Gender Recognition From Vocal Frequency Characteristics

Shreyas Bhat

8/7/2020

1. Introduction

Gender recognition from voice, a seemingly straightforward task for the human brain but can be quite a challenge for a computer. The human brain is able to identify the gender of a speaker on hearing a voice almost subconsciously. However, enabling a computer to perform this task becomes quite tricky. For starters, it needs a microphone to record the voice as a acoustic signal, then comes the interpretation of said signal. We will briefly discuss the initially steps of interpretation in the following sections. This project however, tackles the next step of the problem namely, classifying pre-processed voice samples, originally .wav files, as male or female.

1.1 Understanding acoustic signals

In this era of digital signals , an acoustic signal or audio signal can be analyzed very intricately by studying the frequencies contained within the signal. We can start with the knowledge that, while the human ear has a audible frequency range of 20Hz-20KHz, the human vocal range broadly lies between 20Hz- 280Hz. Therefore, from any given audio sample we only have to analyze the characteristics of this small range of frequencies to classify it by gender.

1.2 The Questions

In the crudest sense we can say that male voice has a lower pitch than female voice. This difference in pitch is essentially due to the different range of frequencies. However, consider cases where the vocal range of an individual is in a more androgynous range, or varied intonations in various languages etc, in realtime samples there are many such condition where samples cannot be simply classified using the frequency range. When we talk about a robust classifying criterion a few important questions present themselves; i) What are the frequency characteristics that differ between genders? ii) Is there a difference in resonance between genders? iii) What features drive the classification and are thereby efficient predictors of the gender?

In this project we delve into these questions and compare the various classification techniques.We will evaluate each of the model by using ‘misclassification error rate’ as the metric as we dont particularly care about false positives or false negatives.

2. Dataset

The dataset we are using here has 3,168 voice sample that are labled as male and female. The original audio (.wav) files have been compiled from various databases and pre-processed using the Specan function in WarbleR R package. The link to original voice samples is attached in the Appendix

The specan function measures 22 frequency parameters on acoustic signals. These 22 parameters and observed values have been compiled into a .csv file. Out of these 22 parameters 2 corresponding to duration and peak frequency are omitted in the dataset as the duration has been set to 20 secs for all samples and peak frequency generates as set of 0's in this case. Therefore including the label we have a dataset that is a 3168 x 21 tibble.

This dataset was originally uploaded on primaryobjects.com a website run by Kory Becker, who is a software developer and has worked on developing AI for popular products such as ALEXA and Google Assistant. She later uploaded the same to kaggle as part of a competition. It can be downloaded here,
https://www.kaggle.com/primaryobjects/voicegender.

The dataset has been imported and the list of properties used for the exercise can be seen below. The columns in the dataset are as listed below:

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   label = col_character()
## )

## See spec(...) for full column specifications.

##  [1] "meanfreq" "sd"       "median"   "Q25"      "Q75"      "IQR"
##  [7] "skew"     "kurt"     "sp.ent"   "sfm"      "mode"     "centroid"
## [13] "meanfun"  "minfun"   "maxfun"   "meandom"  "mindom"   "maxdom"
## [19] "dfrange"  "modindx"  "label"
```

The features corresponding to the displayed columns are explained in the Appendix.

2.1 Split into train and test sets

The dataset has samples labeled with one of classes, male and female in the .csv file. We check for na values and omit them, then check if the samples are ordered by listing all the indices of the samples where the label changes.

