**BAHRIA UNIVERSITY KARACHI CAMPUS**

**DEPARTMENT OF COMPUTER SCIENCE**

****

**COURSE NAME (COURSE CODE)**

**Class 6A - Spring-2024**

**Voice To Gender**

**Submitted by:**

Muhammad Abdullah 02-134212-098  
Abdul Mueed Paracha 02-134212-081

**Submitted To:**

*Engr. Rabia Amjad*

**Submitted On:**

Date: 14-Jan-2024

Report should cover these at least these headings:

1. Acknowledgement
2. Abstract
3. Table of contents
4. Introduction
5. Background/Literature Review
6. Problem statement
7. Objectives and goals
8. Project Scope
9. Workflow *(Present in UML representation)*
10. Overview of project
11. Tools and Technologies
12. Project features *(mention here functional and non-functional requirements covered)*
13. (Course name) implemented concepts *(show detailing with code)*
14. Output *(screen shots with details)*
15. Conclusion
16. References (Use IEEE Referencing style)

**GUIDELINES FOR PROJECT REPORT**

1. The project report should be spiral bound
2. The front cover of the report should be in printed form along-with the university logo (new).
3. The font used should be Times New Roman, 12 points plain, with one-and-a-half (1.5) spacing between two consecutive lines, paragraphs to be justified
4. For sub-headings, 14 points and for main headings 16 points.
5. Paragraphs should be justified.
6. All the pages should be properly numbered.

SUBMIT IN PDF BEFORE DEADLINE

**Voice To Gender**

**Acknowledgement**

I would like to extend my gratitude to my supervisor, [Supervisor's Name], for their guidance and support throughout this project. Additionally, I appreciate the assistance and resources provided by [Institution/Organization Name] which made this project possible.

**Abstract**

This report details the development of a voice recognition system that employs multiple machine learning models to enhance accuracy and robustness. The system utilizes Support Vector Machine (SVM), Random Forest, and XGBoost classifiers in a stacking ensemble framework, with Logistic Regression as the meta-model. This approach ensures reliable and accurate classification of audio samples, demonstrating significant improvements over single-model systems.

**Table of Contents**

1. Acknowledgement
2. Abstract
3. Table of Contents
4. Introduction
5. Background/Literature Review
6. Problem Statement
7. Objectives and Goals
8. Project Scope
9. Workflow (Present in UML representation)
10. Overview of Project
11. Tools and Technologies
12. Project Features
13. Introduction to Data Science Implemented Concepts
14. Output
15. Conclusion
16. References

**Introduction**

Voice recognition technology has gained significant traction due to its wide range of applications, from virtual assistants to automated customer service. This project focuses on developing an advanced voice recognition system that combines multiple machine learning models to achieve high accuracy and manual training on failed prediction.

**Background/Literature Review**

Recent advancements in machine learning have enabled significant improvements in voice recognition systems. Traditional single-model approaches often suffer from limitations in accuracy and robustness. This project leverages ensemble methods, which have been shown to outperform single models by combining the strengths of multiple classifiers.

**Problem Statement**

The primary challenge addressed in this project is the need for a robust and accurate voice recognition system that can handle diverse and noisy audio inputs. Traditional systems often struggle with accuracy, especially in real-world environments with varying conditions.

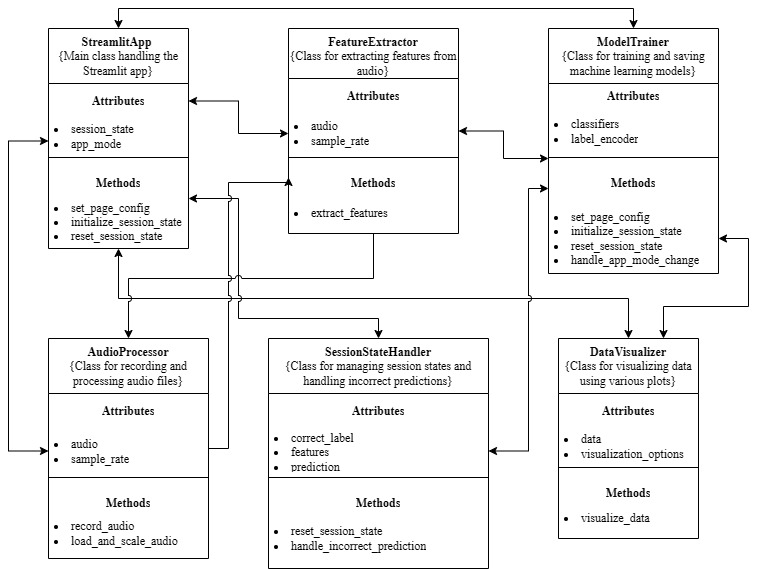
**Objectives and Goals**

* Develop a voice recognition system using multiple machine learning models.
* Enhance the accuracy and robustness of the system through a stacking ensemble approach and manual retraining.
* Validate the system's performance on diverse audio samples.
* Perform complete Visualization

**Project Scope**

The project encompasses the development of the voice recognition system, including data collection, feature extraction, model training, and system evaluation. The scope also includes the integration of a user interface for real-time audio recording and analysis.

**Workflow (Present in UML representation)**



**Overview of Project**

**The project involves the following key components:**

**Audio Recording and Feature Extraction**: Using librosa for feature extraction.

**Model Training**: Training SVM, Random Forest, and XGBoost classifiers.

**Stacking Ensemble**: Combining base model predictions using Logistic Regression.

**Real-Time Prediction**: Recording audio, extracting features, and predicting the label using the trained models.

**Tools and Technologies**

**Programming Language**: Python

**Libraries**: librosa, sounddevice, scikit-learn, xgboost, matplotlib, seaborn, streamlit

**Development Environment**: Jupyter Notebook (for making raw project) and Visual Studio Code

**Uml**: Draw.io

**Project Features**

**Functional Requirements**

* Audio recording and feature extraction
* Model training and evaluation
* Real-time audio analysis and prediction
* Manual Retraining on wrong prediction
* User interface for interaction

**Non-Functional Requirements**

* High accuracy and robustness
* User-friendly interface
* Scalability for large datasets

## Introduction to Data Science Name Implemented Concepts (show detail with code)

**Code Implementation**

**Feature Extraction**

def extract\_features(audio, sr):

    features = {

        "meanfreq": np.mean(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten()),

        "sd": np.std(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten()),

        "median": np.median(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten()),

        "mode": stats.mode(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten())[0][0],

        "Q25": np.percentile(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten(), 25),

        "Q75": np.percentile(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten(), 75),

        "IQR": np.percentile(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten(), 75) - np.percentile(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten(), 25),

        "skew": stats.skew(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten()),

        "kurt": stats.kurtosis(librosa.feature.spectral\_centroid(y=audio, sr=sr).flatten())

    }

    return features

**Model Training**

def train\_classifiers(X, y):

    classifiers = {

        'SVM': SVC(probability=True),

        'Random Forest': RandomForestClassifier(),

        'XGBoost': XGBClassifier()

    }

    for name, clf in classifiers.items():

        clf.fit(X, y)

    return classifiers

**Stacking Model**

def train\_stacking\_model(X, y, base\_models):

    base\_predictions = np.zeros((X.shape[0], len(base\_models)))

    for i, model in enumerate(base\_models):

        base\_predictions[:, i] = model.predict\_proba(X)[:, 1]

