Identifying Gender By Voice Recognition

05/04/2017

Kiyoon Jeong

STATS LEARN DS 01

Introduction

Nowadays, the voice recognition is really important business. For example, Apple's Siri, Samsung's Bixby, Google's Assistant, and Microsoft's Cortana are well-known voice recognition products. As the fourth industrial revolution is coming, this kind of automate techniques become core part that most of big companies invest lots of money to compete with the others. Because these automate techniques are based on statistical modeling, especially machine learning methods, we should think about how it works and how to make it available.

In daily life, we can guess people's gender by the voice. We can possibly infer other's gender through the voice's frequency. However, when it applied to machine, how can we predict the gender more accurately? Which variables affect the significant difference between male and female? My project's object is to describe the machine learning methods briefly, find out the important variables, and to identify the gender by using them.

Dataset

This 'voice' dataset is downloaded from the website called 'Kaggle'. The data is collected from the Harvard-Haskins Database of Regularly-Timed Speech, Telecommunications & Signal Processing Laboratory (TSP), VoxForge Speech Corpus, and Festvox CMU_ARCTIC Speech Database at Carnegie Mellon University. There are 1 response variable (label : male or female) and 20 predictive variables , and 3168 observations.

Description of Variables

meanfreq: mean frequency (in kHz)

sd: standard deviation of frequency

median: median frequency (in kHz)

Q25: first quantile (in kHz)

Q75: third quantile (in kHz)

IQR: interquantile range (in kHz)

skew: skewness

kurt: kurtosis

* kurtosis is the sharpness of the peak of a frequency-distribution curve

sp.ent: spectral entropy

*The spectral entropy is the measure of amount of disorders in a system, which means it contains some high - peak or low -peak.

sfm: spectral flatness

* Spectral flatness is a measure used in digital signal processing to characterize an audio spectrum.

mode: mode frequency

centroid: frequency centroid

peakf: peak frequency (frequency with highest energy)

meanfun: average of fundamental frequency measured across acoustic signal

*Fundamental frequency is the lowest frequency of a periodic waveform

minfun: minimum fundamental frequency measured across acoustic signal

maxfun: maximum fundamental frequency measured across acoustic signal

meandom: average of dominant frequency measured across acoustic signal

*Dominant frequency is the most often occurred frequency

mindom: minimum of dominant frequency measured across acoustic signal

maxdom: maximum of dominant frequency measured across acoustic signal

dfrange: range of dominant frequency measured across acoustic signal

modindx: modulation index.

*It is calculated as the accumulated absolute difference between adjacent measurements of fundamental frequencies divided by the frequency range

Prediction Through Various Methods

Base

This dataset has lots of variables (21 variables). I will use K-nearest neighbor , Ridge regression, LASSO, SVM (Support Vector Machine), CART (Classification and Regression Tree), and Random Forest to predict the gender. Each method has clear strong points, so I will briefly explain about the traits of methods. First, I splitted the data into two, train set and test set. I picked it randomly (0.75 for Train Set, 0.25 for Test Set), because the top half's label is male, and the bottom half's label is female.

K-nearest neighbor Classification

First, I used the k-nn classification. It is quite simple, and non-parametric method. The prediction method is that gathering k nearest observations and categorizing them into male or female by a majority vote. For example, if you use k = 1, each points' area will be categorized into male or female without vote. And if you use k = 10, 10 observations will be grouped and categorized into male or female by majority vote. The weak point is that because it uses the whole observations (3168 data) , it takes some time to process the data. This method usually well-perform, and because the our dataset's response variable is categorical value, this method is one of good choice. I choose the k as 1 ,3, 5, 7, 10, and 20. The results are following:

```
> table(knn1, test$label)

knn1   0   1
   0 368  16
   1  15 393
> mean(knn1 != test$label)
[1] 0.03914141
```

```
> table(knn3, test$label)

knn3   0   1
   0 373  23
   1  10 386
> mean(knn3 != test$label)
[1] 0.04166667
```

```
> table(knn5, test$label)

knn5   0   1
   0 371  26
   1  12 383
> mean(knn5 != test$label)
[1] 0.0479798
```

```
> table(knn7, test$label)

knn7   0   1
   0 366  29
   1  17 380
> mean(knn7 != test$label)
[1] 0.05808081
```

```
> table(knn10, test$label)

knn10   0   1
    0 369  32
    1  14 377
> mean(knn10 != test$label)
[1] 0.05808081
```