```
## # A tibble: 1 x 2
##   index total_rows
##   <int>      <int>
## 1  1585       3168
```

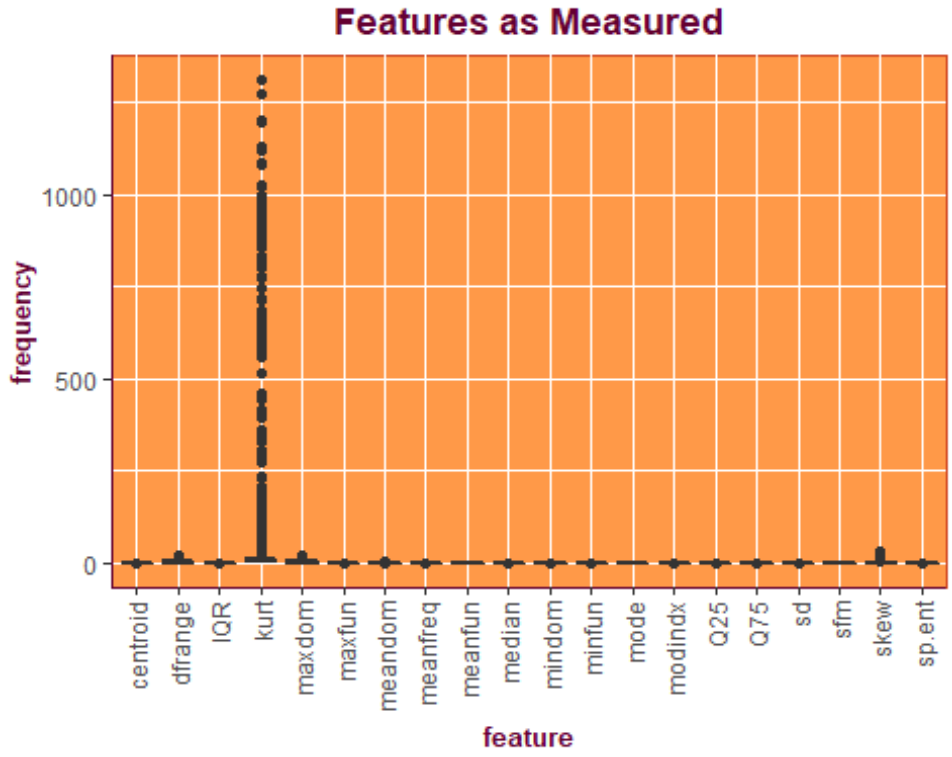
As we can see the only change in label is seen halfway through the our dataset of 3168 rows. This confirms there are no values missing and the data is ordered. Hence, we first shuffle the data and then set aside train and test data with a 80%-20% split respectively.

3. Preliminary Analysis

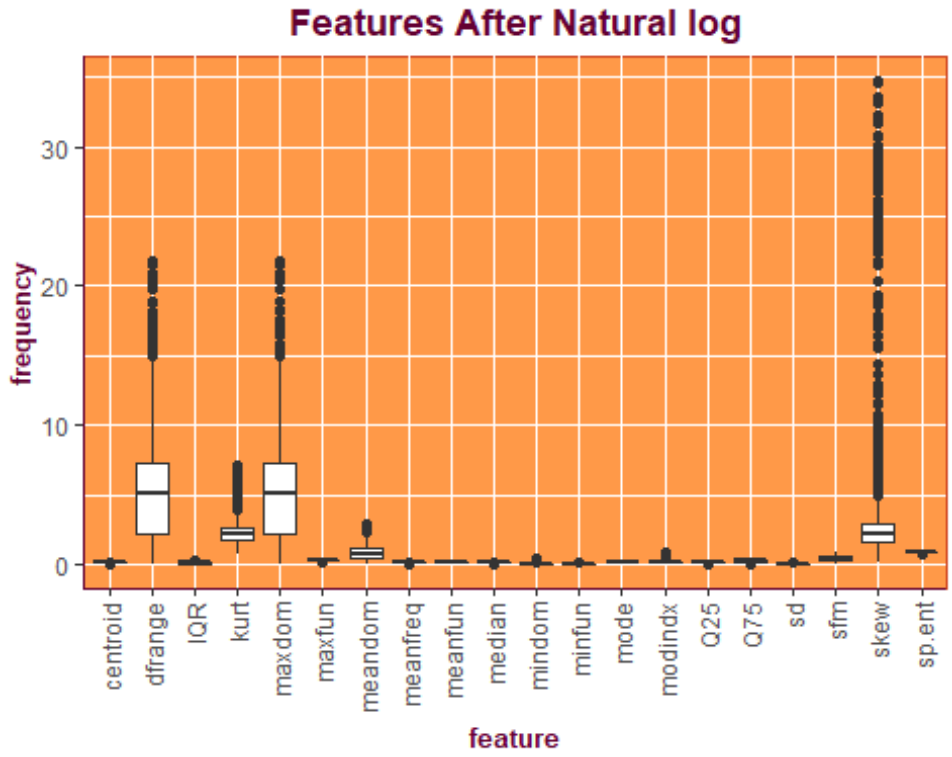
3.1 Feature comparison

Now that we have split the data into train and test sets, we dig a little further into the features. We can view the dataframe to get a better understanding of the nature of values associated with each feature.

However, to efficiently compare the range of values of the features we plot them as shown below in the "Features as Measured" plot.



From the boxplot we can see that the kurtosis is very skewed, therefore we take log of the parameter and re-plot the features.



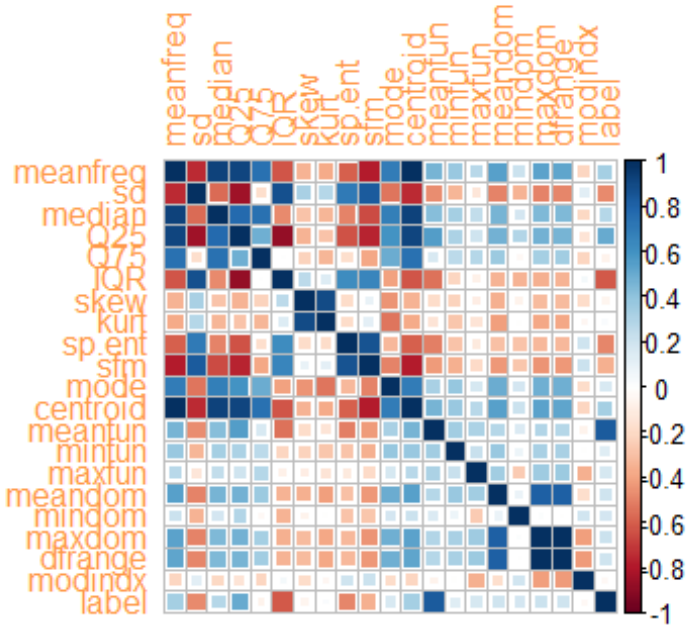
As we can see in “Features After Natural Log” after taking log of the kurtosis the features are more comparable. Now we can explore the features further.