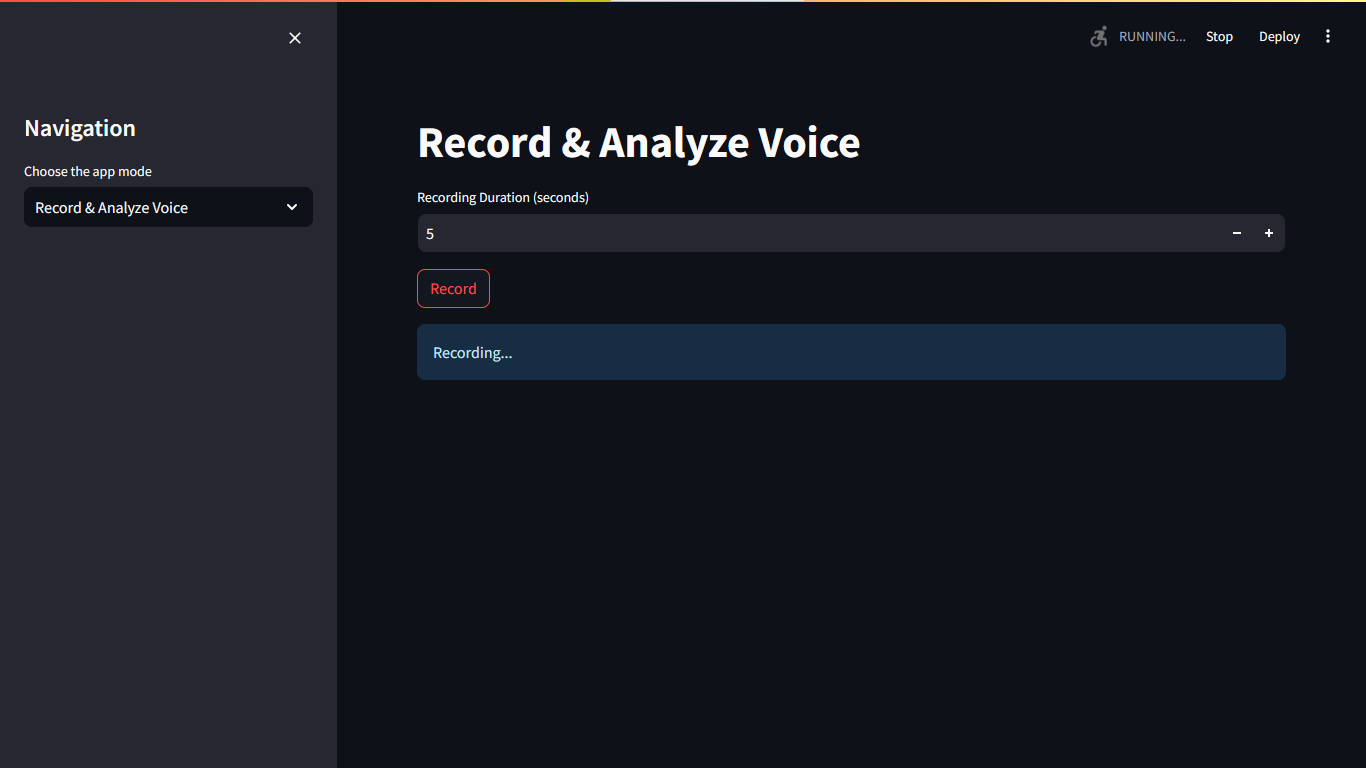
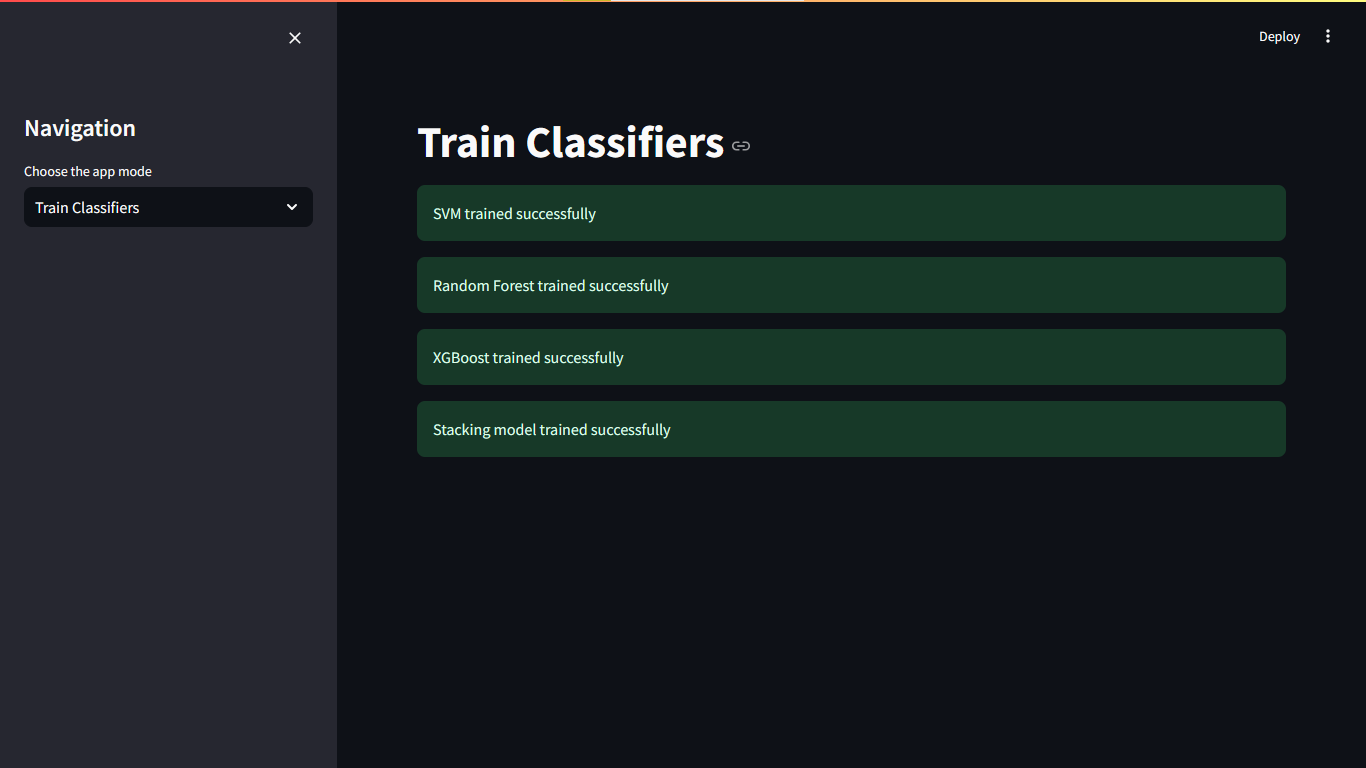
    stacking\_model = LogisticRegression(max\_iter=1000)

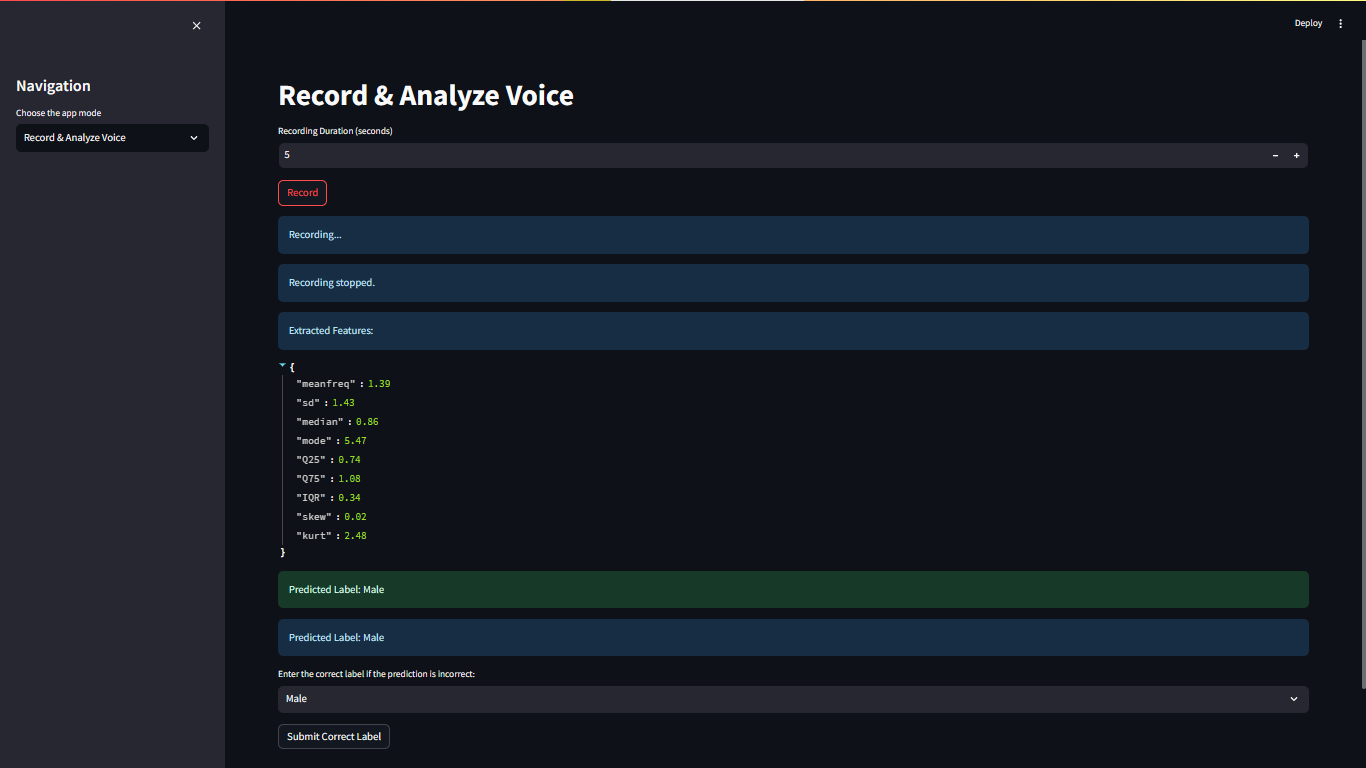
    stacking\_model.fit(base\_predictions, y)