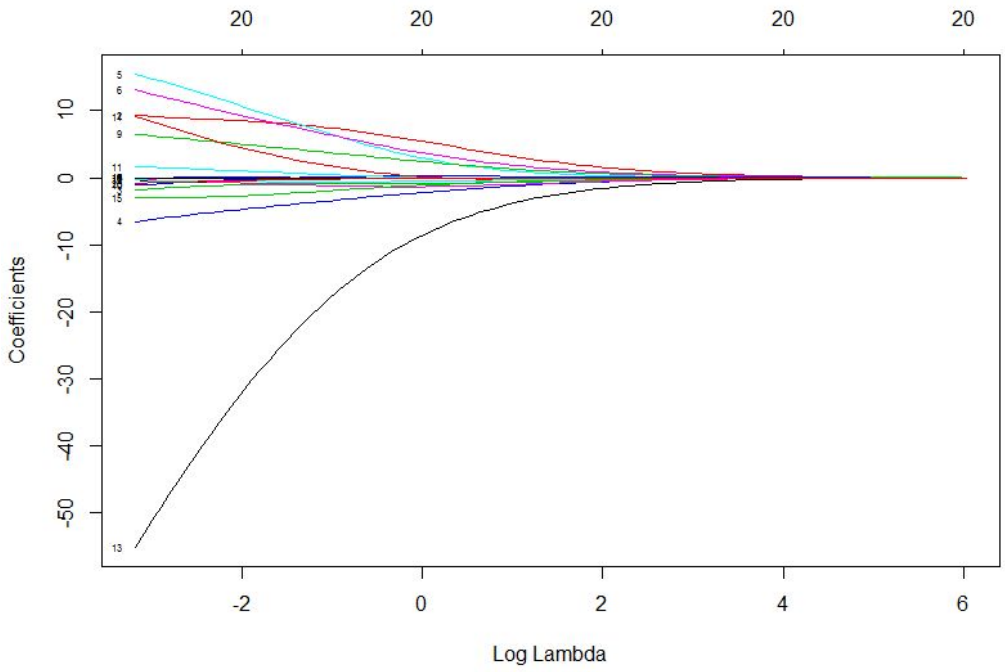
```
> table(knn20, test$label)

knn20   0   1
    0 367  42
    1  16 367
> mean(knn20 != test$label)
[1] 0.07323232
```

This table shows that as the number of k increases, the error rate also increases. However, our dataset has quite large number of observations (3168), so, if k is too small, it has an overfitting problem. Though the bias is small, the variance increase due to the bias-variance trade-off. Thus, for the new observations, knn1 might not predict more precisely than the knn 20.

Ridge Regression (L2-penalty)

For the kernel method, it was hard to control the bias and variance. Let's see the generalized linear models. First, I used the ridge regression. Ridge regression uses the lambda as regularized parameter. By using this lambda, we can adjust the bias to control the variance. If you use big lambda, the coefficients will be shrinked to 0. Thus, we should choose a proper lambda that allow us to distinguish

which coefficients are more important. However, this method does not remove any coefficients in middle, which means it is hard to find proper lambda.



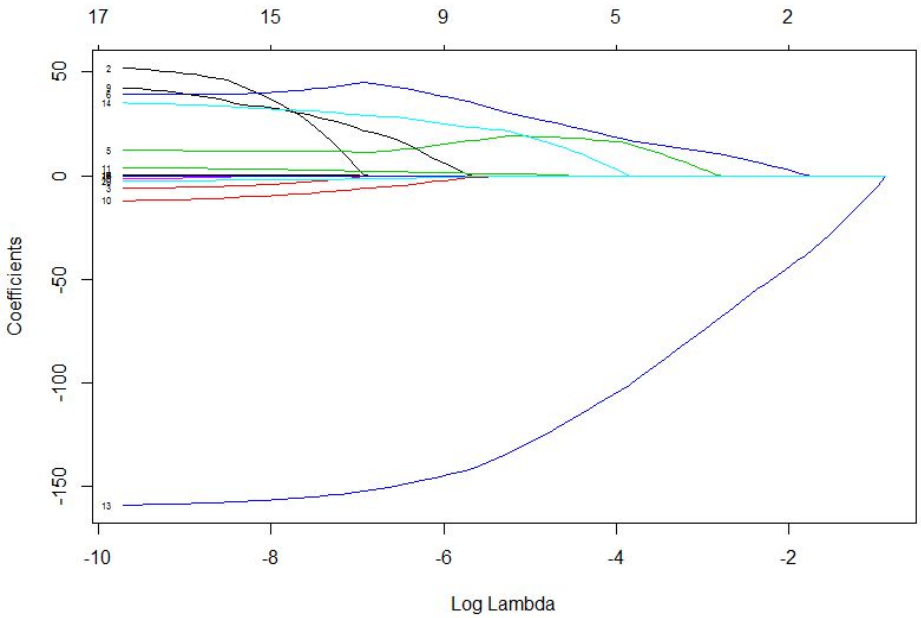
As you can see, it is hard to tell which lambda should be chosen to distinguish important coefficients properly. For me, I choose the lambda as 0 because at that point, except 6 coefficients (4 : Q25, 5 : Q75, 6 : IQR, 9 : sp.ent, 12 : centroid, 13 : meanfun), rest of them shrink to 0. It makes sense because people usually infer the gender by the tone. If they have low tone, gender might be male. And if they have high tone, gender might be female. Thus, Q25, Q75, and IQR could be important coefficients. The spectral entropy is the measure of amount of disorders in a system, which means it counts the high -peak or low -peak. Generally, female use more exaggerated expressions while talking. Comparably, male use less expressions and speak smoothly. Thus, spectral entropy also could be important. Centroid is the value of centroid frequency. It is different with mean and median of frequency. Centroid frequency is a good measure for the brightness of sound. The brightness of sound is the strongest distinctions between sounds. We define it by the existence of high-frequency content in a sound. Since female tend to have

