Gender Recognition by Voice

Nowadays, chatbot becomes very common, Google, facebook and Microsoft are developing chatbot system for their clients, to provide the best user experience to the user of client. Moreover, many robot companies also developing mechanical waiters to replace staff. The market of automatic customer service is increasing, the suppliers are improving the user experience to increase the competitiveness.

This project focuses on the ‘Gender Recognition by Voice’, can the robot recognizes the gender of user?

Data Introduction

“This database was created to identify a voice as male or female, based upon acoustic properties of the voice and speech. The dataset consists of 3,168 recorded voice samples, collected from male and female speakers. The voice samples are pre-processed by acoustic analysis in R using the seewave and tuneR packages, with an analyzed frequency range of 0hz-280hz (human vocal range).” [1]

#	Column	Detail
1	meanfreq	mean frequency (in kHz)
2	sd	standard deviation of frequency
3	median	median frequency (in kHz)
4	Q25	first quantile (in kHz)
5	Q75	third quantile (in kHz)
6	IQR	interquantile range (in kHz)
7	skew	skewness (see note in specprop description)
8	kurt	kurtosis (see note in specprop description)
9	sp.ent	spectral entropy
10	sfm	spectral flatness
11	mode	mode frequency
12	centroid	frequency centroid (see specprop)
13	meanfun	average of fundamental frequency measured across acoustic signal
14	minfun	minimum fundamental frequency measured across acoustic signal

15	maxfun	maximum fundamental frequency measured across acoustic signal
16	meandom	average of dominant frequency measured across acoustic signal
17	mindom	minimum of dominant frequency measured across acoustic signal
18	maxdom	maximum of dominant frequency measured across acoustic signal
19	dfrange	range of dominant frequency measured across acoustic signal
20	modindx	modulation index. Calculated as the accumulated absolute difference between adjacent measurements of fundamental frequencies divided by the frequency range
21 (y)	***label***	***class, male or female***

Therefore, the project is going to recognize the gender by the voice measurement (column 1 - 20).

Research

This project examines several to build up the notable models to identify the gender, such as classification tree, K-NN classification, naive Bayes classification, logistic regression and artificial neural network.

Data Preparation

Because different models have different assumptions, these require the different transformations of dataset. Therefore, this section prepare four types of dataset, original dataset, standardized dataset, log dataset and principal component (PC) dataset.

Original Dataset

[Original.r]

```
data = read.csv('voice.csv')
n_data = nrow(data)
…
levels(data$label)[levels(data$label)=="female"] <- 0
levels(data$label)[levels(data$label)=="male"] <- 1
…
set.seed([int])
id = sample(1:n_data,size=(n_data*0.65),replace=F)
train_data = data[id,]
test_data = data[-id,]

train_X = train_data[,-21]
train_y = as.factor(train_data[,21])
```

```
test_X = test_data[,-21]
test_y = as.factor(test_data[,21])
```

Standardized Dataset

[Stand.r]