3.2 Correlation and Feature Selection

To explore which features make better predictors, we can start by analyzing the correlation between features and removing redundant features. This is effective as highly correlated features can be expressed as a

We first transform the label as male =0 and female = 1 for easier calculations and store is a separate dataframe.

Correlation between Features with Gender label

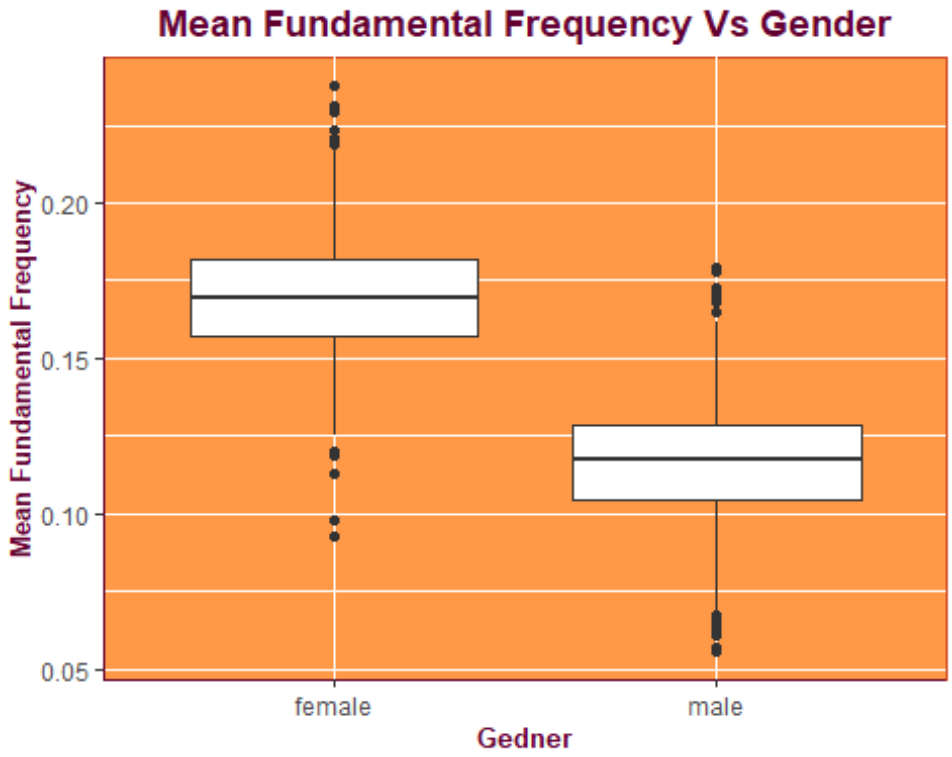


The Heatmap “Correlation between features” shows How independent each feature is from the other. The correlation with the column “label” indicates how much each feature drives the classification. From this we can also see there is strong correlation between several features as indicated by the dark blues(positive correlation) and and dark reds(negative correlation). This indicates that we can reduce number of features used as predictors to build a robust model.

We can furtherexplore the features and analyze their viability as predictors from the heatmap/correlation matrix.

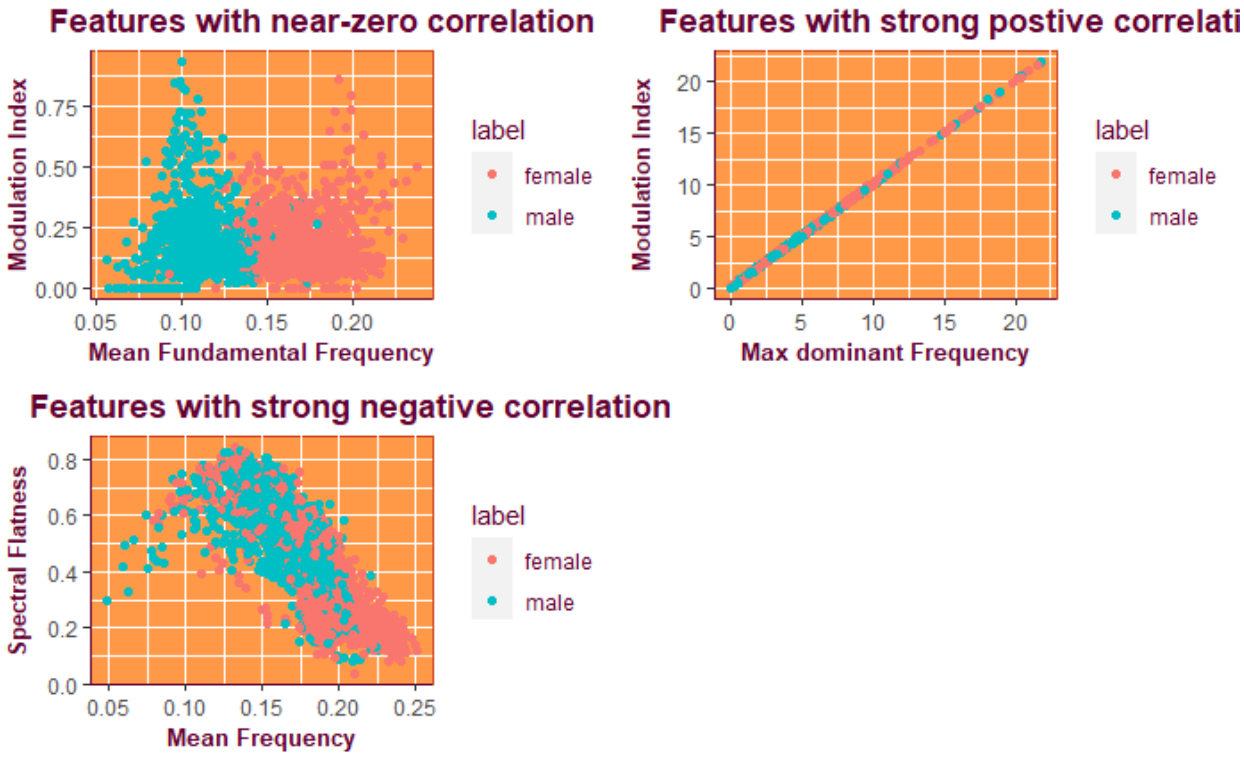
```
#Extract feature with highest correlation with label
which.max(corr.mat[-nrow(corr.mat),ncol(corr.mat)])