high centroid frequency, it also makes sense. ‘Meanfun’ is the mean of fundamental frequency. Fundamental frequency is the lowest frequency of a periodic waveform. Because male tend to have lower fundamental frequency than female, it is also could be important variable. In conclusion, Q25, Q75, IQR, spectrum entropy, centroid frequency, and mean of fundamental frequency are important variables for this ridge regression.

The misclassification error rate is 0.03030303, which is much lower than k-nn classification error.

Lasso (L1-penalty)

Ridge regression has an interpretability problem. However, Lasso method can do both parameter shrinkage and variable selection automatically, which makes it much easier to interpret the plot. However, because it doesn’t have a closed form, it is much slower than Ridge regression. Though it is slow, it usually performs better than ridge regression.



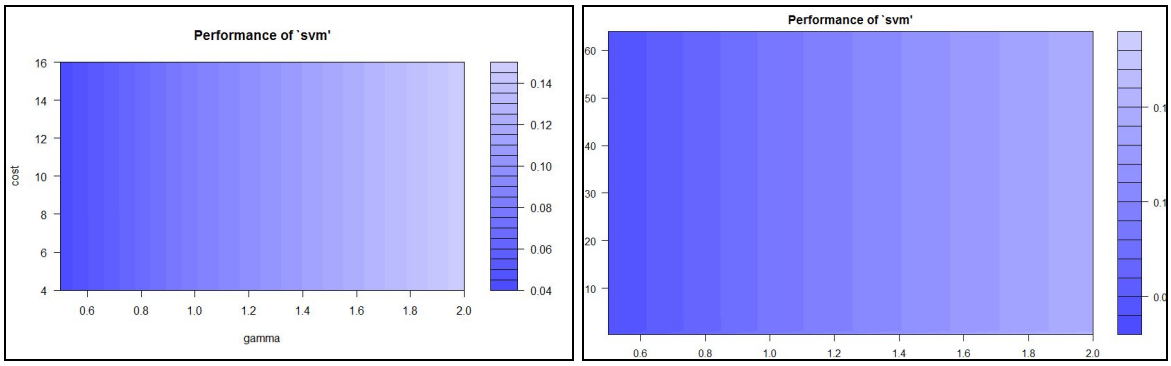
It is obviously more interpretable than ridge plot. So, if you choose log lambda -4, 4 coefficients (5 : Q75, 6 : IQR, 13 : meanfun, 14 : minfun) are remaining important. Like ridge regression, Q75, IQR, and meanfun are also important coefficients in this regression. For this method, minimum fundamental frequency became important, and centroid and spectrum entropy became less important in this model. The misclassification error rate is 0.02525253, which is much less than knn classification and ridge regression error rate.

SVM (Support Vector Machine)