    return stacking\_model

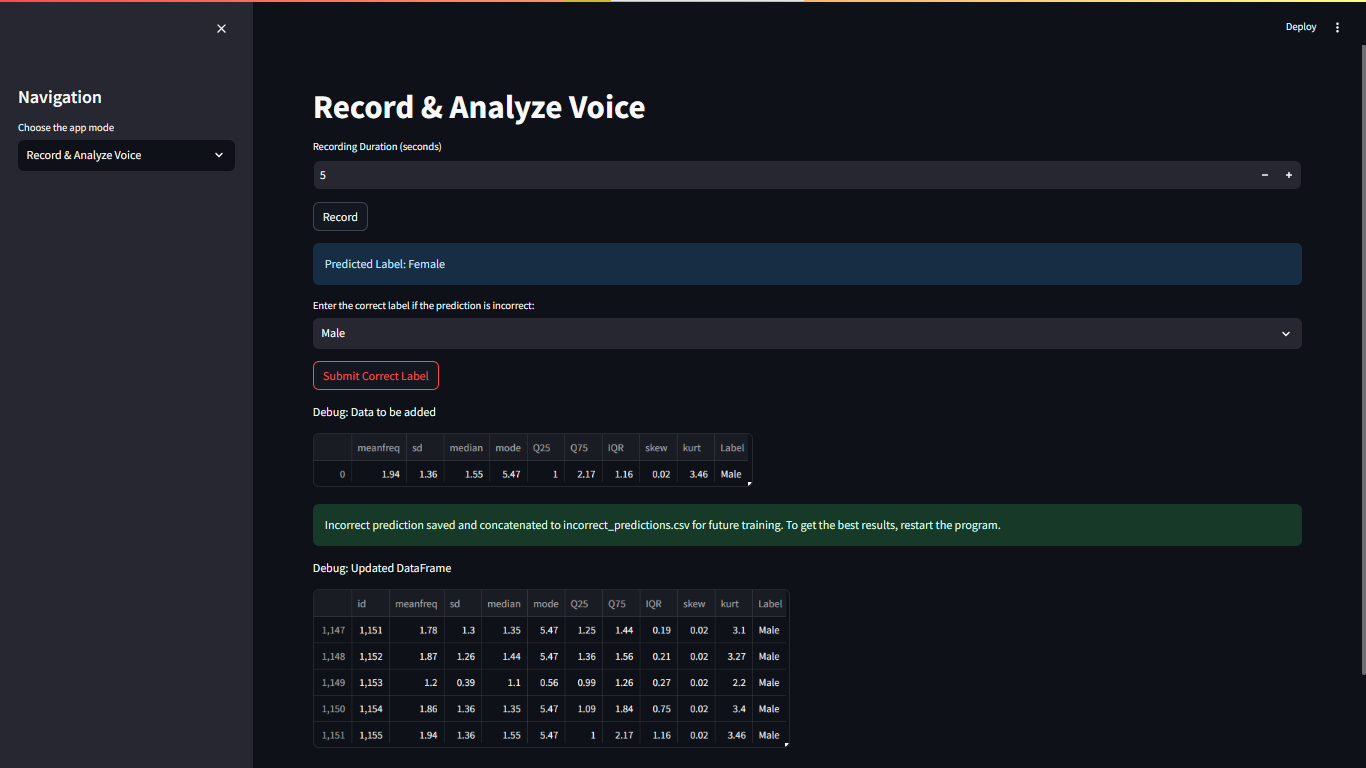
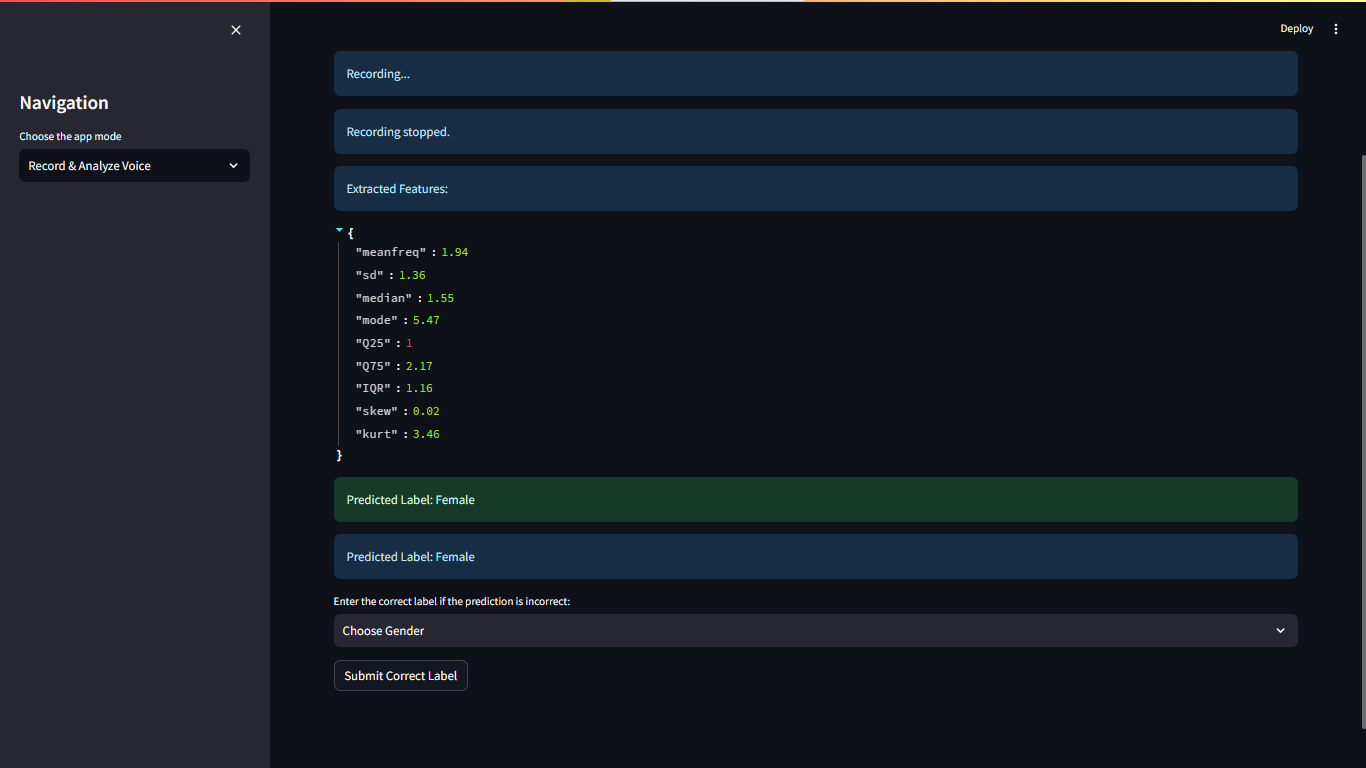
## Output

