CENG 480 Machine Learning

Term Project

VOICE RECOGNATION

Name/Surname : Nurseli BAL

Student ID: 201611656

E-mail : balnurselee@gmail.com

Çankaya University, Computer Engineering Department, Turkey

CONTENTS

Abstract..........3

Keywords..........3

Introduction..........4

Related Works..........5

Methodology..........6

Dataset 7-8

Preprocessing 9

Classification Methods 10

Results.......... 10-13

Conclusion.......... 14

Referances.......... 14

ABSTRACT

From the begining of the world until today, people need to speak to communicate each other, to explain their feelings. Every single person from, since the cradle, they try speak and learn things. Speech is one of the most important thing in the world. Speech and voices has attracted the attention of people. In this experiment one of these attention of people will be discussed. It is voice. Collected dataset about gender voice will be examined. In this study, I will be using a software called orange. I used the gender voice dataset[1] to analyze. My dataset has a label which is categorical and labeled as male and female, and 20 different numerical attributes. Using this dataset and the software, we will distinquish the gender according to the features. I firstly prepared the dataset to preprocess the data. I selected important features to get them into analyze, then I cleaned the outliers. After that, in order to classify my data, I used 4 different classificaton methods. Those are SVM(Support Vector Machine), Navie Bayes, kNN (k Nearest Neighbor), Random Forest. After that, I tested and scored to see the classification metrices. Finally, I used the confusion matrix to see the distribution. To see the difference between data after preprocess and data before preprocess, I compare the results in which the data with preprocessing has higher accuracy. The data also will be visualized to understand the distribution better. This experiment will show us how a machine can distinguish a human voice weather it is a women voice or a man voice.

KEYWORDS

Data, dataset, workflow, classification, preprocess, attributes, features, accuracy, classification metric, confusion matrix, cross vlidation, sampling, visualization, Feature selection, outlier, ranking, widget, numerical, categorical,

1 Introduction

Speech is the most important communication in this world. People understand their emotions by speaking each other. When it is important that much, collecting data for voice recognition is also an important topic. Indeed, the best recognizer is the ear. When this is the case, a question comes to people's mind. Can a machine recognize a voice of a human to perceive the difference in human voices? Science can answer this question. There are a number of difference between man voice and a woman voice. When we handle those difference, we can give the answer. In this report, a collected data called Voice will be dealed. In this data, certain information was collected, and in terms of those information, they distinguish that the voice belongs to a woman or a man. Those values specify the person's gender by making certain calculation. Figure 1 explains those basic calculations in this experiment.

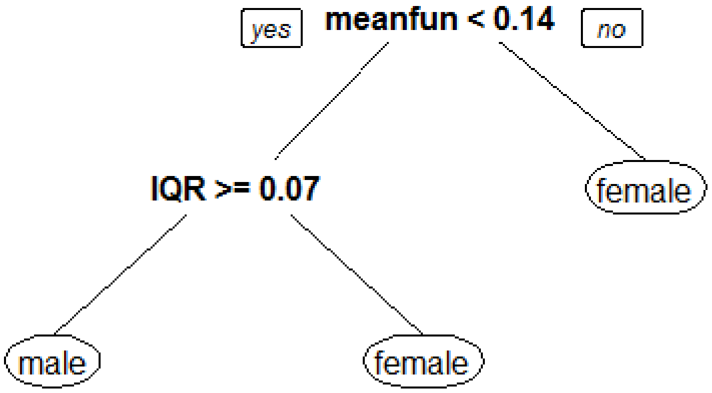


Fig. 1. A figure that shows a tree structure to explain the distinction[2].

We will see the details about the parameters as meanfun and IQR given above in the up-coming sections. Gender recognition can be useful for some applications such as human to machine interaction, muting sounds for a gender, audio/video categorization with tagging, etc. [3].

2 Related Works

During the last decades, Gender recognition is in demand by Machine Learning study. In this chapter, I will focus on the related works of the Gender Recognition.

Niraj Verma [4] from India, is one of the analyst who used Voice dataset, classify the data using Support Vector Machine. He used the Python programming language. He had 95% of accuracy score in his analysis. He also used k-Fold cross validation where k=10 as sampling.

Enes Polat [5] from Turkey, also used the same dataset for his analysis. He also used Python programming language. He used normalization technique as a preprocessing technique. The classification method that he used is Logistic Regression. He has a test accuracy as 98.11 %.

