Voice-Based Gender Identification Using Machine Learning

Steve Jadav
Department of Computer Science and Engineering
Institute of Technology and Management Universe
Vadodara, India
stevejadav1998@gmail.com

***Abstract*— Gender identification is considered to be one of the major problems in the field of signal processing. Formerly, this problem has been solved using various image classification techniques which typically includes information extraction from a set of images. However, gender classification using vocal features has recently been a topic of interest to a lot of researchers across the globe. A close scrutiny of some of the human vocal features reveals that classifying gender goes way beyond just the frequency and the pitch of a person. One of the most challenging problems faced in machine learning is feature selection or as is technically known as dimensionality reduction. A similar problem is faced while deciding gender-specific traits—which serve a significant purpose in classifying the gender of a person. This paper will inspect the efficiency and significance of machine learning algorithms to the voice-based gender identification problem.**

Keywords*— *Machine learning; Support Vector Machine; Nearest Neighbors; Grid Search; Statistical Significance; Gender Classification.

I. Introduction

Machine learning has been a recent trend and a course of study. Institutions and business giants all over the world, from small-scale to large-scale, have started to shift gears and invest more in such techniques. It basically involves knowledge mining using various statistical learning approaches. Applying these methodologies in prediction and classification is seemingly interesting but can be gruesome in certain cases where the data is insufficient to elicit meaningful insights. Also, evaluating the impact of each feature on the final model is an involute task. It is quite easy for a human brain to differentiate between various voices. However, that similar task can be complex for a computer. Gender identification is one such interesting problem, the results of which can be found using standard machine learning techniques. [1]Vogt and Andrè suggested that determining gender from a speech in turn help improve automatic emotion recognition from speech. This experimental study tries to determine the most performant models using our dataset.

II. Dataset, Features, and Preprocessing

Each of the voice samples used for extracting features are of .WAV format. They are then fed into the specan function from the WarbleR R package for acoustic analysis [2]. Specan specifically measures 29 acoustic parameters or characteristics on acoustic signals for which the start and end times are to be provided [3]. 21 out of these acoustic properties are then saved into a CSV file so that it can be further pre-processed and analyzed using Python's data science libraries. The CSV file contains 3168 rows and 21 columns, out of which 20 columns are for each feature and the last column is for the label i.e. male or female. The length of each recorded sample is cut off at 20 seconds and hence the duration feature has been removed from the dataset. The target here is to examine the helpfulness of these features in determining a person's gender. A brief representation of some of the statistical metrics of the data is shown in Fig. 1.

	mean	std	min	25%	50%	75%	max
meanfreq	0.180907	0.029918	0.039363	0.163662	0.184838	0.199146	0.251124
sd	0.057126	0.016652	0.018363	0.041954	0.059155	0.067020	0.115273
median	0.185621	0.036360	0.010975	0.169593	0.190032	0.210618	0.261224
Q25	0.140456	0.048680	0.000229	0.111087	0.140286	0.175939	0.247347
Q75	0.224765	0.023639	0.042946	0.208747	0.225684	0.243660	0.273469
IQR	0.084309	0.042783	0.014558	0.042560	0.094280	0.114175	0.252225
skew	3.140168	4.240529	0.141735	1.649569	2.197101	2.931694	34.725453
kurt	36.568461	134.928661	2.068455	5.669547	8.318463	13.648905	1309.612887
sp.ent	0.895127	0.044980	0.738651	0.861811	0.901767	0.928713	0.981997
sfm	0.408216	0.177521	0.036876	0.258041	0.396335	0.533676	0.842936
mode	0.165282	0.077203	0.000000	0.118016	0.186599	0.221104	0.280000
centroid	0.180907	0.029918	0.039363	0.163662	0.184838	0.199146	0.251124
meanfun	0.142807	0.032304	0.055565	0.116998	0.140519	0.169581	0.237636
minfun	0.036802	0.019220	0.009775	0.018223	0.046110	0.047904	0.204082
maxfun	0.258842	0.030077	0.103093	0.253968	0.271186	0.277457	0.279114
meandom	0.829211	0.525205	0.007812	0.419828	0.765795	1.177166	2.957682
mindom	0.052647	0.063299	0.004883	0.007812	0.023438	0.070312	0.458984
maxdom	5.047277	3.521157	0.007812	2.070312	4.992188	7.007812	21.867188
dfrange	4.994630	3.520039	0.000000	2.044922	4.945312	6.992188	21.843750
modindx	0.173752	0.119454	0.000000	0.099766	0.139357	0.209183	0.932374