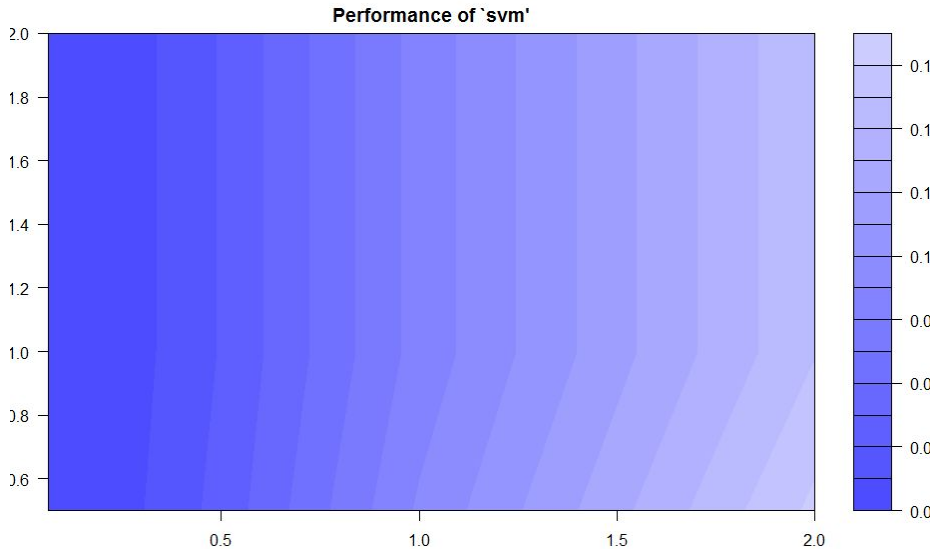
Support vector machine is also L2 shrinkage method, thus performance is similar to ridge regression. It uses hinge loss for training classifier and margin maximization to reduce the overfitting problem. First, I used the gamma (0.5, 1.0, 2.0) and cost (4, 8, 16) to compare the data for finding out the best performance.

```
Parameter tuning of 멎vm?:

- sampling method: 10-fold cross validation

- best parameters:
 gamma cost
   0.5    4

- best performance: 0.04109673

- Detailed performance results:
  gamma cost      error  dispersion
1   0.5    4 0.04109673 0.005837020
2   1.0    4 0.08323102 0.004959464
3   2.0    4 0.14730705 0.005550034
4   0.5    8 0.04114945 0.005863371
5   1.0    8 0.08323102 0.004959464
6   2.0    8 0.14730705 0.005550034
7   0.5   16 0.04114945 0.005863371
8   1.0   16 0.08323102 0.004959464
9   2.0   16 0.14730705 0.005550034
```

I used 10-fold cross validation for this model. As a results, gamma 0.5 and cost 4 gives minimum cross validation error 0.04109673. To see it clear, let's see the performance plot below.



The first performance plot shows us that less gamma gives more precise prediction, but hard to tell which value is good for cost. So I used the large range of cost (0.25, 0.5, 1, 2, 4, 8, 16, 32, 64) to find the best cost value. The result is showed in second performance plot. If you see the plot, it is also hard to tell which value is good for cost. As a result, cost 2 and gamma 0.5 gives the best result and smallest cross validation error, 0.02382257. Now, I also used the large range of gamma(0.0625, 0.125, 0.25, 0.5, 1, 2) to search the best gamma value. As a result, gamma= 0.125, and cost= 2 gives the best result, which the cross validation error is 0.02013453. The last plot is in below. The misclassification error for the last SVM model is 0.01893939, which is currently the smallest error rate.