Hakan Özen [6] from Turkey, preferred to work on the Voice data. The programming language that he used is Python. He use 3-layer ANN method for the classification. He has the results as train accuracy: 98.06% and as test accuracy: 97.056%.

Sheik Mohamed Imran [7] from India preferred to use R language to analyse the Voice data. He used 80% of the data as a test set and the remaining as a train set. He performed PCA for preprocessing. In order to classify the dataset, he used decision tree classification method. The accuracy value he reached is 97.4%.

Giorgiogarziano [8] from Italy, is the last person that will be mentioned here. He used R language. The library that he used is caret. He used many classificaiton method like C5.0, Multilayer Neural Network method, Random Forest using caret, etc. He had the accuracy values in Random Forest as 98%, in Multilayer Neural Network as 97%, in C5.0 as 97%.

As we can see, with different methods and languages, they tried to get better accuracy results in their analysis.

3 Methodology

In this experiment, Voice Dataset is used. Using this data, the analysis will answer us weather the voice belongs to a woman or a man. In order to analyse the data, I preferred to use Orange software. Because it is easy and fast.

Figure 2 shows the workflow of the methodology.

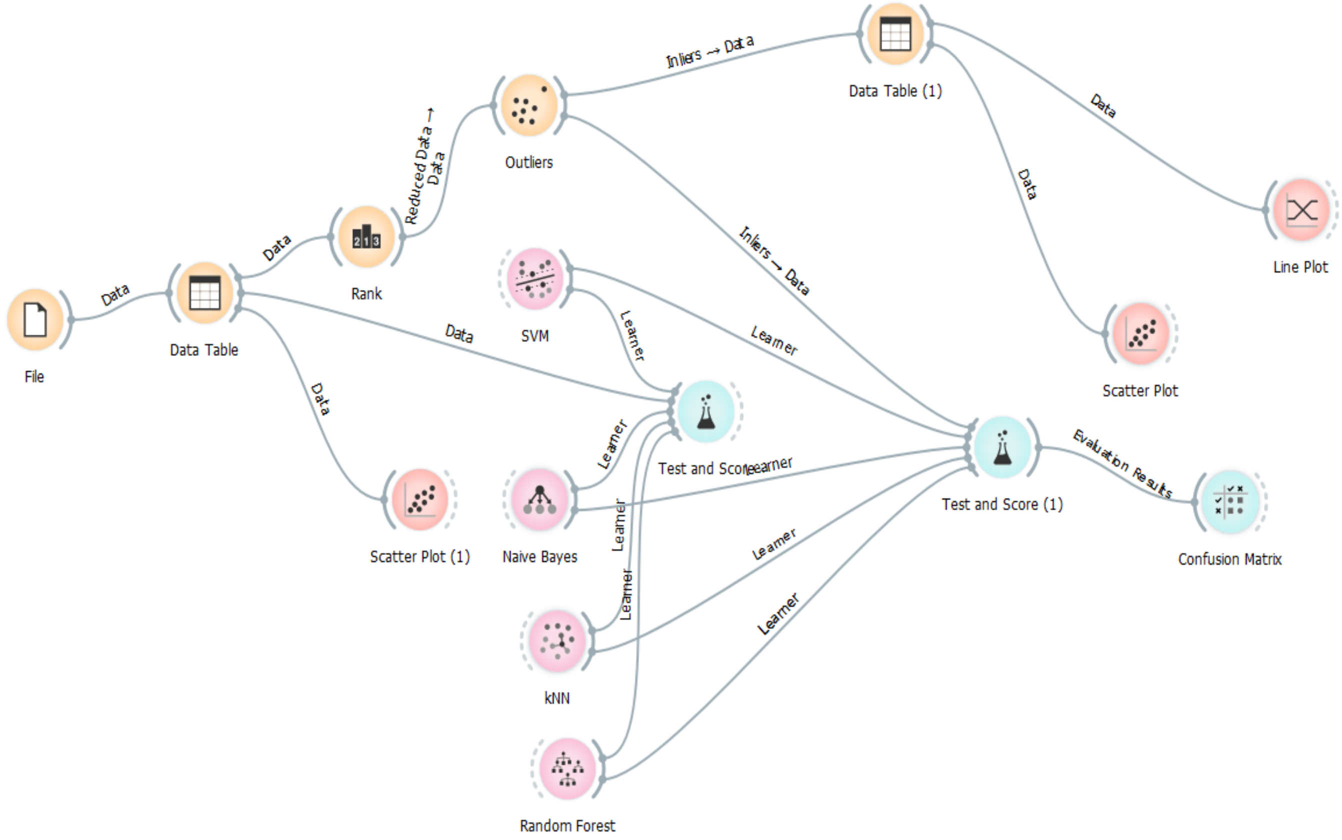


Fig. 2. A figure that shows the workflow of the methodology.