Fig. 1. General Statistical Characteristics of Data

Some of the important features from the above presented set are meanfun: average of fundamental frequency measured across the acoustic signal, Q25 (first quartile frequency): the frequency at which the signal is divided in two frequency intervals of 25% and 75% energy, IQR (interquartile time range): the time difference between Q75 and Q25 (in secs), dfrange: dominant frequency range measured across the original signal, sd: standard deviation of frequency, skew: a measure of asymmetry of the spectrum.

III. Software Libraries

Python is a seemingly flexible language and has an active community. So, performing Machine Learning becomes a straightforward task, rather than writing the entire algorithms from scratch. Data analysis and visualization are at the heart of any knowledge mining task, and with Python's Data

978-1-5386-6947-1/18/$31.00 ©2018 IEEE

Science libraries, one can easily plot and draw conclusions from interactive visualizations. The libraries which are used for this study are, Matplotlib, Pandas, NumPy, Seaborn, and Scikit Learn. [4, 5, 6, 7].

IV. Background Study

Since there are only two categories of response values in the dataset, the problem is narrowed down to binary classification. Any general classification algorithm such as Logistic Regression, Support Vector Machine, Nearest Neighbours, Discriminant Analysis etc. can be applied on the data. These techniques are by far, the most commonly used machine learning algorithms.

[8] Kuynu Chen, in his work, shows that the discriminant analysis classifier gives the most interesting results in terms of test error rate and precision. However, even this model still suffers from a test error rate of greater than 10%. Albeit, Chen states that running backward selection on the feature set can minimize the test error rate. They extracted the audio features, 24 in total, from Yaafe [9]. Their dataset consists of 12004 data points, amongst which 6286 are labeled 'female' and 5718 are labeled 'male'.

[10] Becker's study on a dataset similar to ours, shows that in order to gain a deeper understanding of the model and to determine the exact properties that indicate a gender of a person, a classification and regression tree (CART) should be applied. His results indicate that the mode frequency (mode) serves as a root node for detecting the gender of a person. Traversing further down the tree evinces that minimum fundamental frequency, maximum dominant frequency, first quantile hertz, skewness, median frequency, additionally correspond to gender classification. The CART model results in an accuracy of 81% on the training set and 78% on the test set. He further takes CART, a step further and applies Random Forest to the data. This achieves a positive boost over CART with an accuracy of 100% on the training set and 87% on the test set.

Training machine learning models is not much of a challenge when working with Python's libraries. However, a rather difficult task is to achieve a trade-off between bias and variance. Over-fitting is a common problem which occurs when the model function is too complicated to be able to predict accurate values. This implies that the variance of the model is too high and that it is trying to fit every point inside the resultant curve, eventually performing poorly on unseen data. One typical method to evaluate the performance measure of a model is to analyze the training as well as the test error rate. A good model not only has a good training accuracy but also a good test accuracy, and if both the accuracy scores are closer to each other, in terms of their values, then it is likely that the model is not over-fitting the data.

V. Experimental Results

The entire dataset is first split into training and test datasets. The test dataset contains 30% of the entire data since the larger the training dataset, the better is the performance of the model. If the model is trained with a large training dataset, it is much likely to find significant patterns.

After splitting the data, first, the nearest neighbours classifier is trained. The accuracy of this model largely depends on the value of K (an integer). A low value of K might mean that the model is overly flexible and finds unusual patterns in the data. Although the training error rate might be low in this case, the test error rate is most probable to be high, which indicates a bad model. In general, a low value of K indicates that the classifier has low-bias but high-variance. With a growth in the value of K, the model becomes less flexible and hence corresponds to a high-bias and low-variance classifier. In order to choose an optimum value for K, the elbow method is performed. A plot indicating the error rate with different values of K is shown in Figure 2.