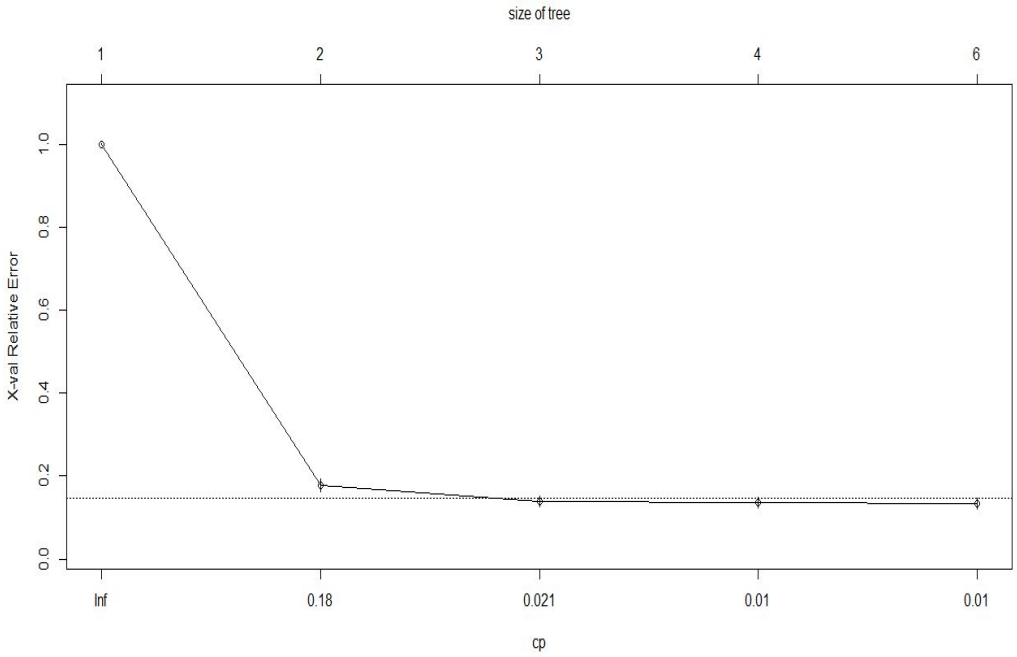


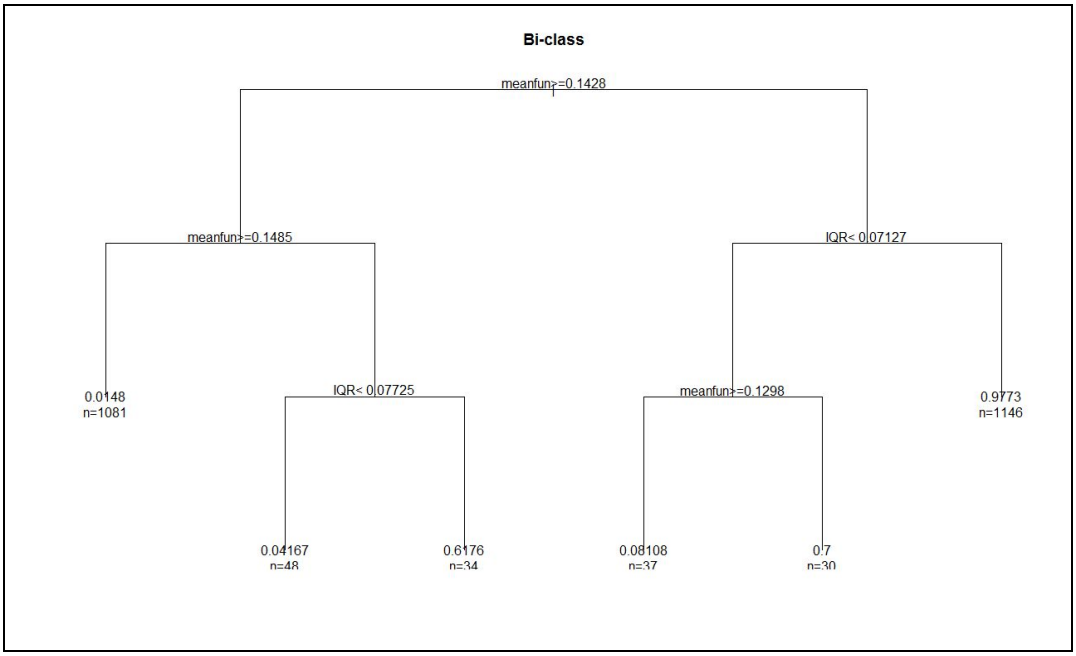
CART (Decision Tree)

Tree method is relatively easy to understand. Each tree has one source (start point) and several nodes (the output). As the number of branches increases, the number of nodes will also increase, so if you use lots of branches, the model will become more complex. Tree is a non-parametric model. It uses recursive classification to attain the model. It could only split each variable into two parts to avoid over-fragmentation. Because most of variables in this voice dataset is about frequency, the range is really important. Thus, this tree method might be powerful. Also, we can easily understand the results through the visualized tree plots.

```
Regression tree:
rpart(formula = label ~ ., data = train)

Variables actually used in tree construction:
[1] IQR     meanfun

Root node error: 593.99/2376 = 0.25

n= 2376

        CP nsplit rel error  xerror       xstd
1 0.826988      0   1.00000 1.00051 0.00027395
2 0.040846      1   0.17301 0.17647 0.01561812
3 0.010684      2   0.13217 0.13792 0.01325111
4 0.010087      3   0.12148 0.13571 0.01339776
5 0.010000      5   0.10131 0.13284 0.01312882
```

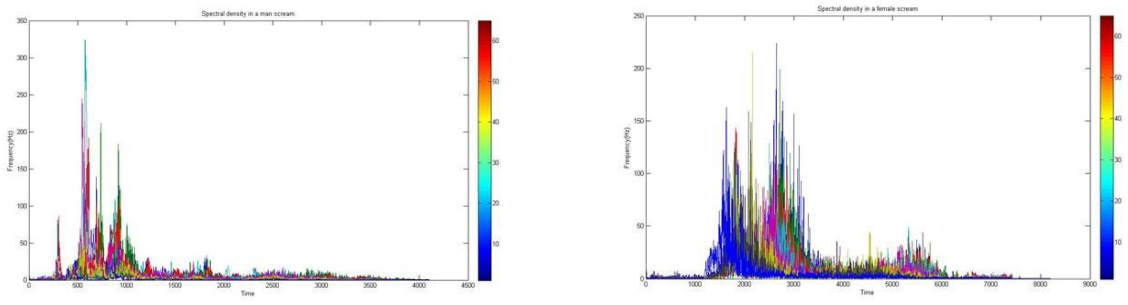
In the table, two variables IQR and ‘meanfun’ are used to construct the tree, which were also important variables in ridge and lasso models. Because the table shows us that 5-splits gives smallest cross validation error 0.10131, 5-splits are enough for this tree.