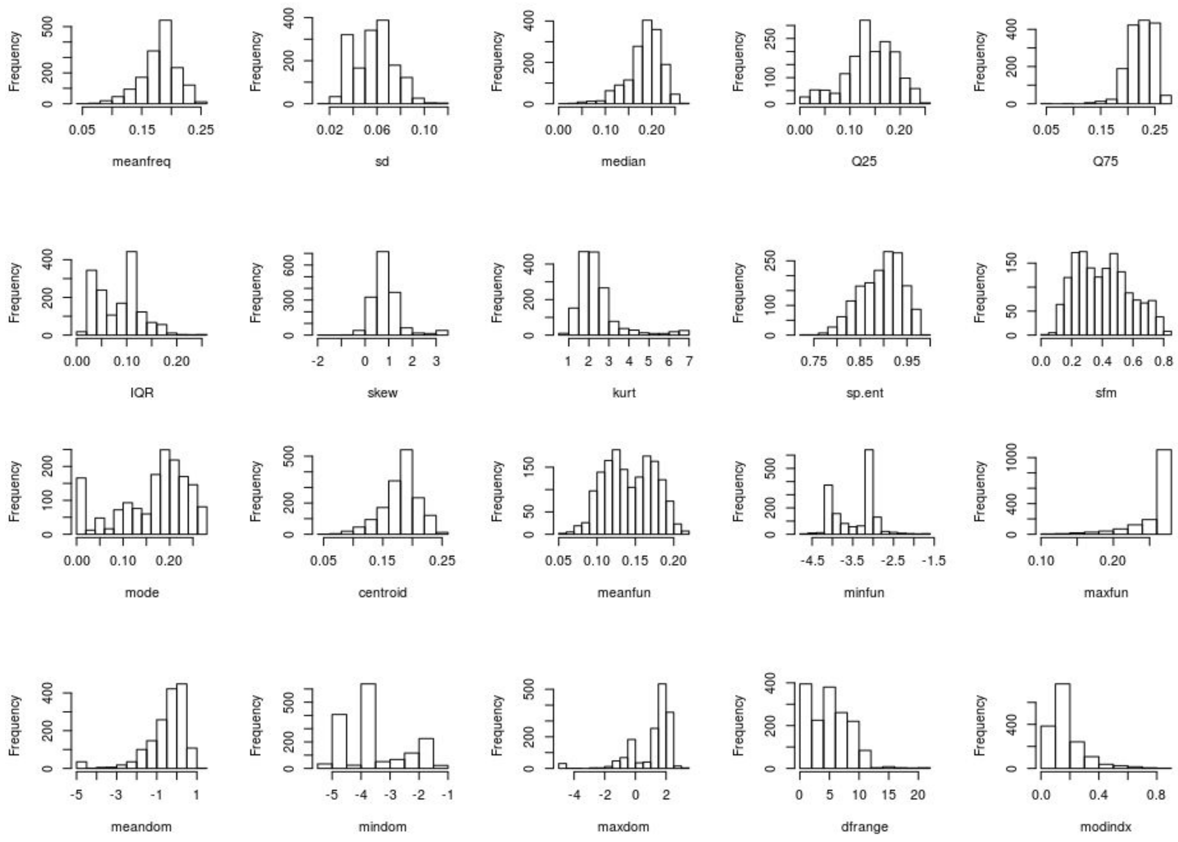
```
z_data = cbind(stand(data[,-21]),data[,21])
set.seed([int])
id = sample(1:n_data,size=(n_data*0.65),replace=F)
train_data = z_data[id,]
test_data = z_data[-id,]

train_X = train_data[,-21]
train_y = as.factor(train_data[,21])
test_X = test_data[,-21]
test_y = as.factor(test_data[,21])
```

Log Dataset

[Log.r]

```
par(mfrow=c(2,5))
for (i in 1:10) {
  hist(l_data[levels(data$label)==1,i],main=NA,xlab=names(data)[i])
}
par(mfrow=c(2,5))
for (i in 11:20) {
  hist(l_data[levels(data$label)==1,i],main=NA,xlab=names(data)[i])
}
```



Because some classification models assume that the variables are normally distributed, therefore we have to transfer the some variables by log, such as *skew, kurt, minfun, meandom, mindom, maxdom, dfrange, modindx*. However, dfrange and modindx contain zero, unable to be applied log transformation.

[Log.r]

```
l_data = data[,1:20]
for (i in 1:21) {
  if (any(i == c(7, 8, 14, 16, 17, 18))) l_data[i] = log(l_data[i])
}
l_data = cbind(l_data,data[,21])
…
set.seed([int])
id = sample(1:n_data,size=(n_data*0.65),replace=F)
train_data = l_data[id,]
test_data = l_data[-id,]

train_X = train_data[,-21]
train_y = as.factor(train_data[,21])
test_X = test_data[,-21]
test_y = as.factor(test_data[,21])
```

PC Dataset

PC analysis can reduce the dimension and compress the dataset which is suitable for this dataset as 20 variables. However, PC analysis will lose some information to reduce the accuracy.