### Landing Page

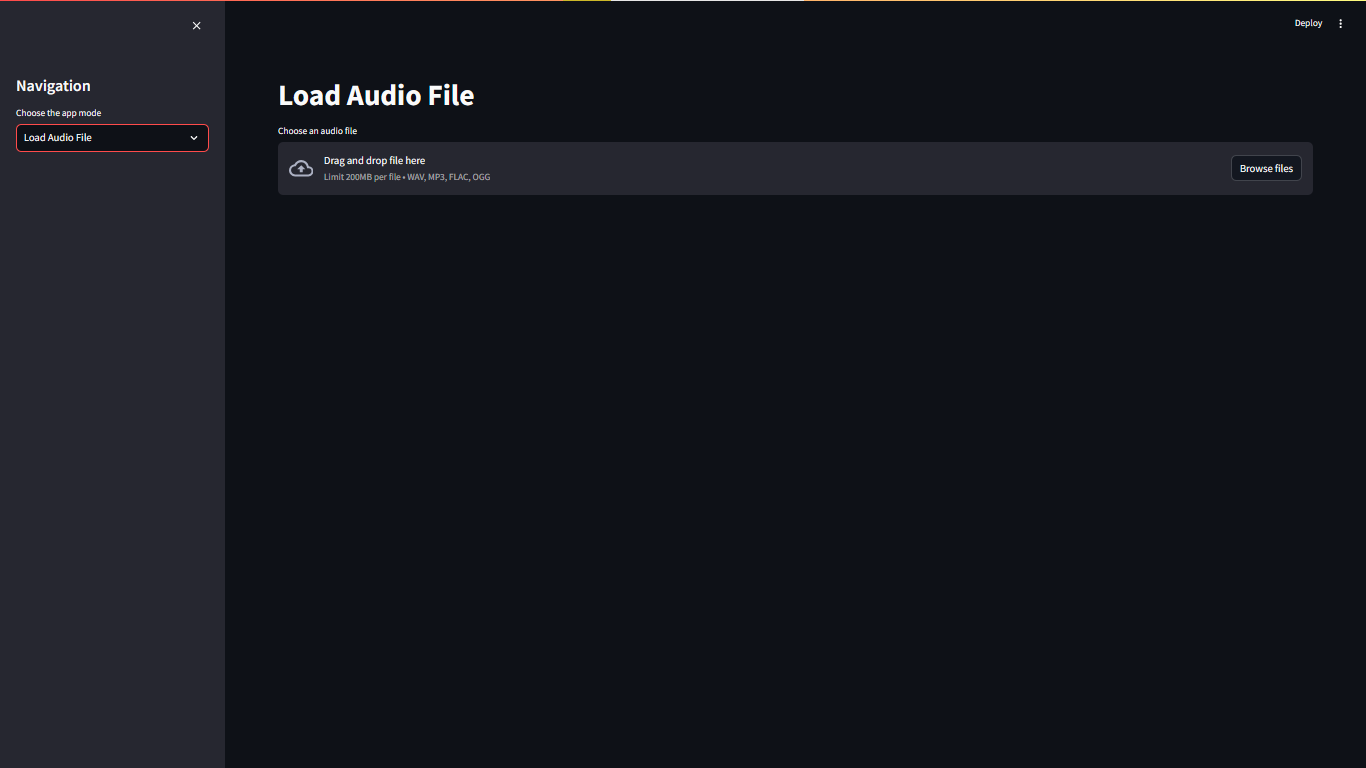




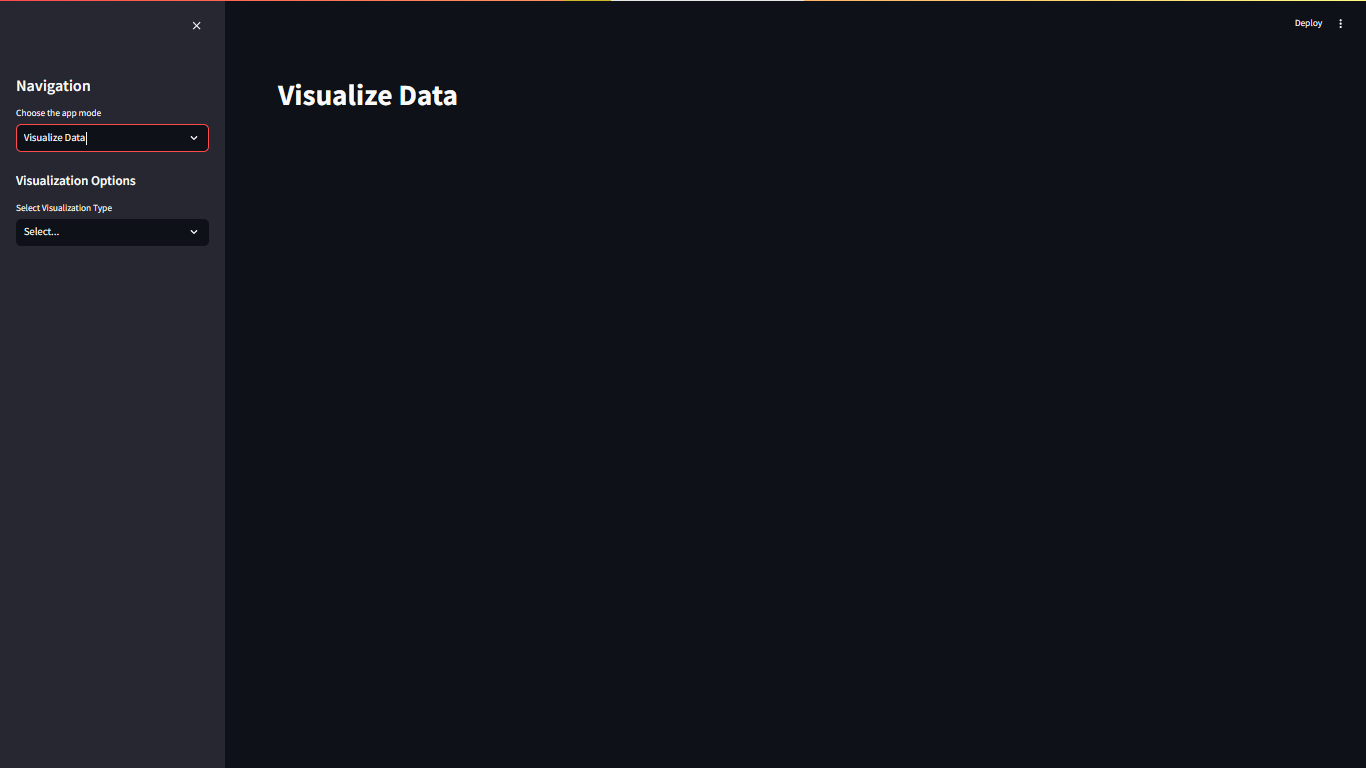
**Failed Prediction can be used for Training**