## meanfun
##      13
```

First, we start by looking at the correlation between label and features. In this case “meanfun” or " Mean fundamental Frequency" is found to have highest positive correlation with the “label(Gender)”. Additionally, “meanfun” also has relatively low correlation with other features.THis indicates it is a good predictor.



The boxplot of “Mean Fundamental Frequency Vs Gender” clearly shows that Gender is largely separable by the feature “meanfun”. This also indicates that it is a strong predictor.

In the next set of plots we visualize the effects of correlation between the features on classification. We choose 3 pairs of features with coefficients ranging between strong positive to strong negative correlation



From the plots we can see that features with near 0 correlation coefficient make better predictors when used together as the data has less overlap and is easily separable. Evidently, we can find features that are unique which can effectively separate the classes by evaluating the absolute correlation. Therefore we can perform effective feature selection by eliminating the features with high absolute correlation.

First, we remove the 'label' column and recalculate the correlation matrix as we only want to eliminate collinear features wihtout affecting features with correlation to the label. This gives us a correlation matrix of features from which we can identify the ones with absolute correlation greater than a set cut off value.

```
##  [1] "Q75"     "IQR"     "skew"    "sp.ent"  "mode"    "meanfun" "minfun"
##  [8] "maxfun"  "meandom" "mindom"  "modindx" "label"
```

We can select a cutoff for absolute correlation coefficient based on how strictly we want to eliminate features. The standard practice is to use a value in the range "0.7 or higher". From the result shown above we know that 11 features are selected by this method of feature selction. We can now test the performances of classification models with these predictors. We will also be using metrics and details obtained with each model to bolster our understanding of the predictors.

4. Modeling

There are number of classification techniques that can be used for the task. However, we limit the scope of this project to a comparison of logistic Regression, LDA, Decision Tree,

Boosted model and SVM. Before we fit, tune and test models we consider a simple baseline model. This model is the simplest way of performing the gender classification and serves as a reference to highlight the improvement obtained due to each step of analysis.

4.1 Baseline

First, let us take a step back and consider that we don't perform the preliminary analysis. In this case, it seems that we can chose "mean frequency" as a predictor since it is the mean of the frequencies in the vocal range, thereby, an obvious descriptor of the High and low pitched natures of the female and male vocal ranges . We therefore , have fit a simple Baseline model is fit using Logistic regression with 'Mean Frequency" as the predictor.