[PC.r]

```
fit = prcomp(data[,1:20],scale=T,retx=T,center=T)
p_data = fit$x[,1:8]
…
set.seed([int])
id = sample(1:n_data,size=(n_data*0.65),replace=F)
train_data = p_data[id,]
test_data = p_data[-id,]

train_X = as.data.frame(train_data[,-21])
train_y = as.factor(data[id,21])
test_X = as.data.frame(test_data[,-21])
test_y = as.factor(data[-id,21])
```

Model with All Variables

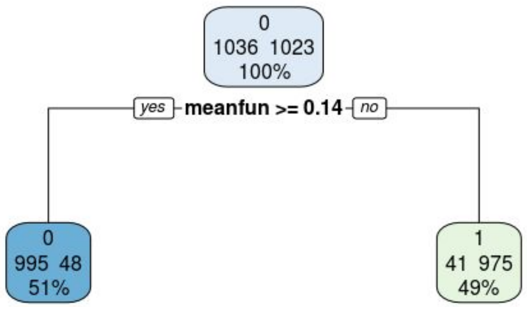
Classification Tree

Classification Tree bases on the binary splitting of variables, one at a time. The tree graph below shows the splitting depends on *meanfun* only, and it is reasonable as we always say female is high pitch, male is low. Log and standardized dataset have the same splitting and result, because the transformations did not loss the information.

[Original.r] (Stand.r | Log.r | PC.r)

```
ctree = rpart(train_y~.,data=train_X,method='class')
rpart.plot(ctree,extra=101)
```

```
...
original.ctree.train = calc_error(c1,train_y)
...
original.ctree.test = calc_error(c1,test_y)
```



[Output graph of original dataset]

F1 Score (Test)	Original	Log	Stand	PC
ctree	0.9468864	0.9468864	0.9468864	0.8848921

Because the PC analysis compresses the dataset, the PC dataset loses the some important information, the F1 score is lower than the others.

K-NN Classification

KNN is lazy learner and classifies the class by distance, therefore if the transformation can reduce the unbalance scale, the prediction will be improved. For example, standardized transformation rescales the variance; log transformation reduces the range of some variables and PC does the rescale in compressing. The F1 Score table shows that the all transformations improve the prediction. PC has lower score than Stand as some information are lost.

[Original.r] (Stand.r | Log.r | PC.r)

```
c1 = k_nn(train_X,test_X,train_y,test_y,v=10)
original.knn.test = calc_error(c1,test_y)
...
(Original) best k= 7  error rate= 0.2912534
(Stand) best k= 3  error rate= 0.02434626
(Log) best k= 5  error rate= 0.2182146
(PC) best k= 5  error rate= 0.0405771
```

F1 Score	Original	Log	Stand	PC
knn	0.6909091	0.7755102	0.9749768	0.957667

Naive Bayes Classification

Naive Bayes classification is another lazy learner, which is a probability model. The mean and standard deviation are requested in the probability measurement of continuous variables, therefore naive Bayes classification assumes the variables are normally distributed.

According to the assumption, the log transformation should be the best transformation, however some important columns contain zero value. PC gets the best result in naive Bayes classification as rebuilds the scale and becomes more normal.

[Original.r] (Stand.r | Log.r | PC.r)

```
nb = naiveBayes(train_X,train_y)
c1 = predict(nb,train_X)
original.nb.train = calc_error(c1,train_y)
```

F1 Score	Original	Log	Stand	PC
nb	0.8975332	0.9009524	0.8975332	0.9192661

Logistic Regression

Logistic regression is similar with the regression, we have to take out the insignificant variables to improve the model. All transformation do not have positive effect on the prediction. These model also simiply the dataset to lower dimension, Q25, Q75, kurt, sp.ent, sfm, meanfun, minfun (log model: +skew, maxdom, dfrange).