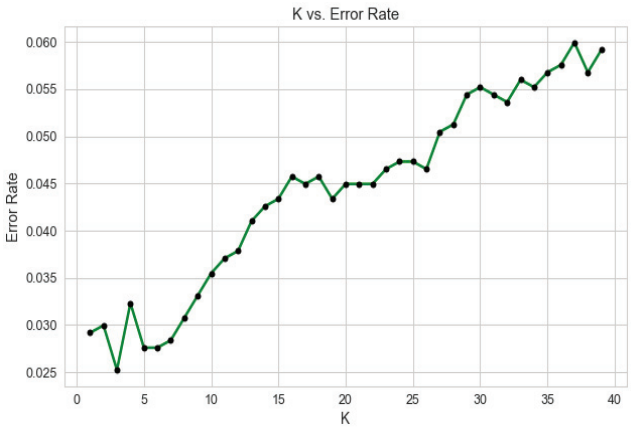

Fig. 2 Growth of Error Rate with respect to K

It can be inferred from the plot that there has been no significant change in the error rate corresponding to an increase or decrease in the value of K. That is, the magnitude of difference between the worst and the best error rate is close to negligible. Hence, a low value of K wouldn't hurt the accuracy of the model. This model, indeed, performs exceptionally well both on training and test data.

At K=2, it achieves a training accuracy of 98.8% and test accuracy of 98%. However, this accuracy is achieved only after transforming the data to a standard scale using the StandardScaler class of Scikit Learn's pre-processing package. StandardScaler normalizes the features by removing the mean and scaling to unit variance. Without standardizing the features, the model performs poorly with a training accuracy of 86% and a test accuracy of 72%. It misclassifies approximately 32% of the data points from the test data. Hence, it can be concluded that a standardized dataset can significantly boost KNN's performance.

Next, Support Vector Machine is trained. The classification accuracy for SVM depends upon the choice of a hyperplane. A performant model will have a hyperplane with the largest minimum margin, and a hyperplane that correctly separates as many data points as possible. Two important parameters are used in this study, C (cost) and gamma. Gamma is known as the kernel coefficient for rbf (radial basis function), poly and sigmoid, while C determines the margin separating the hyperplane. The default value of gamma is 1/no_of_features. Initially, the model is trained with the default parameter values {'C': 1.0, 'gamma': auto}. With these values, it achieves an overall 73% precision and recall. Exhaustive grid search is then run, in order to introduce and try various combinations of parameter values to estimate the labels. This can have a big impact on the performance of the model. The best values of the estimator found by grid search are {'C': 100, 'gamma': 0.01} which gives a training accuracy of 89.5% and a test accuracy of

92%. This is a striking boost to the previously trained SVM with default parameter values.

TABLE I. SVM Accuracy with parameter values {C: 0.001, gamma: 0.1}

	Precision	*Recall*	*F1-score*	*Support*
Female	0.49	1.00	0.65	463
Male	0.00	0.00	0.00	488
Avg / total	0.24	0.49	0.32	951

TABLE II. SVM Accuracy with parameter values {C: 100, gamma: 0.01}

	Precision	*Recall*	*F1-score*	*Support*
Female	0.94	0.90	0.92	463
Male	0.91	0.94	0.92	488
Avg / total	0.92	0.92	0.92	951

$$C \propto \text{variance/bias} \qquad \text{Gamma} \propto \text{bias/variance} \qquad (1)$$

The above two classification reports show how the performance of SVM varies with the value of C and gamma. A large value of gamma yields high-bias and low-variance models whereas a large value of C results in a model with low-bias and high-variance. It is difficult to achieve a trade-off between these two parameters. Grid search is run in order to find the optimum values of these two parameters.

KNN and SVM have been applied to the entire feature-set. However, this might have resulted in an overly flexible fit. Individual relationships cannot be found out from these models despite being greatly performant. Despite of the previous performant models, a minimized set of features could still be able to give reliable results. This will not only reduce the complexity of the model but will also help in understanding the significance of the remaining features in determining the gender of a person.

For feature selection, Variance Threshold method is performed on the feature-set. This approach removes features with variance lower than some specified threshold value. This threshold value is calculated by the formula:

$$\text{Var}[X] = P(1 - P) \qquad (2)$$

Where P is the probability of occurrence.