```
## [1] 3208.301
```

We can extract the AIC or Akaike information criterion (AIC), an estimator of out-of-sample prediction error, from the model summary.The high AIC observed indicates that the model is very inaccurate. We now check the training error rate for the fit.

```
## [1] 36.21302
```

We can see that also the overall training error rate for this baseline model is 36.2%.

4.2 Logistic Regression

4.2.1 Improved Single predictor logit model

Now, let us consider the observations made in Section 3. We can fit another single predictor model with "Mean fundamental frequency" as the predictor instead of "Mean Frequency".As seen from the preliminary analysis in section 3.2 'meanfun" is a powerful predictor. We now check how that reflects on a logit fit.

```
## [1] 737.7149
```

From the much lower AIC of 737.7 indicates that logistic model is better that the baseline model.We now check the training error rate for the fit.

```
## [1] 4.33925
```

The overall train error rate of 4.3% is a big step forward from the baseline as indicated from the AIC.This clearly shows the importance of feature selection when fitting a classification models. And In this case also indicated that mean frequency has a significantly large influence on the classification.

4.2.2 Multi-predictor logit model

Now, that we have illustrated that the feature selection is effective, we fit a model with the reduced 4 predictors obtained form section 3, namely, “Q75”, “IQR”, “skew”, “sp.ent”, “mode”, “meanfun”, “minfun”, “maxfun”, “meandom”, “mindom”, “modindx”.

```
## [1] 510.1516
```

The AIC=510.1 indicates that this multi-predictor logistic model will perform better than the single predictor model. We now check the training error rate for the fit.

```
## [1] 2.564103
```

The overall train error rate of 2.5% is obtained . This reinforces the observation from the AIC that this is a better classifier than single predictor model.Therefore, by all indications the features selected make good predictors.We will continue testing different models with these predictors.

4.3 Linear Discriminant Analysis

First off, we fit a Linear Discriminant Analysis Classifier.

```
## Call:
## lda(lda.formula, data = train.voice.red)
##
## Prior probabilities of groups:
##         0         1
## 0.4946746 0.5053254
##
## Group means:
##         Q75        IQR     skew    sp.ent      mode   meanfun     minfun
## 0 0.2264413 0.11076429 3.452648 0.9163105 0.1525919 0.1157629 0.03428691
## 1 0.2233394 0.05784522 2.994042 0.8729736 0.1778773 0.1697685 0.03968637
##      maxfun   meandom     mindom   modindx
## 0 0.2536565 0.7305132 0.04055460 0.1761349
## 1 0.2643124 0.9459801 0.06329333 0.1675622
##
## Coefficients of linear discriminants:
##                  LD1
## Q75     -10.89576379
## IQR     -17.90598396
## skew      0.08128334
## sp.ent    5.05230490
## mode     -1.88602659
## meanfun  62.60161409
## minfun  -14.35455198
## maxfun   -3.75365705
## meandom   0.33847786
```

