Introduction

For the purposes of the Ensemble Classifier Case Study I chose a dataset from Kaggle titled "Gender Recognition by Voice and Speech Analysis." The purpose of the data is to identify a voice as either male or female based upon the acoustic properties of the voice and speech patterns. This dataset makes a great use case for considering ensemble classifier approaches first and most obviously because the target attribute is categorical as either male or female. What's more, the fact that the target attribute is binary is also helpful when using ensemble approaches such as AdaBoost. The fact that all the descriptive variables are numerical is also helpful in creating unambiguous distance calculations for categorization approaches such as K Nearest Neighbors.

To begin the paper, I'll consider several categorization approaches individually to establish a baseline for model accuracy. I'll next consider algorithms that produce ensembles by iteratively building new models using random resampling of the training data or by assigning weights to the training data for selection in each training iteration. Finally, I'll see if I can create a custom ensemble approach from the underlying models described in the first approach that outperforms any of the original models. Because the target attribute is balanced (equal number of male and female voices in the sample), statistical methodology will be used with each layer to test hypothesis regarding the accuracy of each model or ensemble approach.

Data Description

The dataset consists of 20 descriptive attributes for voices such as the frequency, skewness, and flatness. The target attribute is binary as either male or female. The dataset is pulled from Kaggle and can be found by following https://www.kaggle.com/primaryobjects/voicegender. In total, there are 3168 categorized voice samples which are equally distributed between male and female voices. The data consists of three major categories of attributes along with a handful of assorted others. Specifically, we have a number of descriptive attributes related to the frequency of the voice, descriptive attributes around what the researchers call a "fundamental frequency," and descriptive attributes around what is called a "dominant frequency." A full description of the dataset's attributes can be found in Appendix I.

Data Cleaning

Most probably because of the relatively cheap acquisition cost of the data (simply recording voice samples from volunteers), the data is very clean with no missing data points. The only data cleaning performed was to produce a scaled version of the data using the *scale* function in R so that each attribute in the modified dataset has a mean of zero and a standard deviation of one. To scale the descriptive variables in this manner, I had to first remove the target attribute from the set as it is categorical. I then had a matrix of numerical values that could be scaled using the *scale* function in R. Finally, I needed to add the target attribute back to the normalized dataset. The purpose for normalizing the data in this way is so that it will be better fit for algorithms requiring distance calculations such as K Nearest Neighbors.

Data Analysis

Not surprisingly, the descriptive attributes in each of these subgroups within the dataset described above display high intragroup correlation. If the average dominant frequency for an individual is high, it stands to reason that the 75th percentile of the dominant frequency for this same individual is also high. Comparing the mean frequencies for each of these three groups – frequency, fundamental frequency, and dominant frequency, reveals a high intergroup correlation as well.

Figure 1 displays visualizations for these inter and intragroup correlations. Empirically, it holds that there would be correlation in attributes describing a given individual's voice tones and patterns.