In the above workflow, I used some widgets which are components of orange to use in the analysis. File widget is used for uploading data. Data table is used to show the data. Rank is used for Feature Selection. Outlier is for capturing the outliers and deal with them. To visualize our data, certain visualization techniques available in Orange software are used. Those are Scatter Plot and Line Plot. In order to see the classification metrics, we shall use Test and Score widget.

In this analysis, 2 different preprocessing tecnique has been used. Those are Feature Selecetion and Outlier detection. The reason why I used Feature Selection technique that my data have lots of unnecessary data. Using this technique, I disposed of the unnecessary data. Also the reason why I used Outlier Detection that the data contains outliers. We will see that when we use those techniques, the accuracy will be better. Besides, 4 different classification method have been used. Those are SVM(Support Vector Machine), Navie Bayes, kNN (k Nearest Neighbor), and Random Forest. Finally Confusion matrix is used to see how the classification methods seperate the data as a female and male.

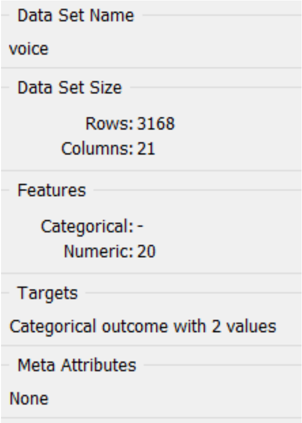
3.1 Dataset

In this experiment, I have used the voice dataset. The source of the data is http://festvox.org/ [9] and they use the tool http://www.voxforge.org/ [10] to collect it. The purpose of collecting this data is to distinguish the human voice weather is a woman voice or a man voice. In order to pre-process the voice samples, the collectors used acoustic ananlysis using R language. The packages that was used are seewave and tuner with an analyzed frequency range of 0hz-280hz[11]. Our dataset contains 21 different attributes being one attribute is the label attribute which categorical. Excluding the label attribute, all of the features are numerical. All of the features will be explained in below.

Name,type,description

meanfreq, float, mean frequency (in kHz)
sd, float, standard deviation of frequency
median, float, median frequency (in kHz)
Q25, float, first quantile (in kHz)
Q75, float, third quantile (in kHz)
IQR, float, interquantile range (in kHz)
skew, float, skewness (see note in specprop description)
kurt, float, kurtosis (see note in specprop description)
sp.ent, float, spectral entropy
sfm, float, spectral flatness
mode, float, mode frequency
centroid, float,frequency centroid (see specprop)
meanfun,float,average of fundamental frequency measured across acoustic signal
minfun,float,minimum fundamental frequency measured across acoustic signal
maxfun,float,maximum fundamental frequency measured across acoustic signal
meandom,float,average of dominant frequency measured across acoustic signal
mindom,float,minimum of dominant frequency measured across acoustic signal
maxdom,float,maximum of dominant frequency measured across acoustic signal
dfrange,float,range of dominant frequency measured across acoustic signal
modindx,float,modulation index. Calculated as the accumulated absolute difference between adjacent measurements of fundamental frequencies divided by the frequency range
label,string,"Predictor class, male or female" .

Figure 3 shows the general information about the Voice Dataset.



Data Set Name
voice
Data Set Size
Rows: 3168
Columns: 21
Features
Categorical: -
Numeric: 20
Targets
Categorical outcome with 2 values
Meta Attributes
None

Fig. 3. A figure shows the info of the dataset.