```
## mindom   -0.49091260
## modindx   0.01007928
```

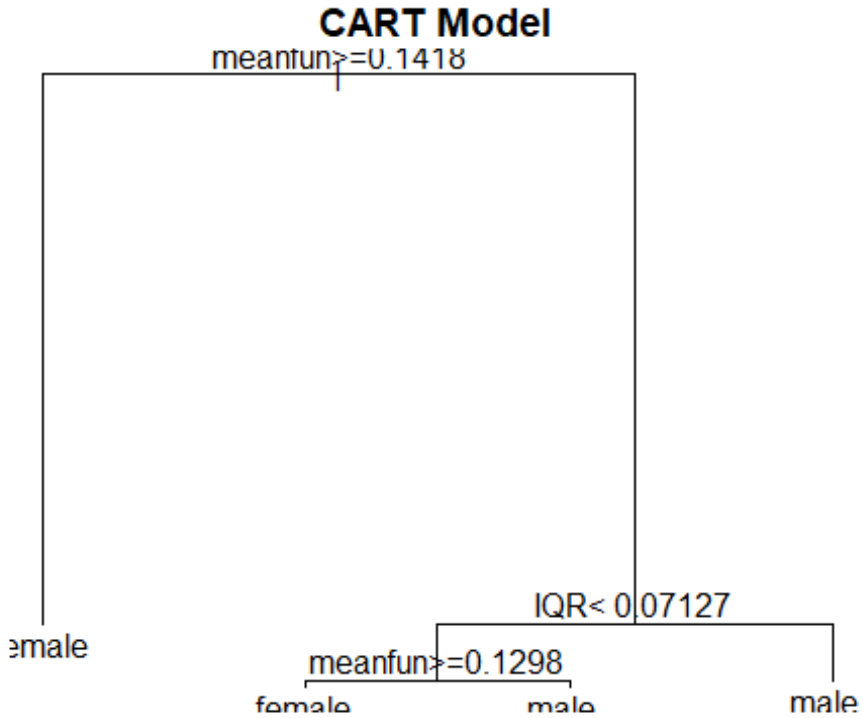
The summary of the LDA fit is as shown here. Here we can see that the linear discriminant(LD1) corresponding to “Mean Fundamental Frequency” has a very high positive coefficient indicating that higher mean fundamental frequency pushes the sample into the class label ‘1’ or ‘Female’. This is consistent with what we observed in the plot “Gender Vs Mean Fundamental Frequency”. Similarly, “Interquartile Range’ or ‘IQR’ has a high negative coefficient indicating that higher IQR pushes the sample towards”0" of “male” label.

```
## [1] 2.840237
```

LDA gives a training error of 2.8% which is slightly higher than corresponding logistic model.

4.4 Decision Tree

Next, we fit a Decision Tree based model. Here we use th rpart package to test and fit a classification and Regression Tree. It selects the best predictors that can separate the dataset into the classes.



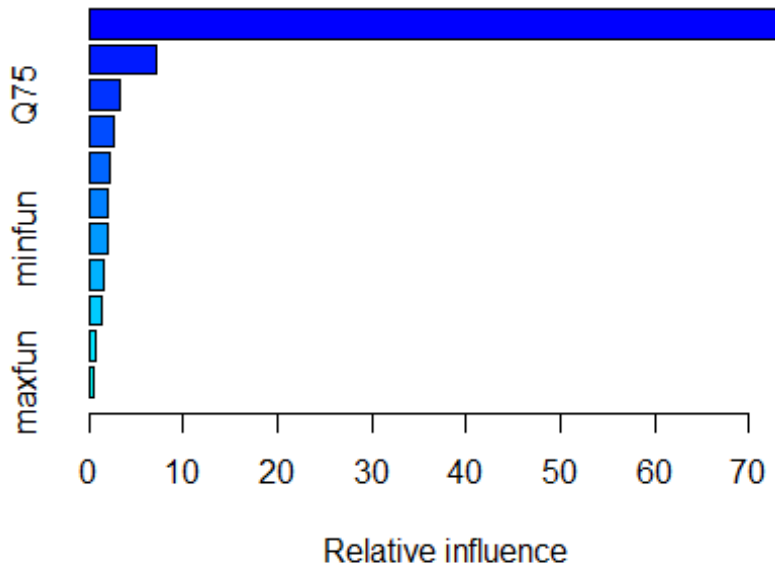
From the plot of the Tree Model we can see that it identifies the Mean Fundamental Frequency of 140Hz as the separator between the classes. We now check the training error rate for the fit.

```
## [1] 3.116371
```

The CART model gives a train error rate of 3.1% which is slightly higher than the multiple predictor logistic model.

4.5 Boosted Model

Next we fit the tree based Boosted Model.




```
##             var    rel.inf
## meanfun meanfun 76.6159712
## IQR         IQR  7.0701666
## Q75         Q75  3.3557844
## mode       mode  2.6595192
## sp.ent   sp.ent  2.1984771
## meandom meandom  2.0234489
## minfun   minfun  2.0103420
## skew       skew  1.4723365
## modindx modindx  1.3342653
## mindom   mindom  0.7181967
## maxfun   maxfun  0.5414922
```

The summary of the Boosted model gives us the plot and a table of the relative influence of the predictors and it reinforces the observations made so far that Mean fundamental Frequency and IQR are the most influential predictors for this classification task. We now check the training error rate of the model.

```
## [1] 0.03944773
```

The Boosted model gives a training error rate near 0%. However, the training error estimates of the boosted model are often substantially underestimated.

4.6 Support Vector Machine

Last model that we fit in this exercise is SVM. We use a Linear Kernel SVM with Cost=10 which is the Measure of the penalty on misclassification used in the optimization.

