the algorithm , and the built in function about the class weight which balance the weight of the two classes .



The second implementation had vary thresholds [0.2 , 0.3 , 0.4 , 0.5 , 0.6 , 0.7 , 0.8]. In both implementations the size of training and testing was 20-80 accordingly and ROC curve measured as well.



Gaussian Mixture Model:

Because the GMMs are prone to over fitting on small databases and do not generalize well to held out test data , the usage of them was only experimental . We can say that K-means appear poor performance for many real world situations . In this part I took a look at GMM which can be as an extension of the idea behind K-means, but can also be a powerful tool for estimation beyond simple clustering. A GMM attempts to find a mixture of multidimensional Gaussian probability distributions that best model any input dataset. In the simplest case, GMMs can be used for finding clusters in the same manner as K-means, and thats how GMM used in the specific case.

References

[1] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing.* Upper Saddle River, N.J.: Pearson Education-Prentice Hall, 2001.

[2] J. R. Deller, J. H. L. Hansen, and J. G. Proakis, *Discrete-Time Processing of Speech Signals.* New York, N.Y.: Wiley-IEEE, 1999.

[3] L. R. Rabiner and R. W. Schafer, *Theory and Applications of Digital Speech Processing.* Upper Saddle River, N.J.: Pearson Education- Prentice Hall, 2011.

[**4**] Daniel Povey, Arnab Ghoshal,Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann,Petr Motlcek, Yanmin Qian, Petr Schwarz, Jan Silovsk, Georg Stemmer, Karel Vesel ,Microsoft Research, USA,Saarland University, Germany,deCentre de Recherche Informatique de Montreal, Canada,Brno University of Technology, Czech Republic,SRI International, USA,Go-Vivace Inc., USA,IDIAP Research Institute, Switzerland,Tsinghua University, China,Technical University of Liberec, Czech Republic,University of Erlangen-Nuremberg, Germany, ,The Kaldi Speech Recognition Toolkit, https://homepages.inf.ed.ac.uk/aghoshal/pubs/asru11-kaldi.pdf

[**5**] Ganesh B. Janvale, and Ratnadeep.R.Deshmukh, Speech Feature Extraction using Mel-Frequency cepstral coefficient(MFCC)

[**6**] Wei-Ning Hsu, Yu Zhang, and James R. Glass, Unsupervised Domain Adaptation for Robust Speech Recognition via Variational Autoencoder-Based Data Augmentation,2017, http://arxiv.org/abs/1707.06265.

[**7**] Voice and Speech Laboratory, Massachusetts Eye and Ear Infirmary, Boston MA,Voice Disorders Database, 1.03 edition, 1994, Kay Elemetrics Corp.

[**8**] Mai Shouman, Tim Turner, Rob StockerApplying k-Nearest Neighbour in Diagnosing Heart Disease Patients , (LACSIT Press 2012).