The detailed information will be given below.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3168 entries, 0 to 3167
Data columns (total 21 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   meanfreq  3168 non-null   float64
 1   sd        3168 non-null   float64
 2   median    3168 non-null   float64
 3   Q25       3168 non-null   float64
 4   Q75       3168 non-null   float64
 5   IQR       3168 non-null   float64
 6   skew      3168 non-null   float64
 7   kurt      3168 non-null   float64
 8   sp.ent    3168 non-null   float64
 9   sfm       3168 non-null   float64
 10  mode      3168 non-null   float64
 11  centroid  3168 non-null   float64
 12  meanfun   3168 non-null   float64
 13  minfun    3168 non-null   float64
 14  maxfun    3168 non-null   float64
 15  meandom   3168 non-null   float64
 16  mindom    3168 non-null   float64
 17  maxdom    3168 non-null   float64
 18  dfrange   3168 non-null   float64
 19  modindx   3168 non-null   float64
 20  label     3168 non-null   object 
dtypes: float64(20), object(1)
memory usage: 519.9+ KB
None
```

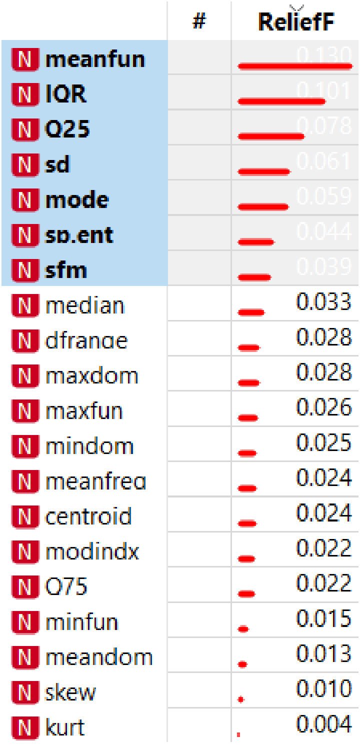
The information is achieved using this line of code in Python language:

print(data.info())

3.2 Preprocessing

Unfortunately sometimes, datasets are not good enough to analyse. They may have missing values, unnecessary features, outliers, etc. When this is the case, they before need to be pre-processed. Preprocessing means preparing our data to analyse. In this experiment, two different preprocessing techniques that were used in the workflow will be discussed.

First one is Ranking. We sometime may have data that are not necessarily in our dataset. Figure 4 below shows the order of importance of the features. I used ReliefF scoring method for ranking.



	#	ReliefF
N **meanfun**		0.130
N **IQR**		0.101
N **Q25**		0.078
N **sd**		0.061
N **mode**		0.059
N **sp.ent**		0.044
N **sfm**		0.039
N median		0.033
N dfrange		0.028
N maxdom		0.028
N maxfun		0.026
N mindom		0.025
N meanfreq		0.024
N centroid		0.024
N modindx		0.022
N Q75		0.022
N minfun		0.015
N meandom		0.013