```
## [1] 1496
```

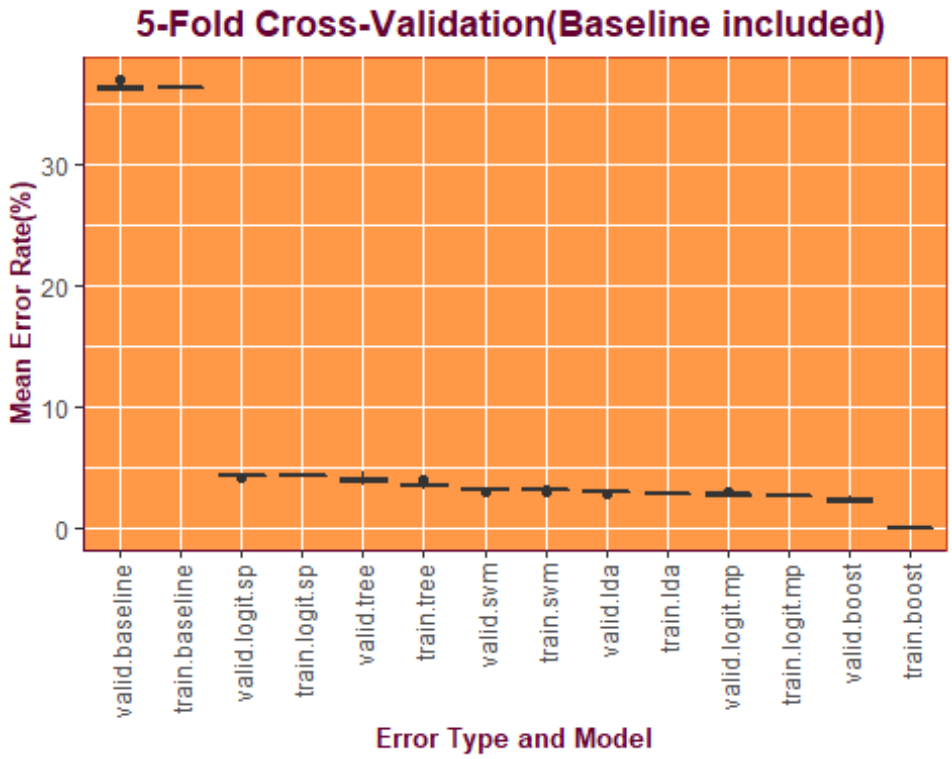
The SVM fit in this case is hard to visualize due to the multi-dimensional nature of the problem. However, we can see that the separating hyperplane uses a total of 1496 support Vectors to separate the dataset. To estimate misclassification, we check the training error rate.

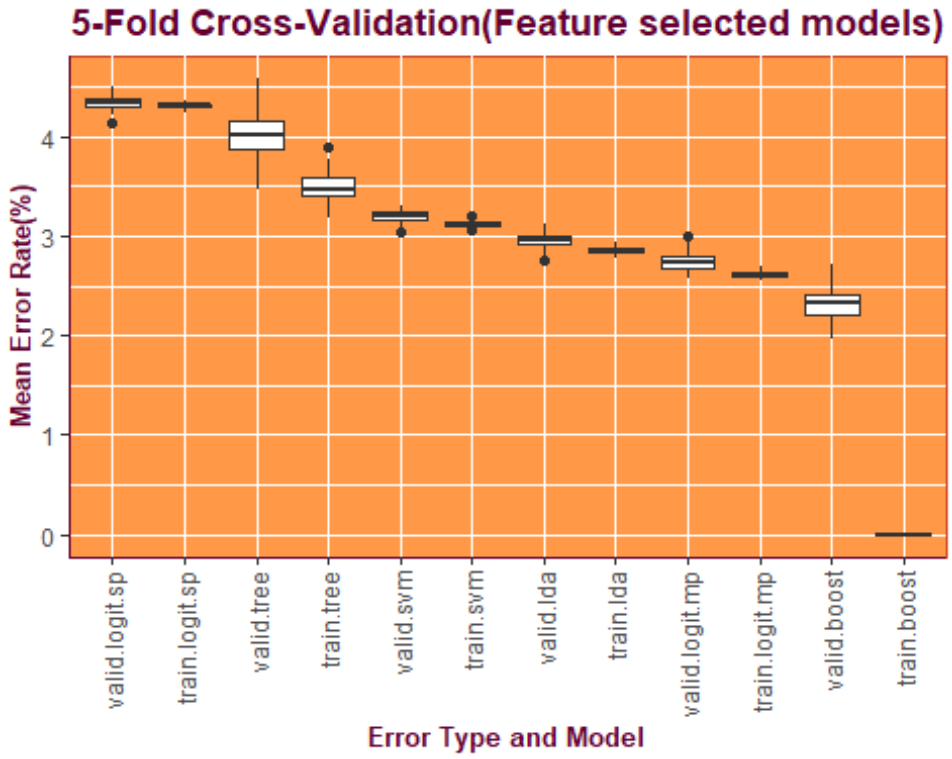
```
## [1] 3.155819
```

SVM model gives a training error rate of 3.1% . We haven't tuned the cost parameter in this exercise. However, it has similar training error as the Boosted model and the Multi-predictor Logistic model. To properly evaluate the fits of all the model we now can use k-fold cross validation to check their performance with an unseen dataset.

4.7 k-fold cross validation of the fits

We can get a good evaluatation of the models by cross validation. In this case, we use iterated 5-fold cross validation over 100 iterations to evaluate the models fitted through sections 4.2 and 4.7.





The boxplots of 5-fold cross validation show us the train and validation error rates of models in decreasing order of their error. From two plots we can see the relative performance of the models. The plot with the baseline shows us that the biggest improvement, in this case near 30% drop in error rate, is achieved from proper feature

selection.
From the boxplots ,“feature selected models” , we can see that the boosted model classifies the training data with nearly 0% error , it has validation error near 2.5%. However, this is the best model indicated so far. Mutli-predictor logistic model comes a close second with train error just over 2.5% and validation error just below 3%.

4.9 Evaluation on test set

Finally we evaluate our models on the test set to see how they perform on a fresh dataset.