This method gave the following features: skew, kurt, meandom, maxdom, dfrange. SVM was then trained using this feature-set and the results were somewhat less extraordinary. This model achieved a training accuracy of 75% and a test accuracy of 71% using default parameter values. Plotting various distribution plots further indicated that these features were not able to explain the difference between genders. Figure 3 indicates that dfrange values are insufficient to explain gender-specific nuances.

Hence, this feature-selection approach yielded poor results or at least no improvement over the previous models. The next approach for feature-selection was to eliminate each feature manually.

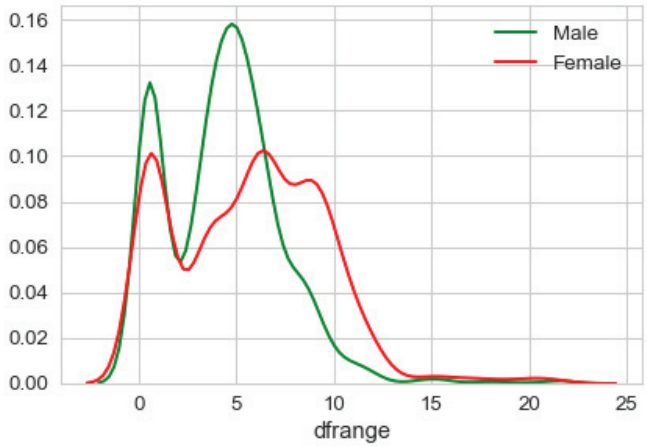

Fig.3. Distribution plot of dominant frequency measured across acoustic signal

Box-plots of each feature by gender (male/female) gave a deeper comprehension of feature importance. These plots show an overall distribution of the values of that specific feature. Box-plots depict groups by their quartiles. If there is a considerable difference between the quartiles or spread of each group (male and female in our case), then it can be inferred that the plotted feature can serve a useful purpose in the model. Below are the different box-plots plotted using seaborn [11].

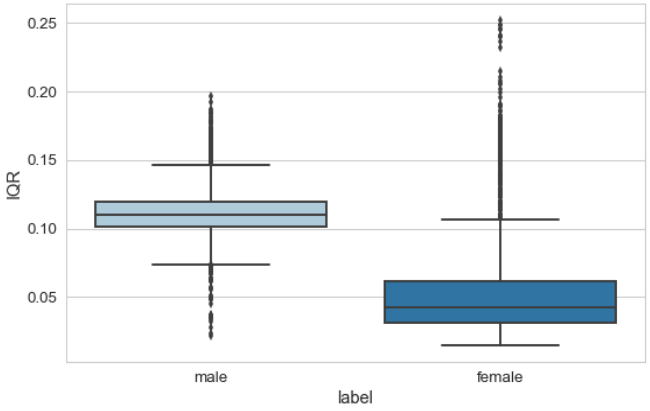

Fig. 4 Interquartile Range (IQR)

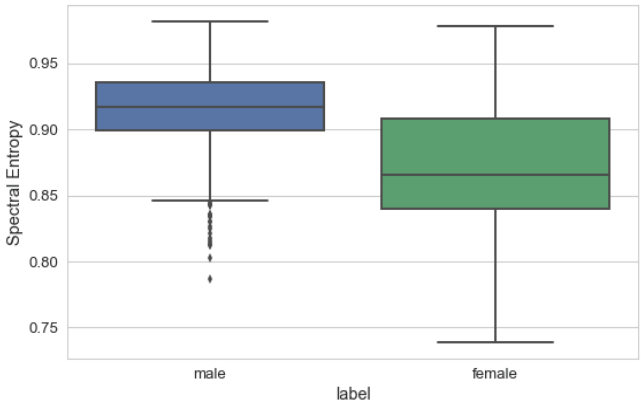

Fig. 5 Spectral Entropy (sp.ent)

Although there are some outliers present in both of the features, their combined impact on the model is significant. This process narrowed down the useful feature-space from 20 (initially) to 5. The features that were thought to be the

most ambitious in gender determination are sd (standard deviation of frequency), Q25 (first quartile), IQR (Inter-quartile range), sp (Spectral Entropy) and meanfun (mean fundamental frequency). Training SVM with default parameters on this feature-set yielded 89.9 % training accuracy and 91 % test accuracy.