N skew		0.010
N kurt		0.004

Fig. 4. A figure that shows the order of importance of the features.

By looking at the Figure 4, we can easily say that, some of the attributes are not necessary to use.

The second preprocessing technique that was used is Outlier. Specifying outliers will give us better results which we will see in the Result section. The reason why I have used that preprocessing technique is that my data has outliers.

I did not use any imputation technique because my dataset does not include any missing values.

3.3 Classification Methods

After preprocessing our data, we can use classification methods. There are 4 methods that was used in this experiment. Those are

SVM(Support Vector Machine),

Navie Bayes,

kNN (k Nearest Neighbor),

Random Forest.

The reason why I have chosen those methods in my analysis is that they work well in high dimensional data. They fit perfectly on the data. I did not change any of their parameters. I used the default parameters. They fit perfectly as well.

4 Results

After preprocessing and making some classification methods, we are ready to see the accuracy result in our work. I have used the CA (Accuracy Classification Score), Precision, Recall classification metrics[12] to see the accuracy results. The result comes from the unseen records.

Before coming to the final result, I would like to show the results before the preprocessing. This will help us to see importance of preprocessing in the analysis. Table 1 below shows the results.

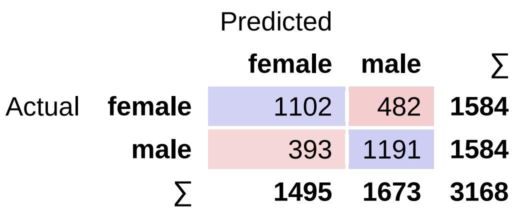
Sampling type: Stratified 10-fold Cross validation
Target class: Average over classes

Scores

Model	Train time [s]	Test time [s]	CA	F1	Precision	Recall
kNN	0.272	0.173	0.7196969696969697	0.7194705609942597	0.7204085157835645	0.7196969696969697
SVM	2.401	0.079	0.9738005050505051	0.9737993537723962	0.9738837964374766	0.9738005050505051
Random Forest	0.755	0.050	0.9753787878787878	0.9753786995620249	0.9753856087237248	0.9753787878787878
Naive Bayes	0.246	0.035	0.8873106060606061	0.8868820274531442	0.8932706631361563	0.8873106060606061