If you see the first plot (Size of tree), tree of size 3 also has quite low error rate. It is quite interesting because it doesn't have big difference between size 3 and size 6 tree. If you see the second graph (Bi-class tree), the number of nodes in this [meanfun >= 0.1428, meanfun>= 0.1485] condition is

1081, and the number of nodes in this [meanfun<0.1428, IQR < 0.07127] condition is 1146. If you add this two values, it is more than 2000, which means this two branches cover major portion of the dataset. For reference, the value of node in this tree means the mean value of the observations which satisfied the conditions. So, if Meanfun is higher than 0.1485, it might be female (*if the node's value is close to 0, it is female, and if 1, it is male). In case that meanfun is bigger than 0.1428 but less than 0.1485, if IQR is less than 0.07725, it would be female. And in same case, if IQR is bigger than 0.07725, it would be male. It is quite interesting because people usually think that female tend to have wider range of note than male. However, through this observation, the male tends to have higher IQR than female in this model. In one voice research about the male vs female voice characteristic, the result gives us following graph.



As you can see, the spectrum of frequency is pretty same, but there is a big difference in average. The female voice frequency are more located in average, but the male voice frequency is too big or too small, which means male tends to have bigger IQR.

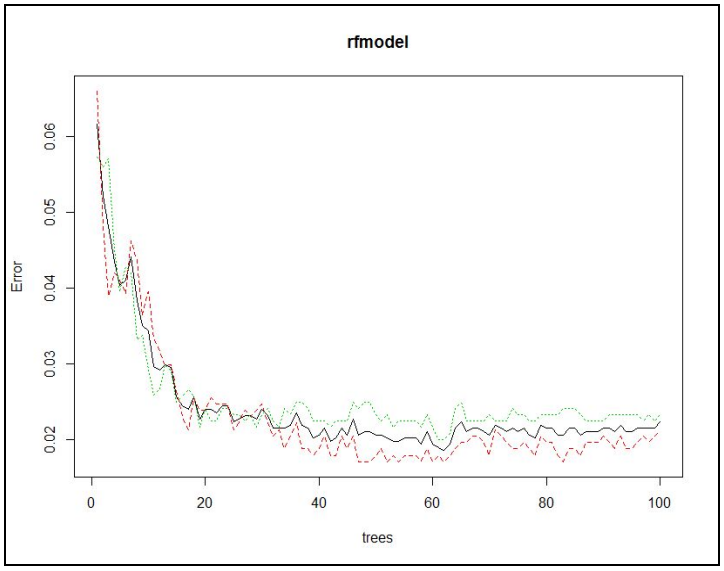
Also, in case that meanfun is less than 0.1428, if IQR is bigger than 0.07127, it is male. It also claim that male tend to have wider IQR. In case that meanfun is less than 0.1428 and IQR is less than 0.07127, if meanfun is bigger than 0.1298, the result is female. It means that even though someone have low mean voice, if IQR is less than 0.07, the result is female. Also, if IQR is bigger than 0.07, it is male.

However, there is one problem that the values of nodes for male are lower than 1. (The values were 0.6 and 0.7). We could improve the performance by adding more branches to these two nodes.

In conclusion, male tends to have low meanfun value and high IQR value, and female tends to have high meanfun value and low IQR value. Misclassification Error for this tree method is 0.03156566.