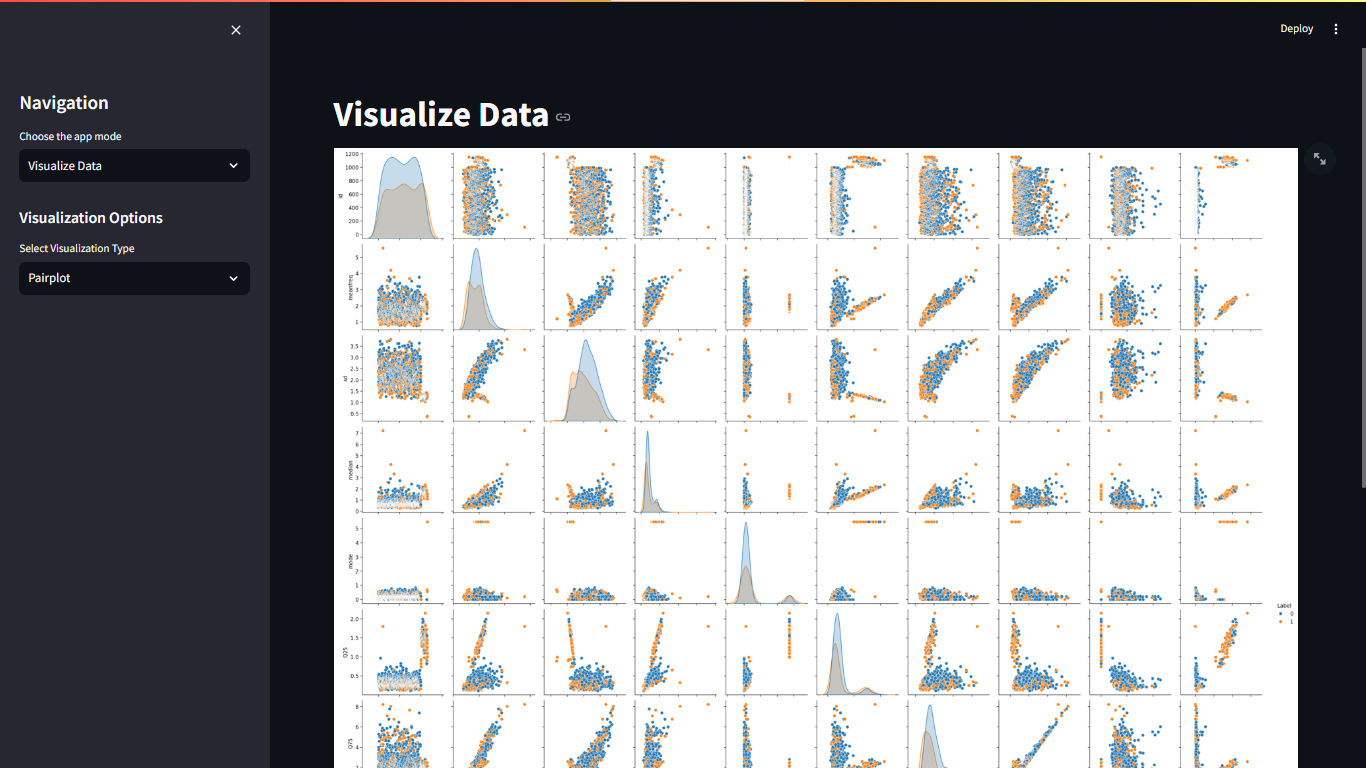
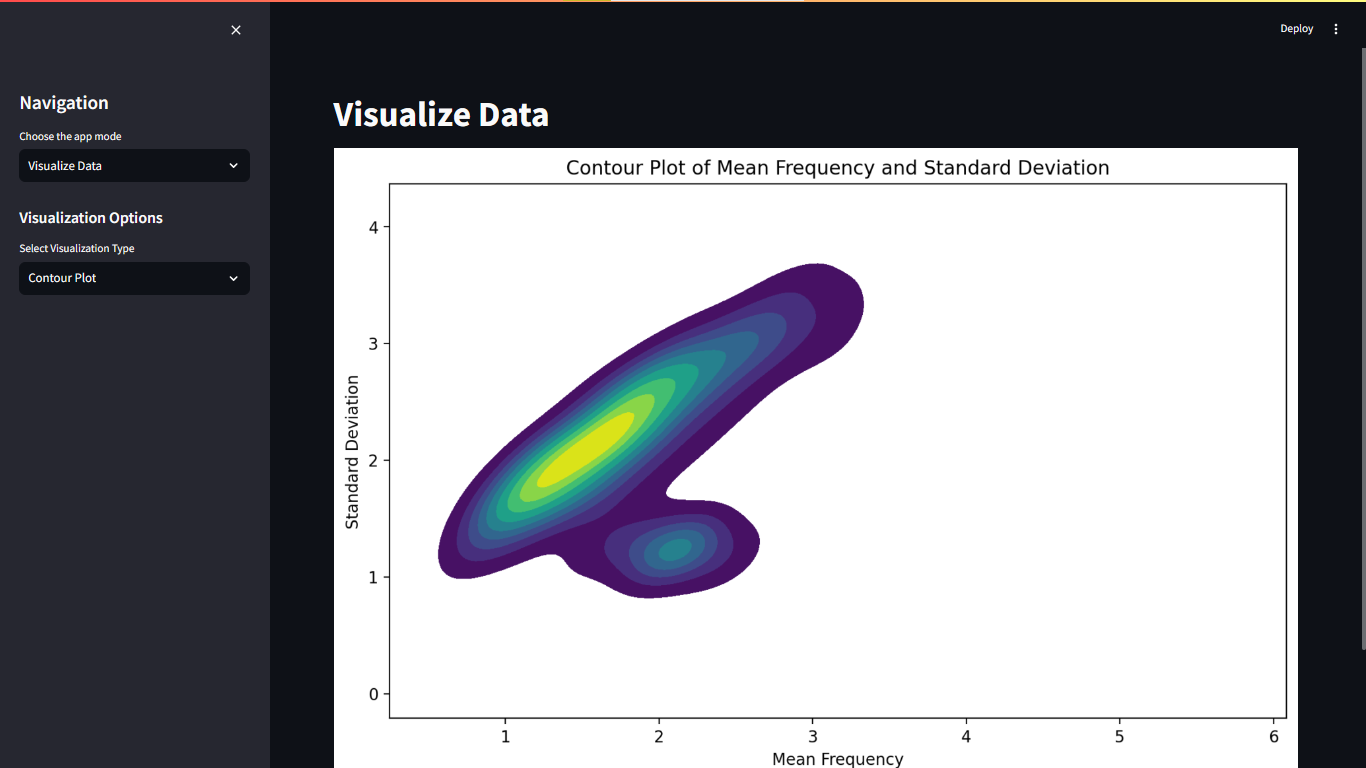
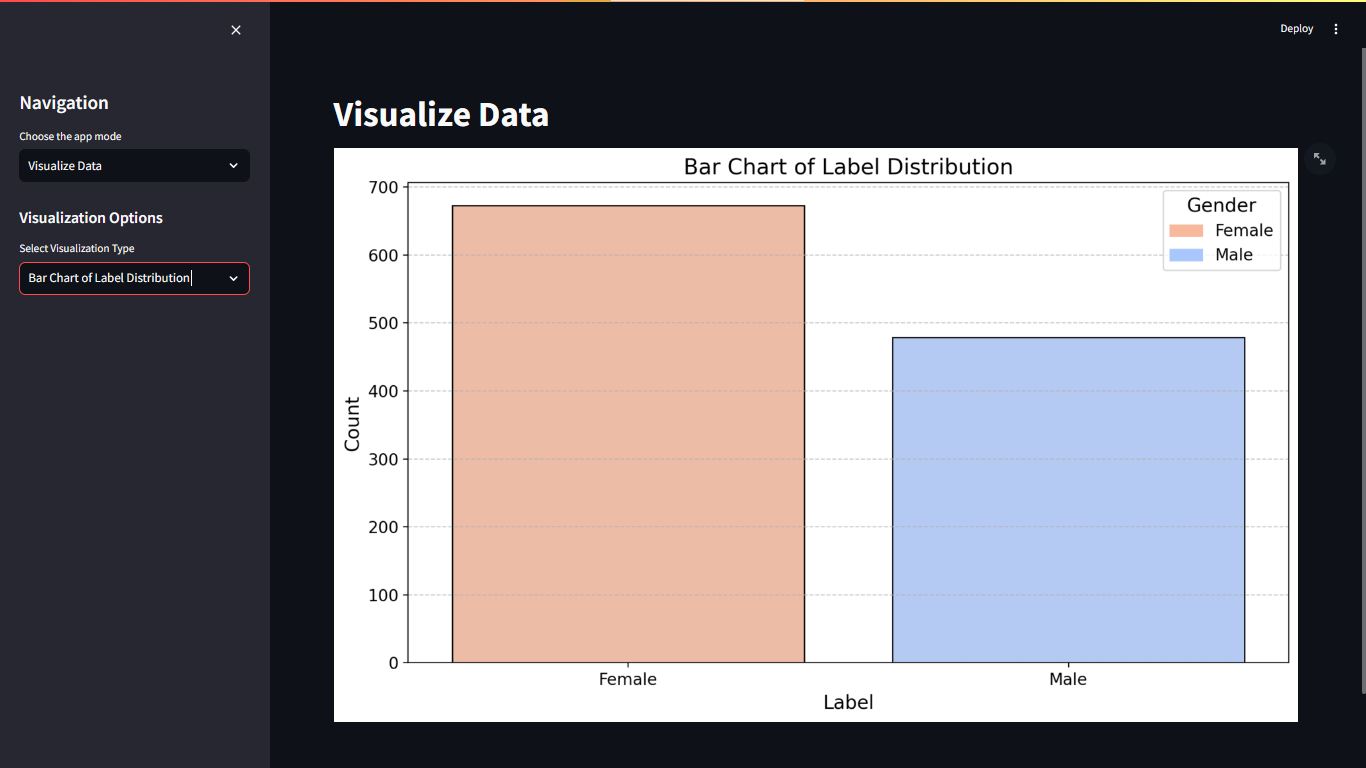