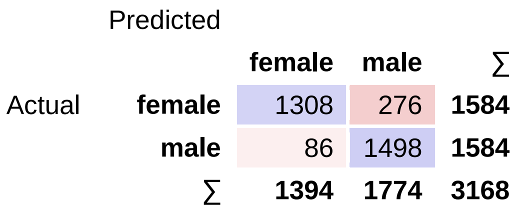
Table 1. A table that shows the classification metrics results.

Using confusion matrix, we can see how the program separate the data before the preprocessing in the Table 2-3-4-5.



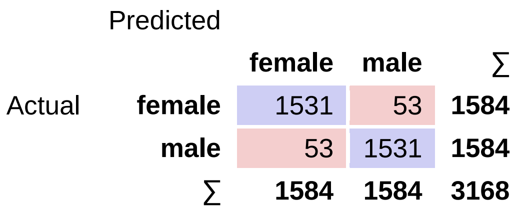
		Predicted		
		female	**male**	**∑**
Actual	**female**	1102	482	**1584**
	male	393	1191	**1584**
	∑	**1495**	**1673**	**3168**

Table 2. Confusion matrix for kNN (showing number of instances)



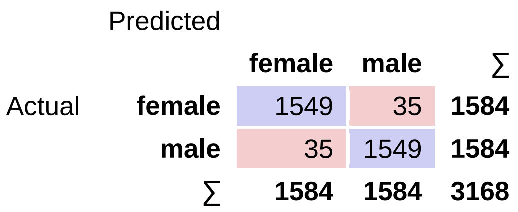
		Predicted		
		female	**male**	**∑**
Actual	**female**	1308	276	**1584**
	male	86	1498	**1584**
	∑	**1394**	**1774**	**3168**

Table 3. Confusion matrix for Naïve Bayes (showing number of instances)



		Predicted		
		female	**male**	**∑**
Actual	**female**	1531	53	**1584**
	male	53	1531	**1584**
	∑	**1584**	**1584**	**3168**

Table 4. Confusion matrix for SVM (showing number of instances)



		Predicted		
		female	**male**	**∑**
Actual	**female**	1549	35	**1584**
	male	35	1549	**1584**
	∑	**1584**	**1584**	**3168**

Table 5. Confusion matrix for Random Forest (showing number of instances)

Now, we will see the visualized data before the preprocessing in Figure 5.

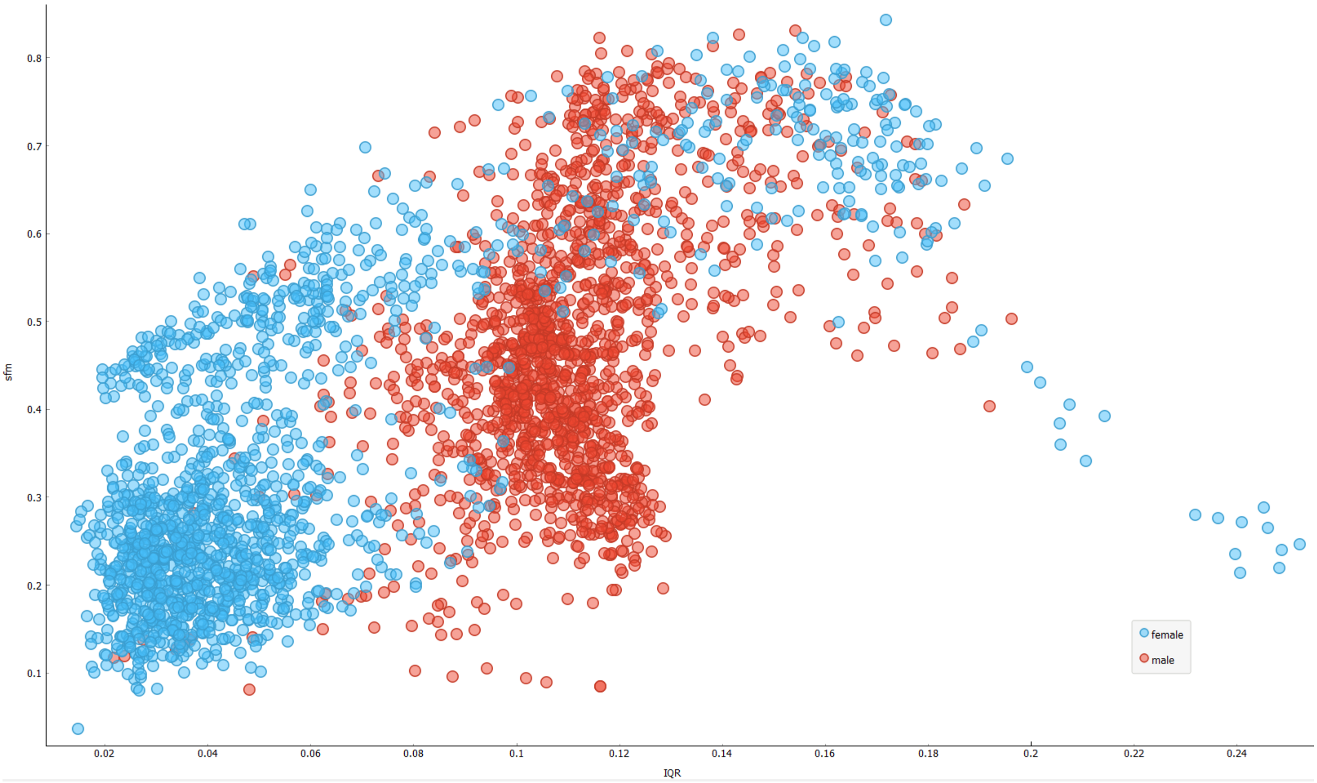


Fig. 5. A figure that shows the distribution of the data before the preprocessing as IQR in x axes, sfm in y axes

As we can see in the Table 1, results are 72%, 97%, 97%, 88%. The results are good enough even in this state. However, using preprocessing, they were made better. We can see the results after preprocessing in the Table 6. I also used Cross Validation with k=10 as sampling.

Sampling type: Stratified 10-fold Cross validation
Target class: Average over classes

Scores

Model	Train time [s]	Test time [s]	CA	F1	Precision	Recall
kNN	0.090	0.088	0.9864690721649485	0.9864677066958452	0.986532722347536	0.9864690721649485
SVM	0.454	0.026	0.9890463917525774	0.9890461052069137	0.9890532114400742	0.9890463917525774
Random Forest	0.271	0.034	0.9851804123711341	0.9851797046764784	0.9851994539154811	0.9851804123711341
Naive Bayes	0.082	0.013	0.9774484536082474	0.9774428800725684	0.9776721148160527	0.9774484536082474