Random Forest

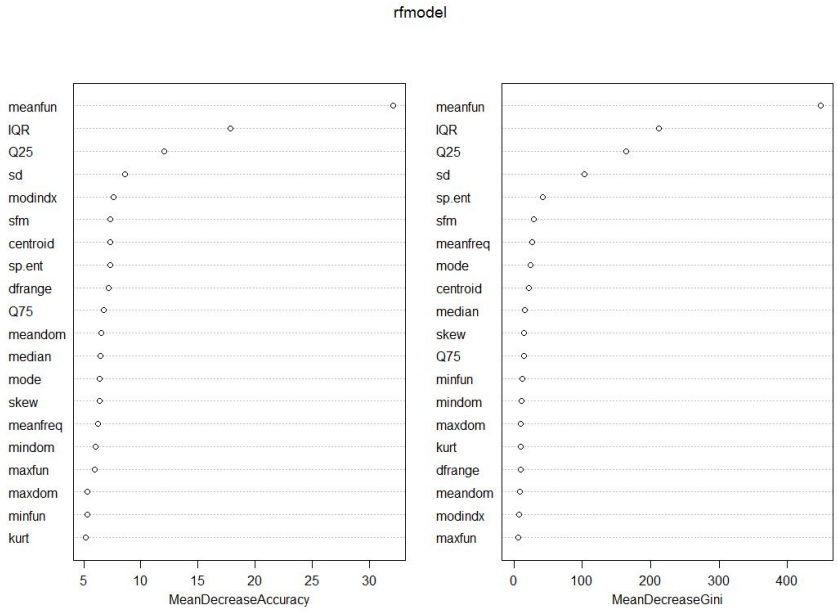
A tree method is built using the whole dataset to construct the tree, but in random forest, it uses a part of the rows, which is selected at random, and also part of features chosen at random. Thus, the random forest is the collection of tree. So, it is much slower than tree, but it has lower error rate. Also, because this method uses bagging (bootstrap means), which works well for low bias and high variance procedures, performance of tree could be enhanced.



This is the plot of number of tree versus error rate. As you can see, when the number of trees are 60, it gives the smallest error rate. The error rate are not going down after 60. Even though random forest prefer the large number of trees, for this dataset, 60 trees are enough.

To get more accurate random forest model, it is important to find out which coefficients could give us more information. This is important procedure to improve the split criterion. The below table and plot are about the importance of coefficients.

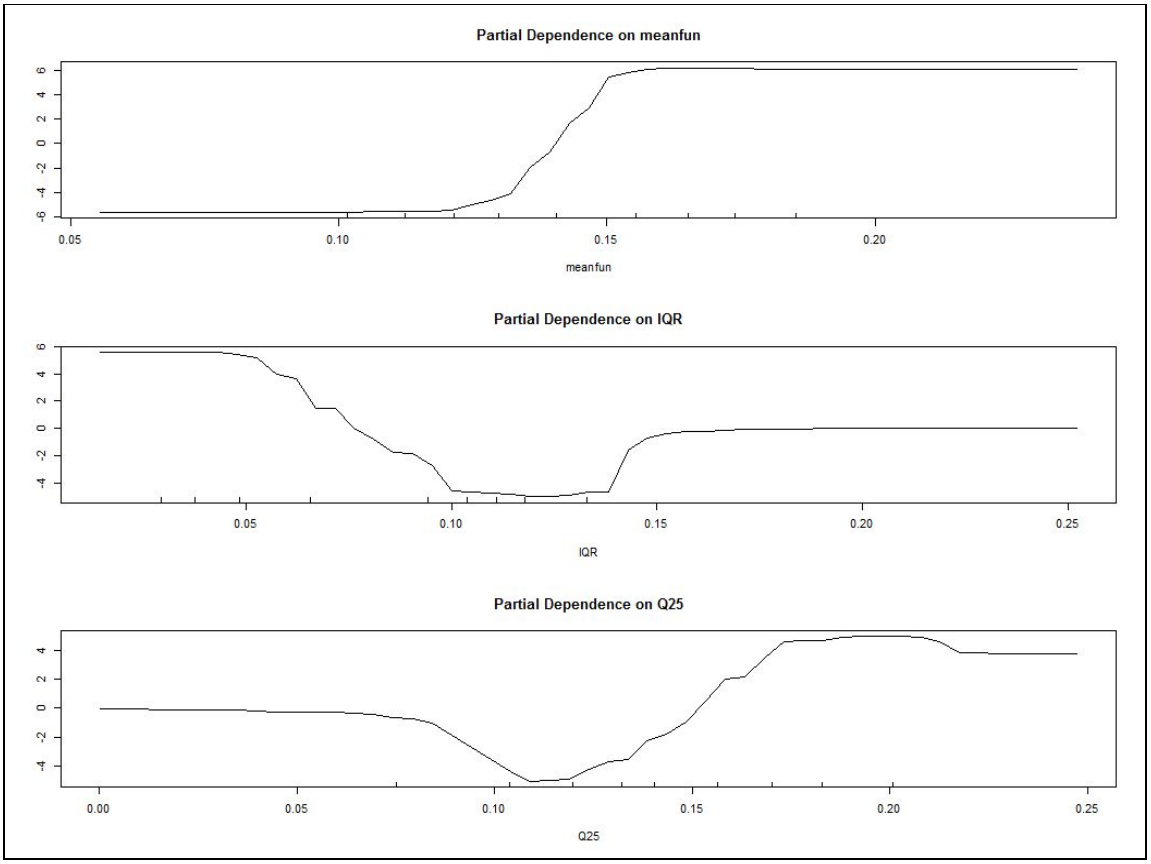
	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
meanfreq	5.059679	4.939992	6.275270	26.700294
sd	6.233322	6.898914	8.601134	102.586457
median	3.608646	5.568098	6.454507	15.533087
Q25	7.222923	9.428067	12.026964	164.001796
Q75	4.618600	6.561079	6.749066	13.991127
IQR	8.165727	18.232266	17.817451	212.456959
skew	5.381038	4.807341	6.385062	14.323437
kurt	3.656159	3.857956	5.192242	9.419982
sp.ent	6.524132	3.946878	7.322210	41.708417
sfm	5.620394	5.688709	7.323891	28.373643
mode	4.661992	6.587787	6.410878	23.614540
centroid	4.189727	5.763360	7.322610	21.163375
meanfun	20.430592	27.884496	32.065367	449.307496
minfun	3.853599	3.955671	5.316798	12.208063
maxfun	4.979853	2.567024	5.984604	6.092606
meandom	4.906585	5.849794	6.517608	8.318242
mindom	3.578158	5.426686	6.062321	10.587781
maxdom	3.772972	4.337273	5.320778	10.318643
dfrange	5.817673	4.012488	7.154272	9.320234
modindx	6.429805	4.168326	7.586002	7.572953