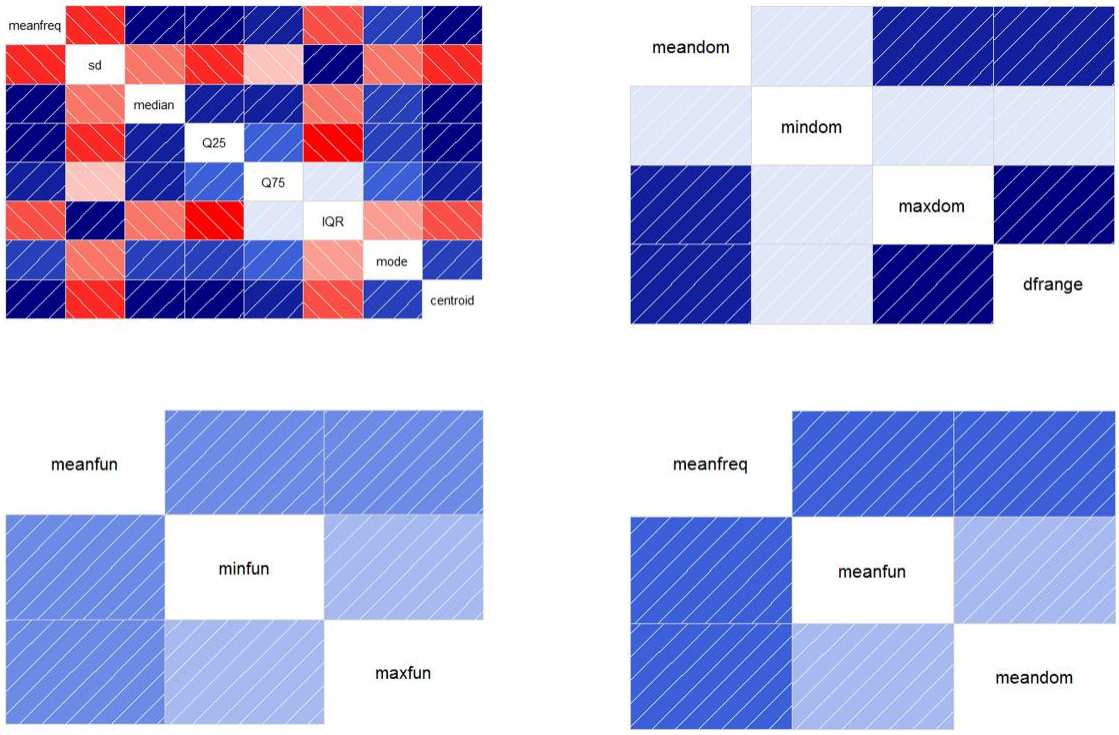

Figure 1 - Correlation between Frequency Variables

Experimental Results

Independent Categorization Approaches

I'll begin the analysis by creating models using specific categorization approaches and then comparing their results as a baseline for evaluating the ensemble methods. For the purposes of this exercise, Decision Trees, Naïve Bayes, and a K-Nearest Neighbor were used as categorization approaches to label instances as either male or female voices. 10-fold cross validation was used to evaluate each method. The K Nearest Neighbor model included a bit more tuning by allowing the *train* function in R to select the optimal number of nearest neighbors (k = 5 is given as the optimal number). This model also used the scaled dataset where the mean of each attribute was zero and the standard deviation was one. Results for the accuracy of each of these three models along with a 95% confidence interval is shown in Figure 2.

Method	Accuracy	95% CI
Decision Tree	0.9618	(0.9545, 0.9682)
Naïve Bayes	0.9299	(0.9205, 0.9386)
K Nearest Neighbor	0.9842	(0.9792, 0.9883)

Figure 2 - Accuracy of Individual Categorization Models

Upon evaluating the results, it appears that the K Nearest Neighbor model outperforms the Decision Tree model which outperforms the Naïve Bayes model. However, can we state this with statistical significance? To make this assertion we will first need to perform hypothesis testing. Figure 3 illustrates the main data points and conclusion for both the hypothesis that the Decision Tree model outperforms the Naïve Bayes model and that the KNN model outperforms the Decision tree model. Note that the n for both calculations is 3168 (every instance in our dataset). In both cases, we can reject the null hypothesis at a significance level of .01. Translated, we can then accept the alternative hypothesis that the accuracy of the Decision Tree model is greater than the accuracy of the Naïve Bayes model and the accuracy of the K Nearest Neighbors model is higher than the accuracy of the Decision Tree model.

Null Hypothesis	Sample Mean	Hypothesized Value	Population s.d.	Lower p-tail value	Conclusion
Naïve Bayes performs better than decision tree	0.9299	0.9618	0.003405505	0	Reject Null Hypothesis
Decision Tree performs better than KNN	0.9618	0.9842	0.00221553	0	Reject Null Hypothesis

Figure 3 - Hypothesis Test Results for Base Models

Algorithmic Ensemble Approaches

Next, we'll evaluate algorithms that generate ensemble approaches based on the way they resample the training data. These types of models do not require cross validation because the training data will be resampled until a specific parameter is met as set either by the algorithm by default or by the practitioner writing the model. The first approach we will consider is a Random Forest. Random Forests not only randomly resample the training data used for creating decision trees and leverage majority voting for classification such as would occur in a Bagging approach, but they also randomly choose the features to be included in the trees. We will also consider the AdaBoost algorithm with the *ada* algorithm in R. AdaBoost leverages a Boosting approach. Unlike Bagging, Boosting does not randomly resample the training data. Instead, the algorithm assigns weights to the training input based on the accuracy of the model for that input in previous iterations. Instances that have been miscategorized receive a higher weight, making them more likely to be included in the training data for future iterations of the model. Instances that are correctly classified conversely receive lower weights so they are less likely to be included in the training data.