[Original.r] (Stand.r | Log.r | PC.r)

```
summary(glm(train_y~.,data=train_X,family=binomial))
lm = glm(train_y~Q25+Q75+kurt+sp.ent
        +sfm+meanfun+minfun-1,data=train_X,family=binomial)
names(train_X)[c(4,5,8,9,10,13,14)]

c1 = lm$fit>0.5
original.lm.train = calc_error(c1,train_y)

c1 = predict(lm,test_X[,c(4,5,8,9,10,13,14)]) > 0.5
original.lm.test = calc_error(c1,test_y)
```

F1 Score (Test)	Original	Log	Stand	PC
lm	0.98268	0.9772106	0.9817851	0.9455535

Artificial Neural Network

ANN mimic the functions and mechanism of our brain, to minimize the error by the back-propagation algorithm. The abnormal distribution might incur mistake, therefore transformation would improve the prediction.

[Original.r] (Stand.r | Log.r | PC.r)

```
set.seed(1155090847)
nn = ann(train_X,train_y,10)
original.nn.train = calc_error(round(nn$fitted.values),train_y)

c1 = round(predict(nn,test_X))
original.nn.test = calc_error(c1,test_y)
```

F1 Score (Test)	Original	Log	Stand	PC
lm	0.9725275	0.9665158	0.9655814	0.9576427

According to the table above, ANN model provides a convenient method to predict the class successfully (All F1 Scores > 0.95) without any manual adjustment. Moreover, all transformations do not have any positive effect on the prediction, which does not match our expectation.

Conclusion

F1 Score (Test)	ctree	knn	nb	lm	nn
Original	0.9468864	0.6909091	0.8975332	0.98268	0.9725275
Log	0.9468864	0.7755102	0.9009524	0.9772106	0.9665158
Stand	0.9468864	0.9749768	0.8975332	0.9817851	0.9655814
PC	0.8848921	0.957667	0.9192661	0.9455535	0.9576427

In this section, the logistic regression with original dataset and standardized dataset have the best F1 score, however other models would be improved by taking out irrelative variables.

Model with Relative Variables

According to the logistic regression in the previous section, the dimension of dataset would be reduced to 10 [Q25, Q75, kurt, sp.ent, sfm, meanfun, minfun (log: +skew, maxdom, dfrange)]. The following models are results with variable selection.

[ModelwithRelated/Original.r] (Stand.r | Log.r | PC.r)

```
data = data[,c(4,5,8,9,10,13,14,21)]
# data = data[,c(4,5,7,8,9,10,13,14,18,19,21)] # Log
```

F1 Score (Test)	ctree	knn	nb	lm	nn
Original	0.9468864	0.8152985 (+)	0.9362101 (+)	0.98268	0.9826167 (+)
Log	0.9468864	0.7609302 (-)	0.9187675 (+)	0.9772106	0.9782214 (+)
Stand	0.9468864	0.981685 (+)	0.9362101 (+)	0.9817851	0.9660239 (+)

After filtering irrelative variables, many models have positive effect, such as KNN, naive Bayes and ANN models, because some noises have been eliminated.

Conclusion

Three columns have been highlight at the table above as having the best testing F1 score (>0.98), KNN with stand, logistic regression and ANN models. I would like to suggest the logistic regression model with the original dataset as which is the simplest and accuratest model.

$$Pr(label = male) = -0.5327\, Q25 + 0.5101\, Q75 - 0.004712\, kurt + 0.2418\, sp.ent$$
$$- 8.232\, sfm - 0.01682\, meanfun + 0.3391\, minfun$$
$$if\ Pr(label = male) > 0.5,\ prediction = male\ else\ prediction = female$$

Reference

[1] Gender Recognition by Voice - Identify a voice as male or female, https://www.kaggle.com/primaryobjects/voicegender