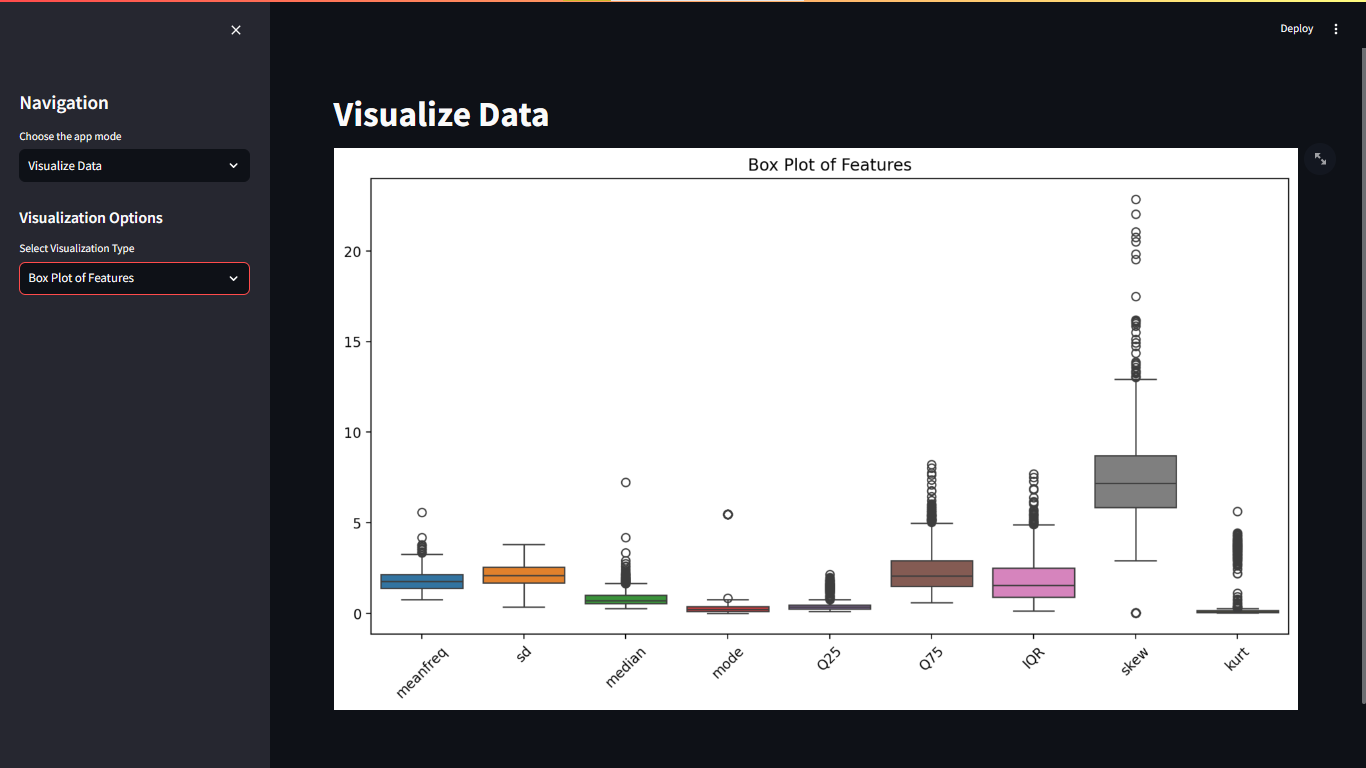
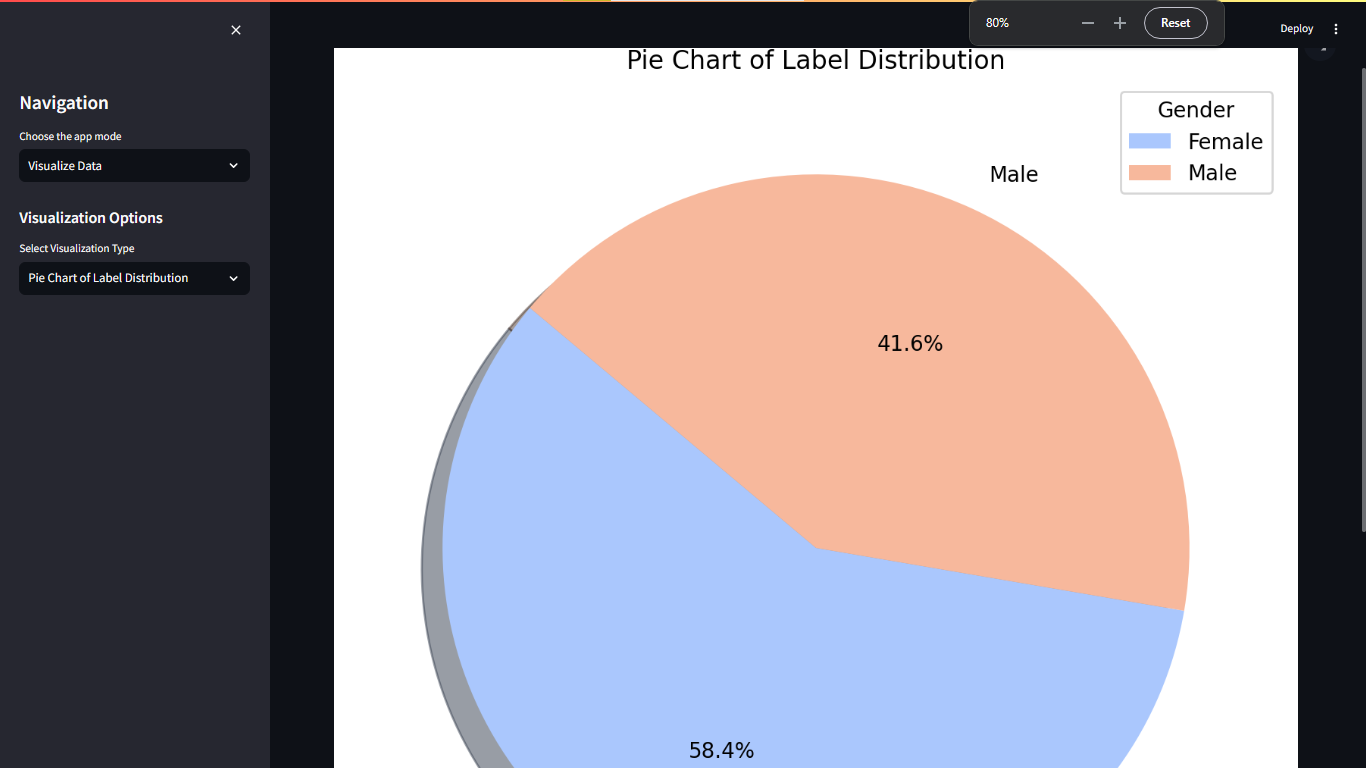
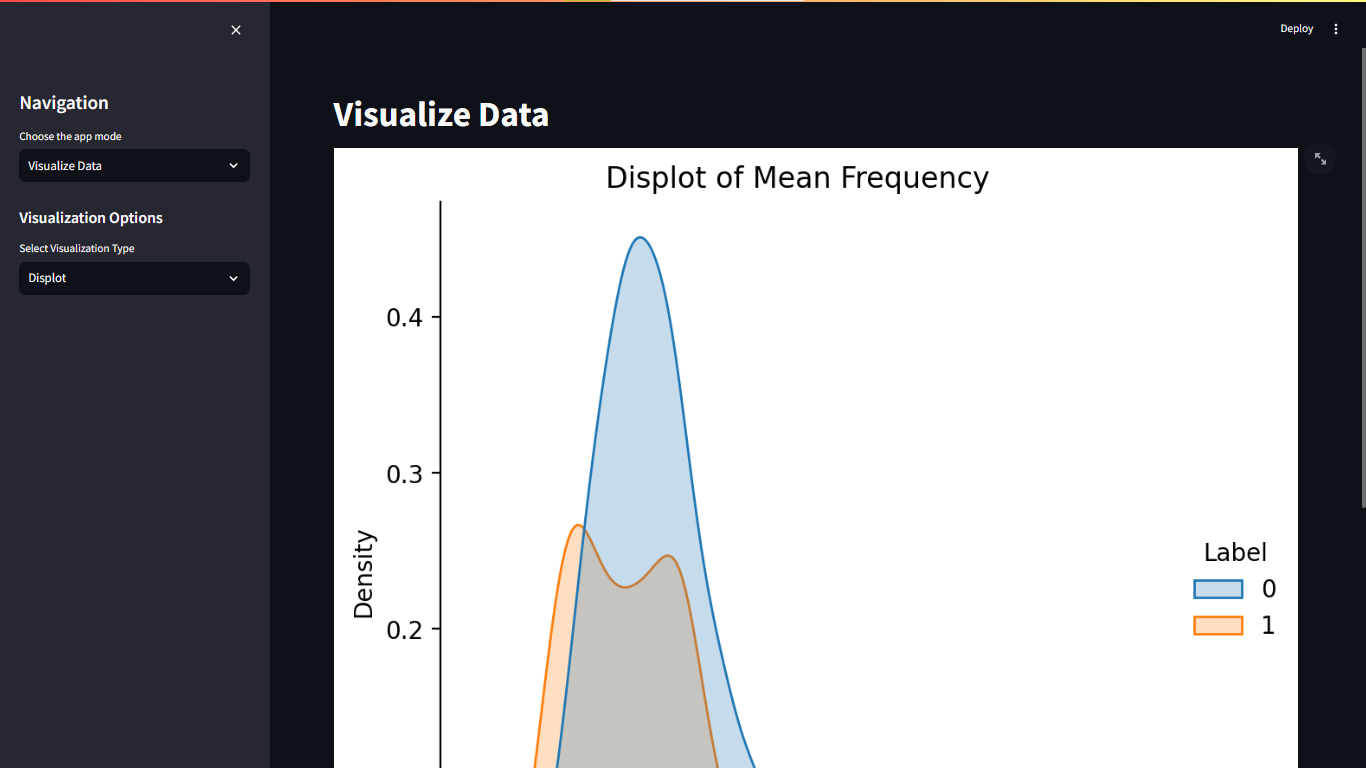
