Daniel Youssef

206150

AI Group 1

**Gender Voice Classification**

**Description**

Dataset Source: <https://www.kaggle.com/datasets/primaryobjects/voicegender>

This dataset consists of several acoustic properties that make up human sounds. In order to classify these voices correctly, three different classification algorithms will be used to train three models, and three evaluation scores will be used to compare the models in order to find the best possible one.

**Features**

In this dataset, there are 21 features (including the target class). Here are all features:

1. meanfreq

2. sd

3. median

4. Q25

5. Q75

6. IQR

7. skew

8. kurt

9. sp.ent

10. sfm

11. centroid

12. peakf

13. meanfun

14. minfun

15. maxfun

16. meandom

17. mindom

18. maxdom

19. dfrange

20. modindx

21. label

For this dataset, these will be the goals of my notebook:

1. Finding out the difference(s) that separate male voices from female voices
2. Making sure the data is clean enough to not cause any errors or inaccuracies for training and testing
3. Coming up with a machine learning model that has the highest accuracy for classifying different voices

These were the following steps that were taken:

1. Importing Libraries

2. Data Cleaning

3. Data Visualization

4. Model Training (No Feature Selection)

5. Model Testing (No Feature Selection)

6. Feature Selection

7. Model Training (With Feature Selection)

8. Model Testing (With Feature Selection)

9. Comparison And Analysis of Results

10. Conclusion

**Result of Data Cleaning**

* **Target classes are balanced**
* **Outlier removal will affect scores negatively, and should be kept.**
* **Basic biological facts about men and women’s voices were confirmed (i.e., no “incorrect” data).**

**Feature Selection**

In order to begin selecting the most relevant features, first it is important to decide on which feature selection method to use.

There are three feature selection methods:

1. Filter Methods
2. Wrapper Methods
3. Embedded Methods

\*A Filtering Method was picked.

**Modeling**

Three different models will be trained using three different algorithms:

1. Decision Tree

2. K-Nearest Neighbour

3. Logistic Regression

A 70/30 train-test split was chosen for this dataset.

The following features were removed due to feature selection:

{'IQR', 'Q25', 'centroid', 'dfrange', 'kurt', 'maxdom', 'median', 'sfm'}

**Modeling Results**

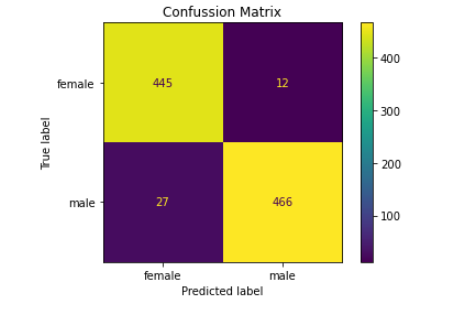
* **Decision Tree with no FS: 97.2% (Best)**
* Decision Tree with FS: 95.9%
* KNN with no FS: 69.5%
* **KNN with FS: 79.7%**
* Logistic Regression with no FS: 95.5%
* **Logistic Regression with FS: 95.7%**

Decision Tree with no FS Confusion Matrix:

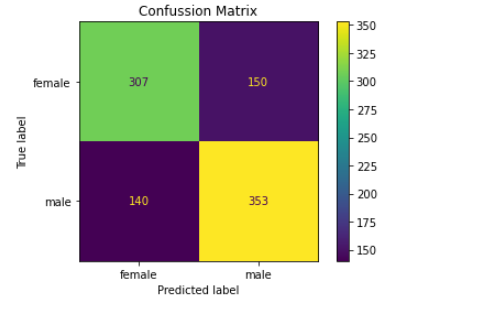
Chart, treemap chart

Description automatically generated

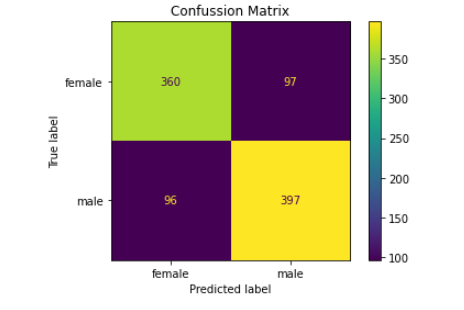
Decision Tree with FS Confusion Matrix:



KNN with no FS Confusion Matrix:



KNN with FS Confusion Matrix:

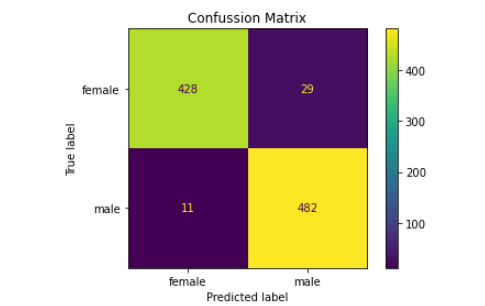


Logistic Regression with no FS Confusion Matrix:

Chart, treemap chart

Description automatically generated

Logistic Regression with FS Confusion Matrix:

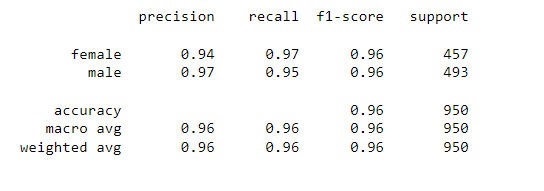


Decision Tree with no FS Classification Report:

Table

Description automatically generated with medium confidence

Decision Tree with FS Classification Report:

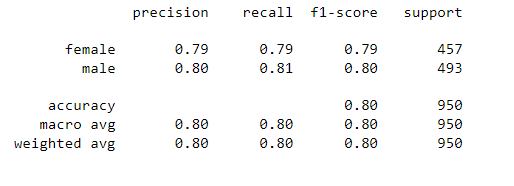


KNN with no FS Classification Report:

Table

Description automatically generated

KNN with FS Classification Report:



Logistic Regression with no FS Classification Report:

Table

Description automatically generated with medium confidence

Logistic Regression with FS Classification Report:

Table

Description automatically generated with medium confidence

**Conclusion**

**\* Removing outliers for mean frequencies and mean fundamental frequencies can produce less accurate results, as these values rely on a multitude of factors, which cannot be measured with the current feature set.**

**\* Decision Tree was the best algorithm for this problem.**

**\* KNN may have performance on a different feature selection method.**

**\* It is crucial to compare models before and after feature selection, as some irrelevant features can cause some models to perform worse.**