Table 6. The results of the classification metrics

By looking at the Table 6, we can easily say that results are better. The CA that shows the subset accuracy, Precision that shows the exactness, and Recall that shows the completeness values has increased; and the Train time and the Test time has decreased as needed.

Using confusion matrix, we can see how the program separate the data after the preprocessing in the Table 7-8-9-10.

		Predicted		
		female	**male**	∑
Actual	**female**	755	11	**766**
	male	5	781	**786**
	∑	**760**	**792**	**1552**

Table 7. Confusion matrix for kNN (showing number of instances)

		Predicted		
		female	**male**	∑
Actual	**female**	746	20	**766**
	male	6	780	**786**
	∑	**752**	**800**	**1552**

Table 8. Confusion matrix for Naïve Bayes (showing number of instances)

		Predicted		
		female	**male**	**∑**
Actual	**female**	753	13	**766**
	male	5	781	**786**
	∑	**758**	**794**	**1552**

Table 9. Confusion matrix for SVM (showing number of instances)

		Predicted		
		female	**male**	**∑**
Actual	**female**	761	5	**766**
	male	6	780	**786**
	∑	**767**	**785**	**1552**

Table 10. Confusion matrix for Random Forest (showing number of instances)

Now, we will see the visualized data after the preprocessing in Figure 6.

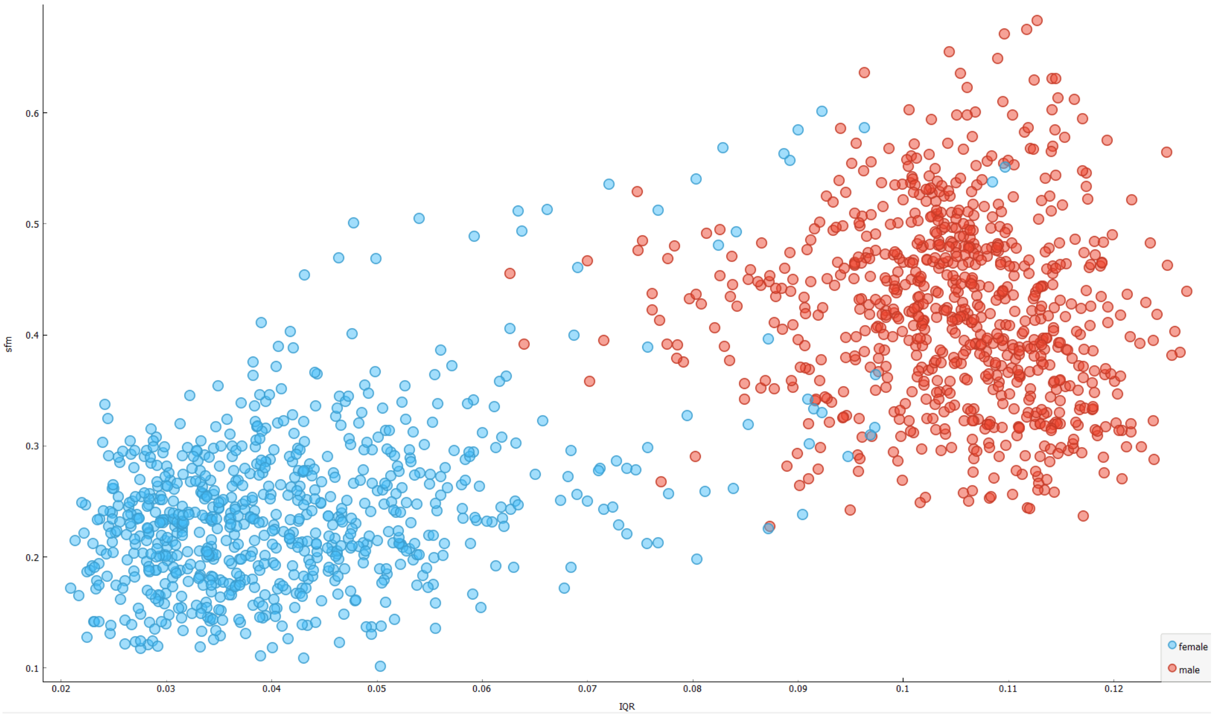


Fig. 6. A figure that shows the distribution of the data after the preprocessing as IQR in x axes, sfm in y axes

Let’s now see the line plot of the distribution of the all features that we used after ranking in Figure7.

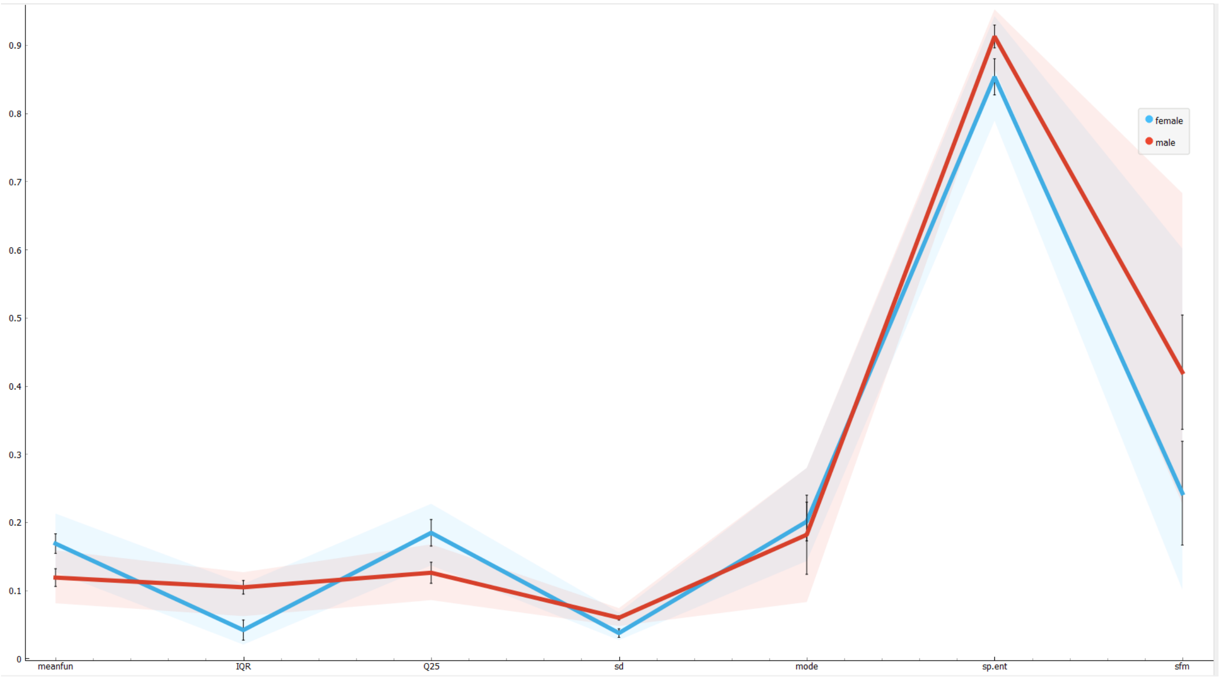


Fig. 7. A Line-Plot graphic that shows all of the data distributions.

5 Conclusion

In this work, we have seen how to recognize a which voice belong to which gender. We have used some features to make the calculations but we also saw that not all of the features are necessary to analyse the voice gender. We firstly preprocess our data to make it better. Then, we have used 4 different classification methods to classify our data. After that, we have seen the results on classification