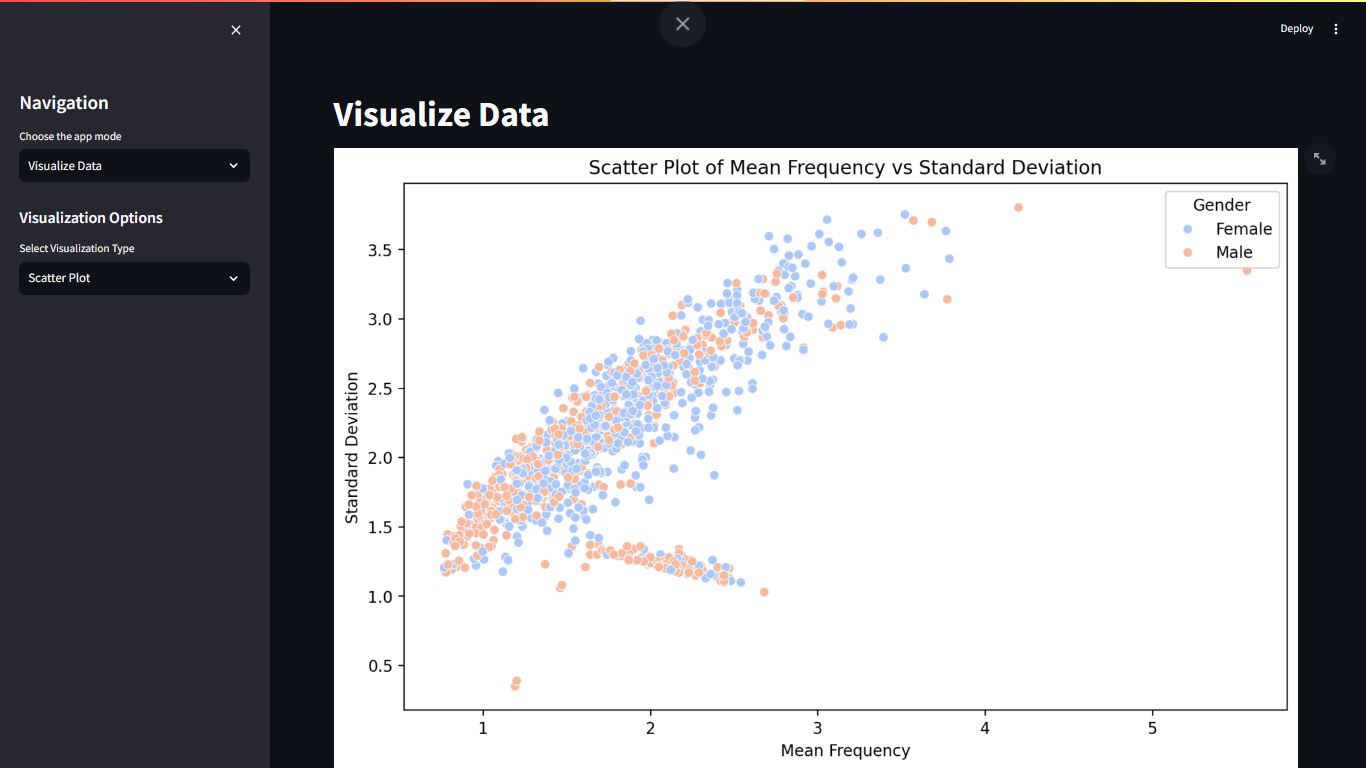
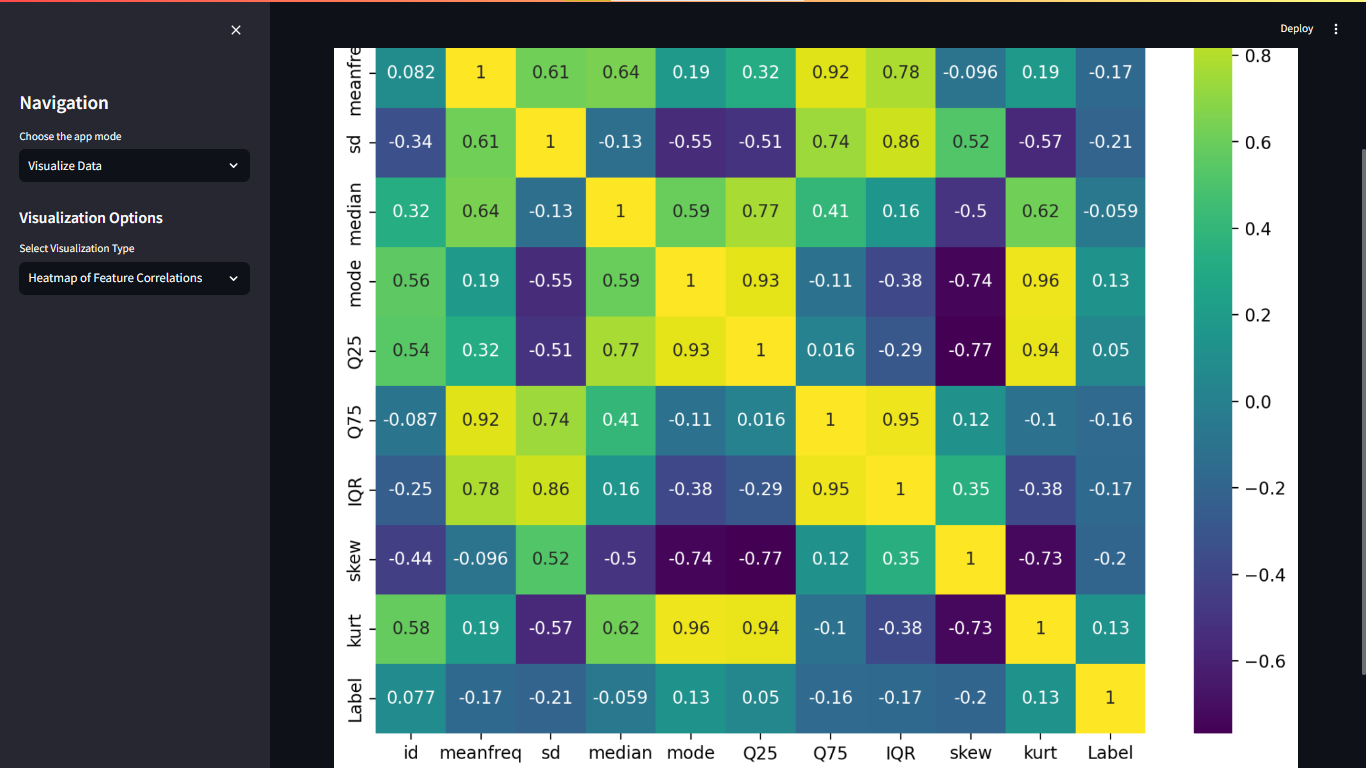
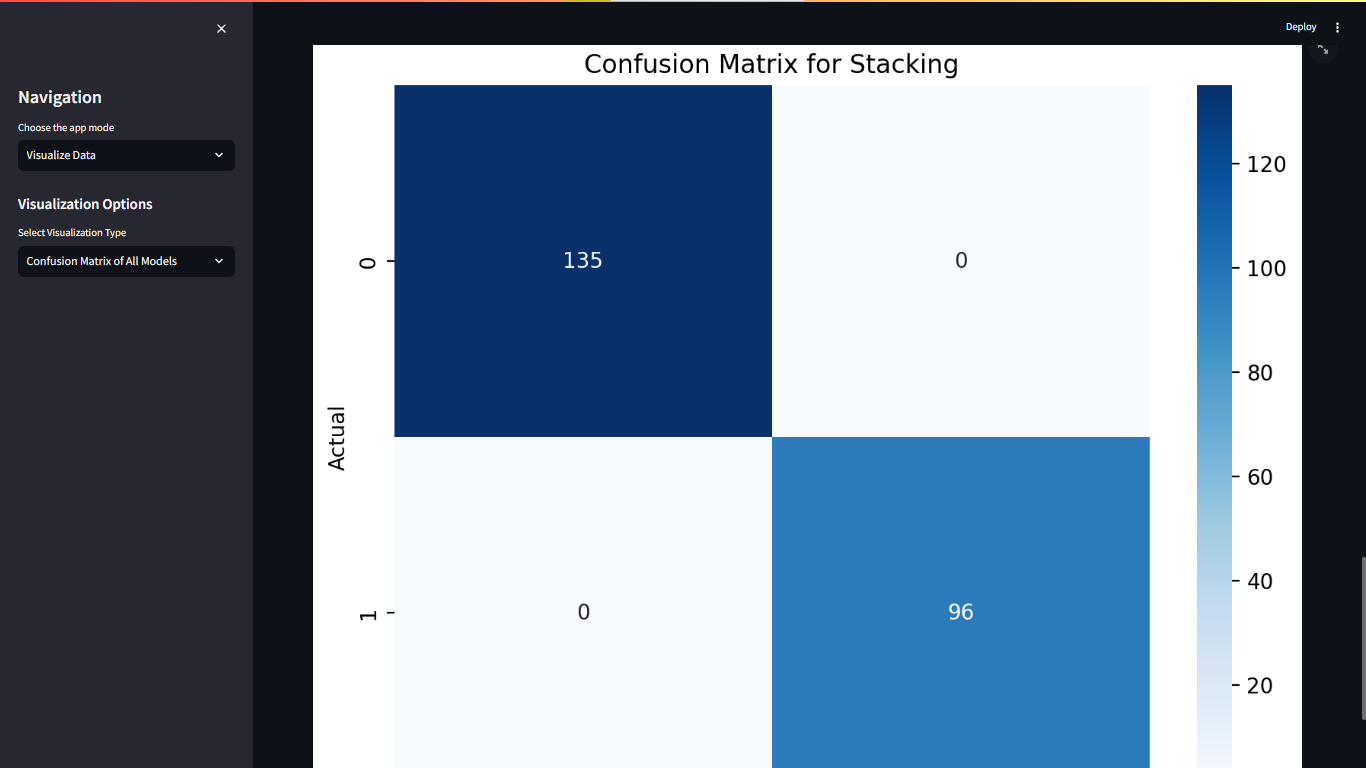
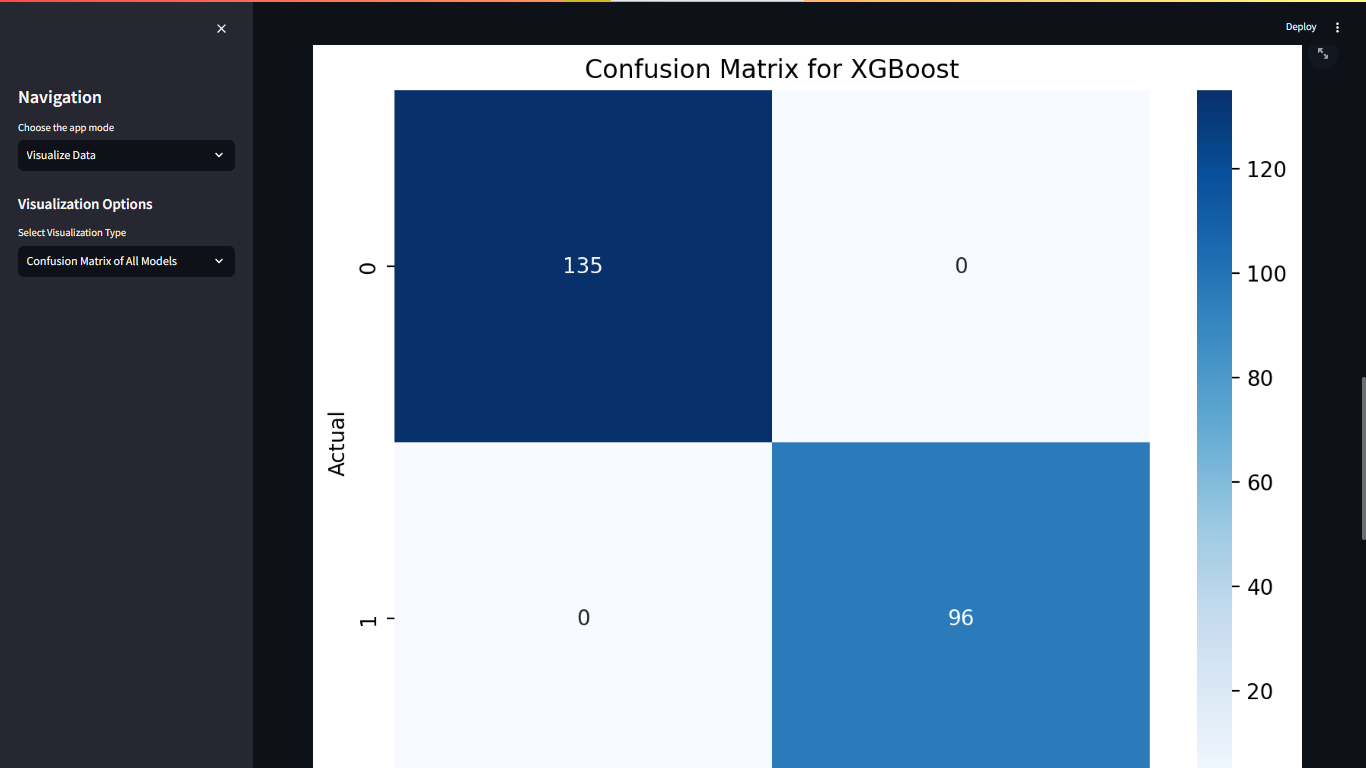