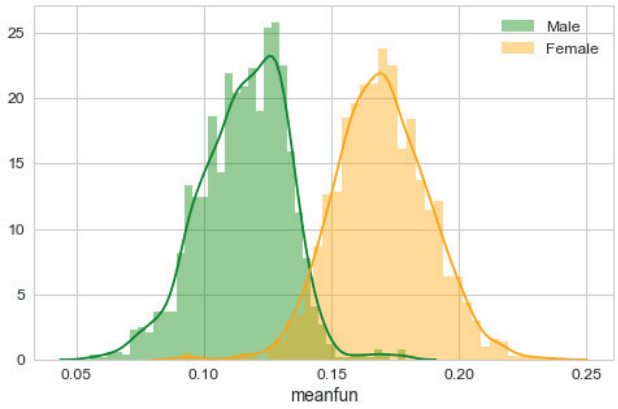

Fig. 5 Distribution plot of mean fundamental frequency by gender

The above plot represents the distribution of meanfun with y-axis indicating the count of values. It clearly indicates that mean fundamental frequency would be extremely helpful for calculating the response values. Running exhaustive grid search on the new feature-set and after tuning the hyper-parameters to the values {C: 10, gamma = 1}, SVM achieved 96.6 % training accuracy and 97 % test accuracy. It misclassified only 21 out of 951 data points. [12]

VI. Conclusion

In conclusion, most of the above models have performed well but in some cases, interpretability outweighs inference. In order to understand gender-specific characteristics, it is important to eliminate all the insignificant features from the model. By the end of the experimental study, it can be concluded that a great level of accuracy can be achieved by selecting some specific features. This will reduce the overall model training time, model complexity and also increase inference simplicity.

VII. Future Work

One of the obstructions in classification using audio-cues is that the audio samples are usually obtained from noisy environments. Such environmental or artificially generated noise detrimentally limits the accuracy of classification. Different and more efficient ways to eliminate noise can be found out and this becomes a course for future research. [13]

References

[1] T. Vogt and E. Andrè, “Improving automatic emotion recognition from speech via gender differentiation”, in Proc. “Language Resources and Evaluation Conference”, Genoa, 2006.

[2] WableR, Araya-Salas, M. and Smith-Vidaurre, G. (2017), warbleR: an r package to streamline analysis of animal acoustic signals. Methods Ecol Evol. 8, 184-191.

[3] Dataset, https://www.kaggle.com/primaryobjects/voicegender

[4] Python, https://docs.python.org/3/faq/general.html

[5] Matplotlib, John D. Hunter. Matplotlib: A 2D Graphics Environment, Computing in Science & Engineering, 9, 90-95 (2007), DOI:10.1109/MCSE.2007.55, https://matplotlib.org

[6] Pandas, Wes McKinney. Data Structures for Statistical Computing in Python, Proceedings of the 9th Python in Science Conference, 51-56 (2010), https://pandas.pydata.org

[7] NumPy, Travis E, Oliphant. A guide to NumPy, USA: Trelgol Publishing, (2006), http://www.numpy.org

[8] K. Chen, “Gender identification by voice”, Stanford University.

[9] YAAFE, an Easy to Use and Efficient Audio Feature Extraction Software, B.Mathieu, S.Essid, T.Fillon, J.Prado, G.Richard, proceedings of the 11th ISMIR conference, Utrecht, Netherlands, 2010.

[10] Kory Becker, “Identifying the Gender of a Voice using machine learning”, 2016, unpublished.

[11] Seaborn, Michael Waskom, Olga Botvinnik, Drew O'Kane, Paul Hobson, Joel Ostblom, Saulius Lukauskas, … Adel Qalieh. (2018, July 16). mwaskom/seaborn: v0.9.0 (July 2018) (Version v0.9.0). Zenodo.

[12] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[13] “Multimodal gender classification using Support Vector Machines”, Rajeev Sharma, Mohammed Yeasin and Leena A. Walavalkar, State College, PA (US).