Machine Learning Nano Degree
Gender Recognition of Voice

Jiacheng Shen

April 27, 2018

Contents

1 Definition

1.1 Project Overview

Voice recognition has a long history. In 1952, Bell Lab designed a system to recongize single digit. Then, great progress was made by methods like Linear Predictive Coding, Dynamic Time Warp and Hidden Markov Model. Nowadays, Machine Learning models like RNN are applied.[5] For example, the famous Chinese company iFLYTEK can reach an accuracy of 98%.[2]

In this project, the goal is to build a model to recongize the gender of voice. The dataset can be found on Kaggle[3]. The raw wave files have been extracted features by R. There are 1584 female samples and 1584 male samples.

1.2 Problem Statement

This project is a **suprivised binary classification** problem. The voice can be classfied to female or male. The goal is to use the given features of voice, build a model and recongize the gender of voice.

1.3 Merrics

Since the dataset is balanced (half female and half male) and the importance of both genders are the same, so recall and precision rate mean nothing. Therefore, accuracy is chosed to evauate the model. What's more, training and predicting time is also considered.

1. **Accuracy**. Accuracy counts how many samples are predicted correctly.

$$\text{Accuracy} = \frac{\sum \text{Correctly predicted}}{\sum \text{All Samples}} \times 100\%$$

2. **Time**.

2 Analysis

2.1 Data Exploration

The dataset is downloaded from Kaggle. There are 22 features and 3168 samples. All the features are numbers. No missing features. It is a balanced dataset with 1584 females and 1584 males.
As shown in Table 1, there are 22 features about the voice and the last one is the label, which need to be predicted female or male.

Category	Names						
Frequency	meanfreq	sd	median	Q25	Q75	IQR	
Spectrum	skew	kurt	sp.ent	sfm	mode	centroid	peakf
Fundamental Frequency	meanfun	minfun	maxfun				
Domain Frequency	meandom	mindom	maxdom				
Range	dfrange	modindx					
output	label						

Table 1: features

2.2 Algorithms and Techniques

I use sklearn package from python to do this project. The features of dataset are all numbers, so no encoding is needed. The label has only two cases, so converting it into a binary number is enought. Some features may have some skewness, so the log transformation may need to be applied. Normalization should be implemented, too. I want to try LogisticRegression, DecisionTree, RandomForest and SVM . After choosing the model with the best performance, I will use GridSearch to optimize this model.

2.3 Benchmark

Using a simple model like LogisticRegression(shown in the jupyter notebook of the repo), I can achieve accuarcy of 90.96% on the training dataset and 90.38% on the testing dataset. After model selecting and parameter tuning, there should be higher performance, so I want to achieve bias and variance as followings:

1. Bias: The ideal model should predict each voice to the right gender. I set the accuarcy threshold as 97.5% on the **Testing** dataset.

2. Variances: The good model shoule perform similar on different dataset to avoid over-fitting. I limit the difference between Training and Testing dataset to less than 1%.

3 Methodology

3.1 Data Preprocessing

The features are numbers so no one-hot encoding is needed. The label has only two values, 'female' and 'male', so encoding it into binary is enough. If some features have kind of skewness, implementing log transformation on them. The ranges of features may differ from each other, so normalization should be implemented.

3.2 Implementation

I will try LogisticRegression, DecisionTree, RandomForest and SVM with default parameters. And then compare them on aspects of **Training Time**, **Predicting Time**, **Training Accuracy** and **Validation Accuracy** to find the most balanced one.

3.3 Refinement

I will use GridSearch to tune the parameters.

4 Application

I want to deploy a web service to recongize the uploaded voice. I choose Python as the main language to implement it. After searching, I will use Flask[1] as the framework, use rpy2[4] to extract features from raw voice by R packages.

5 Reference

References

[1] flask. Flask (a python microframework). http://flask.pocoo.org/.

[2] iFLYTEK. iflytek open platform. http://www.xfyun.cn/.

[3] Kaggle. Gender recognition of voice. https://www.kaggle.com/primaryobjects/voicegender.

[4] rpy2. R in python. https://rpy2.bitbucket.io/.

[5] Wikipedia. Speech recognition. https://en.wikipedia.org/wiki/Speech_recognition#History.