metrics. We have had really good results on this experiments. After all, we have used some visualization techniques as Scatter Plot and Line Plot to get a better understanding of the data distribution. Finally, in order to see the distribution on a table we have used confusion matrix and this answered us how good distribution we have made.

References

[1] Kaggle.com. 2020. *Gender Recognition By Voice*. [online] Available at: <https://www.kaggle.com/primaryobjects/voicegender> [Accessed 22 May 2020].

[2] Kaggle.com. 2020. *Gender Recognition By Voice*. [online] Available at: <https://www.kaggle.com/primaryobjects/voicegender> [Accessed 21 May 2020].

[3] Mdpi.com. 2020. [online] Available at: <https://www.mdpi.com/2504-4990/1/1/30/pdf> [Accessed 21 May 2020].

[4] Kaggle.com. 2020. *Niraj Verma | Kaggle*. [online] Available at: <https://www.kaggle.com/nirajvermafcb> [Accessed 22 May 2020].

[5] Kaggle.com. 2020. *Enes Polat | Kaggle*. [online] Available at: <https://www.kaggle.com/enespolat> [Accessed 22 May 2020].

[6] Kaggle.com. 2020. *Hakan Ozen | Kaggle*. [online] Available at: <https://www.kaggle.com/hakanozen> [Accessed 22 May 2020].

[7] Kaggle.com. 2020. *Sheik Mohamed Imran | Kaggle*. [online] Available at: <https://www.kaggle.com/imrandude> [Accessed 22 May 2020].

[8] Kaggle.com. 2020. *Giorgiogarziano | Kaggle*. [online] Available at: <https://www.kaggle.com/giorgiogarziano> [Accessed 22 May 2020].

[9] Festvox.org. 2020. *Festvox: Home*. [online] Available at: <http://festvox.org/> [Accessed 21 May 2020].

[10]Voxforge.org. 2020. *Free Speech... Recognition (Linux, Windows And Mac) - Voxforge.Org*. [online] Available at: <http://www.voxforge.org/> [Accessed 21 May 2020].

[11] Kaggle.com. 2020. *Gender Recognition By Voice*. [online] Available at: <https://www.kaggle.com/primaryobjects/voicegender> [Accessed 21 May 2020].

[12] Docs.biolab.si. 2020. *Scoring Methods (Scoring) — Orange Data Mining Library 3 Documentation*. [online] Available at: <https://docs.biolab.si//3/data-mining-library/reference/evaluation.cd.html> [Accessed 22 May 2020].