Figure 4 describes the results seen for each of these two algorithmic ensemble approaches. Because both models correctly predict all 3168 instances in the training set, we won't go through with computing confidence intervals to confirm that these models outperform our base classification models.

Model	Accuracy	95% CI
Random Forest	1	(0.9988, 1)
AdaBoost	1	(0.9988, 1)

Figure 4 - Performance for Algorithmic Ensemble Approaches

Custom Ensemble Approach

In addition to algorithms that generate ensemble approaches by the way they resample the data, custom ensemble models can also be created based on the results of individual classification

approaches. Can we create a custom ensemble model combining the results of the Decision Tree, K Nearest Neighbors, and Naïve Bayes models created for baseline comparison that outperforms any of the original models? One approach to explore this type of custom ensemble is to assign weights to the predictions made by each of our three underlying models and then make predictions based on the weighted sum.

To assign weights, we first create a (3168 x 3) matrix consisting of the predictions for each instance in our dataset made by each of the three classification models. In addition, we must ensure that the data is numerical – in this case, male is represented by 1 and female is represented by 2. We then create a (3168 x 1) vector that holds the true label for each instance – again represented as 1 for male and 2 for female. Next, we can solve the system of equations using matrix multiplication and the *solve* function in R. Figure 5 displays the weights assigned to each of our underlying models. The weights assigned hold empirically when we consider the performance of each model. It makes sense that the highest weight value would be assigned to the K Nearest Neighbor approach as this is the categorization algorithm with the highest accuracy. Similarly, we're assigning the lowest weight value to the Naïve Bayes approach as this categorization model had the lowest accuracy on our data.

Model	Decision Tree	Naïve Bayes	K Nearest Neighbors
Weight	0.18548240	0.03188396	0.78091626

Figure 5 - Model Weights for Custom Ensemble Approach

The next step is to apply the weights to the predicted values of each of the underlying categorization algorithm to generate a summed score. This again can be accomplished with matrix multiplication if we multiply our (3168 x 3) matrix of predictions made by each of the underlying values by the (3 x 1) matrix of weights, we're left with a (3168 x 1) vector of weighted results. After rounding each of these values to the nearest integer, we know have the weighted predictions from our three underlying models. Figure 6 displays the accuracy of our ensemble approach and the results of a hypothesis test. Interestingly, the results of our custom ensemble model are identical to what we saw with the K Nearest Neighbors approach, and it follows that we fail to reject the null hypothesis that the K Nearest Neighbors approach has a higher accuracy than the custom ensemble approach.

Method	**Accuracy**	**95% CI**
Custom Ensemble Approach	0.9842	(0.9792, 0.9883)

Null Hypothesis	**Sample Mean**	**Hypothesized Value**	**Population s.d.**	**Lower p-tail value**	**Conclusion**
KNN method outperforms the custom ensemble approach	0.9842	0.9842	0.00221553	0.5	Fail to Reject

Figure 6 - Custom Ensemble Results

Experimental Analysis

What we've seen is that each of our individual models perform relatively well in predicting the gender of the volunteer based on the acoustic properties of their voice with our worst model – the Naïve Bayes algorithm – correctly predicting 93% of the instances. As noted in the Data Analysis section,