These table and plot tell us that meanfun ,IQR, and Q25 are the Top 3 important variables. In most of the models above, meanfun and IQR are also important variables. For this model, Q25 became

important variables. When I interpret the tree, the value of node for male was a little bit incorrect. (less than 1). So, if we add more information about Q25 (low frequency), we might be able to guess the male more precisely.

We found out which coefficients are important. So, now let's see these important coefficients' performance by partial dependence plots, which shows the important part in each coefficients.



In these partial plots, if the absolute value of y is high, it means corresponding x-value is important. The meanfun partial plot apparently shows us that absolute value of y is big (absolute y = 6) when 'meanfun' <0.12 or meanfun > 0.15. Thus, if meanfun is bigger than 0.15, the result is female, and if it is less than 0.12, the result is male. Also, the 'IQR' partial plot shows us that 'IQR < 0.05 gives the highest y value (absolute y=6). It means that if IQR is less than 0.05, the result is female. If 'IQR > 0.1,

the result is male. For ‘Q25’, it is quite not obvious, but we can say that the range ‘Q25’ > 0.11 and ‘Q25’< 0.12 or the other range ‘Q25 >0.17 and ‘Q25<0.22 gives the highest y value (absolute y = 4). The first range(0.11< Q25 < 0.12) result is male, and the second range result is female. Misclassification error for this rainforest method is 0.02525253, which is less than CART(Tree) method’s error rate.

Conclusion

6 models, K-nn classification , ridge regression, lasso, svm, tree, and random forest, are used for this dataset. Each methods have their strong points, and mostly perform well. IQR, and meanfun was the most important variables for most of methods. The highest error rate was less than 0.1, which is great result. This is the chart of misclassification error of each models.

Model	Misclassification Error	Model	Misclassification Error
K-nn (k=1)	0.03914141	Ridge Regression	0.03030303
K-nn(k=3)	0.04166667	LASSO	0.02525253
K-nn(k=5)	0.0479798	SVM	0.01893939
K-nn(k=7)	0.05808081	CART (TREE)	0.03156566
K-nn(k=10)	0.05808081	Random Forest	0.02525253
K-nn(k=20)	0.07323232		

Except K-nn classification method, the average misclassification error rate is 0.026, so the model accuracy is about 97.4 % overall. The best result is SVM model, which gives 0.018939 error rate, so we can predict the model with 98.1 % accuracy.

Through this model, we could predict the gender by voice more precisely. Now, we can use this result to enhance the voice recognition system. However, I cannot say that SVM is the best model. In the world, there are 8 billion people. However we only used about 4000 observations to predict the test case. Thus, if I use the other train and test set, the results might be different.

Reference

1. Voice Dataset

The website : https://www.kaggle.com/primaryobjects/voicegender

Original :

http://www.primaryobjects.com/2016/06/22/identifying-the-gender-of-a-voice-using-machine-learning/

Author : Kory Becker

2. Male vs Female Voice Plots (page.11)

The website : https://www.projectrhea.org/rhea/index.php/Male_vs._Female_Voice_characterestics

Author : Ananya Panja