```
## # A tibble: 1 x 7
##   Baseline Logit.sp  Tree   SVM   LDA Logit.mp Boosted
##      <dbl>    <dbl> <dbl> <dbl> <dbl>    <dbl>   <dbl>
## 1     33.5     6.00  4.58  3.95  3.16     2.84    2.369668
```

The overall test error rates also achieve the biggest step forward from feature selection. The error rates and model performances on Test set are concurrent with the observations from validation errors. Boosted Model stands as the best model for the task with a Test Error of 2.3%. However, the recommended model would be the Multi-Predictor Logistic Regression as it is much more efficient and still gives a test error of 2.8%.

5. Conclusion

From this exercise showed that the Boosted Tree model is the best for the task of Classification of Gender from Vocal Frequency. The model showed a train error of nearly 0%, validation error below 2.5% and a Test Error of 2.3%. However, as mentioned earlier the Logistic model is more efficient with using computing resources and still has a train error 2.5%, valid error under 3% and test error of 2.8%.Therefore, I would recommend a multi predictor logistic regression model for the task. While these may not exactly reflect real world accuracies. We can be confident in the models.

We also answered the questions about Gender vs Frequency characteristics that we set out to explore. Firstly, most Frequency characteristics are marginally different between genders. However, mean fundamental frequency is the most distinct feature of the voice between genders. Resonance is a quality which is directly related to the Fundamental Frequency, it is therefore, clearly a feature distinct to gender. Another important feature is the Interquartile range (IQR). While the other features differ between voices and genders they aren’t as distinct.

While the exercise explains most of questions, it poses a few that are beyond the scope of the dataset. For instance, can we characterize falsetto as male/female accurately? This is a question for a dataset of falsetto samples. The good news is that it is fairly easy to create our own voice samples and effectively analyze them with effective use of the techniques used in this exercise.

Appendix

A)

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: interquartile range (in kHz)
7) skew: skewness of frequency distribution
8) kurt: kurtosis of frequency distribution
9) sp.ent: spectral entropy
10) sfm: spectral flatness
11) mode: mode frequency (in kHz)
12) centroid: frequency centroid
13) meanfun: average of fundamental frequency measured across acoustic signal (in kHz)
14) minfun: minimum fundamental frequency measured across acoustic signal (in kHz)
15) maxfun: maximum fundamental frequency measured across acoustic signal (in kHz)
16) meandom: average of dominant frequency measured across acoustic signal (in kHz)
17) mindom: minimum of dominant frequency measured across acoustic signal (in kHz)
18) maxdom: maximum of dominant frequency measured across acoustic signal (in kHz)
19) dfrange: range of dominant frequency measured across acoustic signal (in kHz)
20) modindx: modulation index. Calculated as the accumulated absolute difference between adjacent measurements of fundamental frequencies divided by the frequency range
21) label: male or female

B)

Audio samples used in dataset can be downloaded here:http://festvox.org/cmu_arctic/