many of the attributes in our dataset are highly correlated with each other. As we know, the basic "naïve" assumption of the Naïve Bayes model is that the descriptive characteristics are independent. Because this is clearly not the case in our dataset, it holds that the Naïve Bayes approach will perform poorly. Our next best standard categorization approach was the Decision Tree. This is an interesting benchmark because the Decision Tree is the underlying approach in both the Random Forest and the AdaBoost ensemble algorithms. The standard Decision Tree approaches builds one "smart" tree where the information gain of each attribute is considered at every node and was able to correctly predict 96% of the instances. The best performing individual categorization approach, with 98% of the instances correctly predicted, was the K Nearest Neighbors model where the scaled dataset is used and the 5 nearest neighbors are selected to determine the predicted categorization. The fact that the K Nearest Neighbor approach outperformed the Decision Tree model indicates that there is not a small subset of features that clearly differentiate male voices from female voices. If this were case, these attributes would lead to a much higher information gain and improve the accuracy of the Decision Tree. Instead, the K Nearest Neighbors approach consider all fields equally and indicates that the voices were grouped by gender more generally across all of their attributes.

When we think about the fact that our "smart" Decision Tree was able to correctly identify 96% of the instances, it's remarkable that both the Random Forest and the AdaBoost algorithms were able to correctly identify 100% of the instances with an ensemble of weak learners. This is an excellent display of the power of ensemble approaches and how they can be used to improve categorization results. Our custom ensemble approach, on the other hand, had a performance that matched exactly the performance of the base K Nearest Neighbors approach. This can be attributed to two things. First, the weight assigned to the K Nearest Neighbor algorithm was significantly higher than the other two algorithms. Second, there were many instances that were miscategorized by two or even all three of the algorithms. In order for full benefits to be gained from ensemble approaches, the underlying models need to be "wrong in different ways." In other words, we'd see better performance from the ensemble if each misclassified instance was incorrectly classified by only one of the three algorithms. Although we weren't able to find gains in this particular instance by generating a custom ensemble, this method is still valid and can improve accuracy under the right conditions.

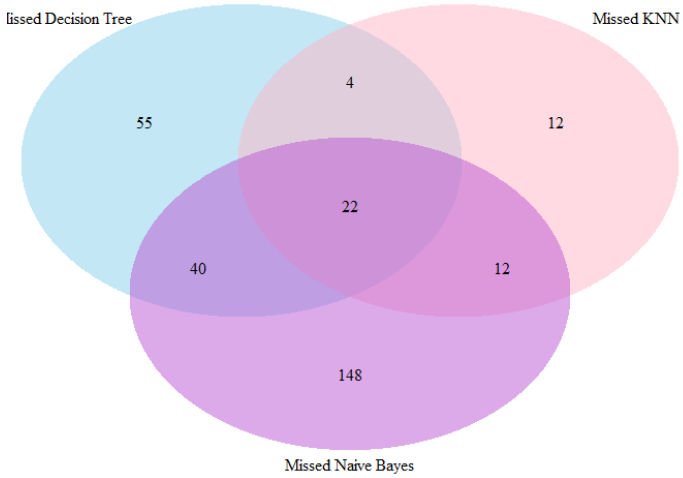

Figure 7 - Instances Missed by Classification Algorithms

Conclusion

Overall, we found that our algorithmic ensemble approaches were able to correctly classify 100% of the instances. This was an improvement over the best individual categorization approach of K Nearest Neighbors which correctly classified 96% of instances. This Case Study demonstrates a

significant measurable improvement that can be gained with an ensemble of weak classifiers over a single strong classifier. High correlation among the underlying description attributes resulted in weaker performance for the Naïve Bayes classification method. Our custom ensemble approach was unable to improve upon the performance of the K Nearest Neighbor approach due to high relative weight assigned to the algorithm and the relative few instances that were misclassified only by the KNN approach. Future work can consider including additional base categorization approaches to be included in the custom ensemble method that will help to spread the weight assigned to each classifier. Other approaches can perform feature extraction to reduce the correlation in the feature set.

Appendix I – Data Dictionary

Label	Description
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 (see note in specprop description)
kurt	kurtosis (see note in specprop description)
sp.ent	spectral entropy
sfm	spectral flatness
mode	mode frequency
centroid	frequency centroid (see specprop)
peakf	peak frequency (frequency with highest energy)
meanfun	average of fundamental frequency measured across acoustic signal
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
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. Calculated as the accumulated absolute difference between adjacent measurements of fundamental frequencies divided by the frequency range
label	male or female