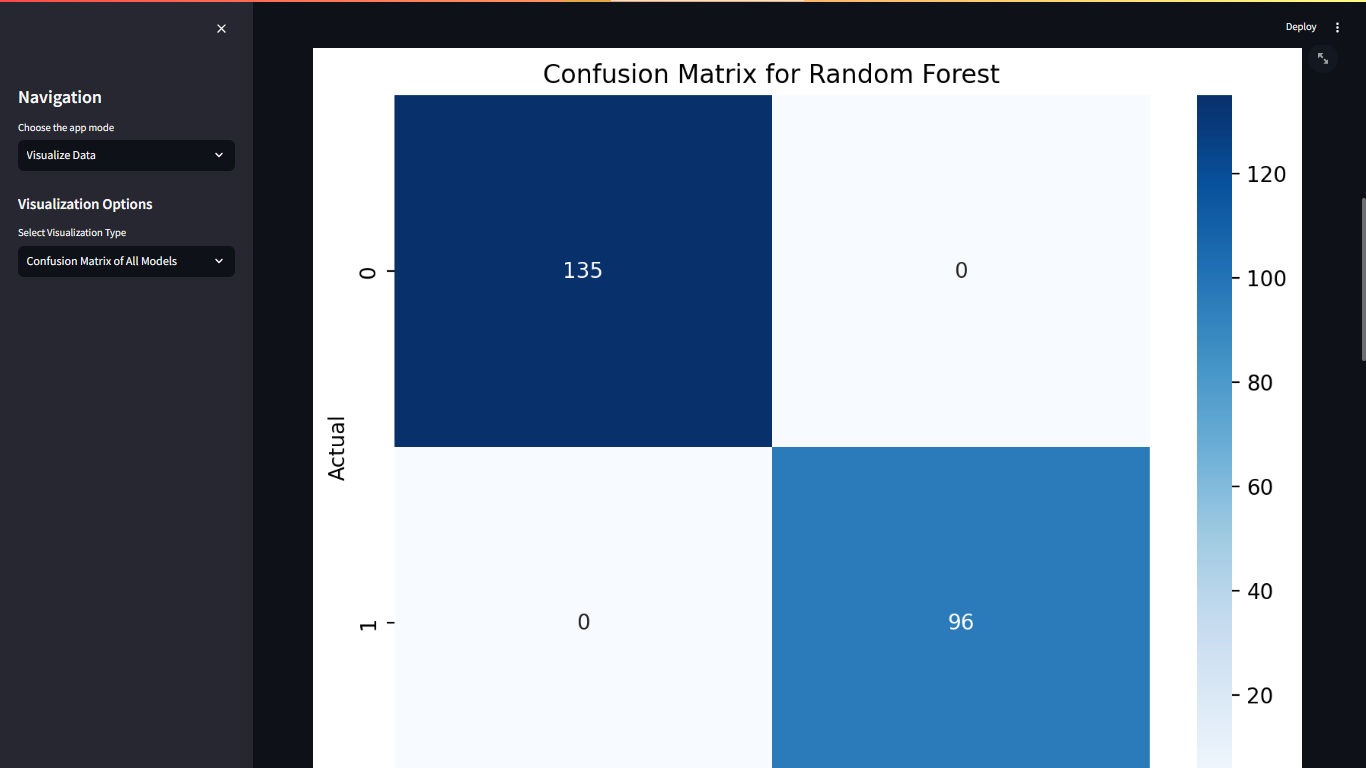
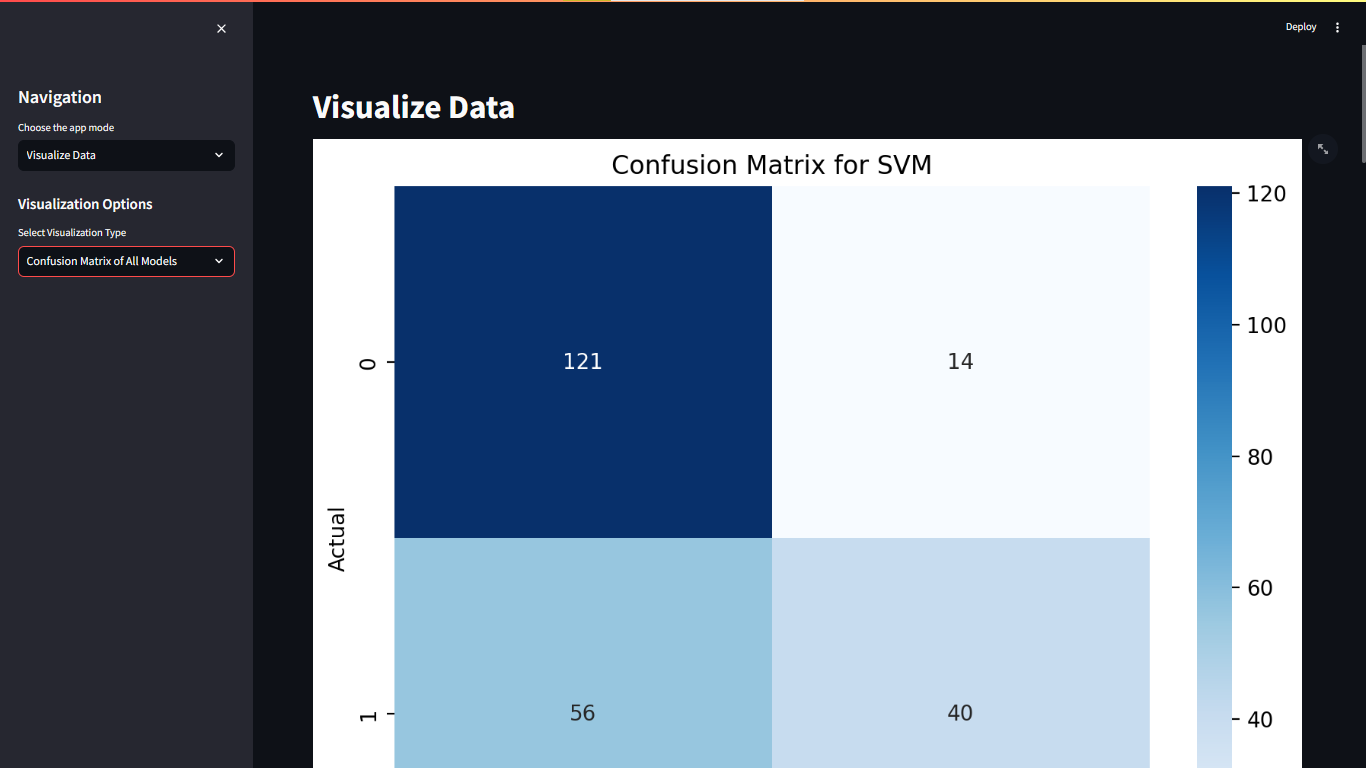
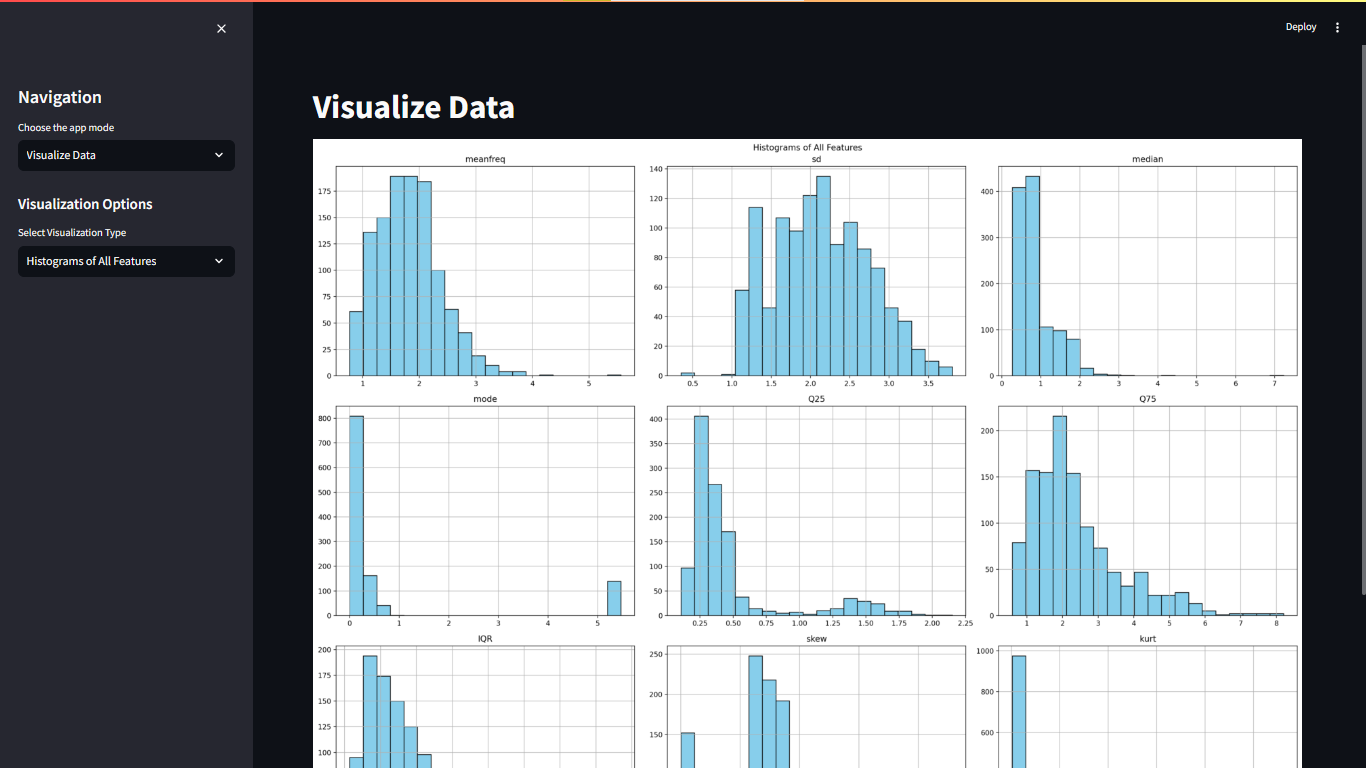
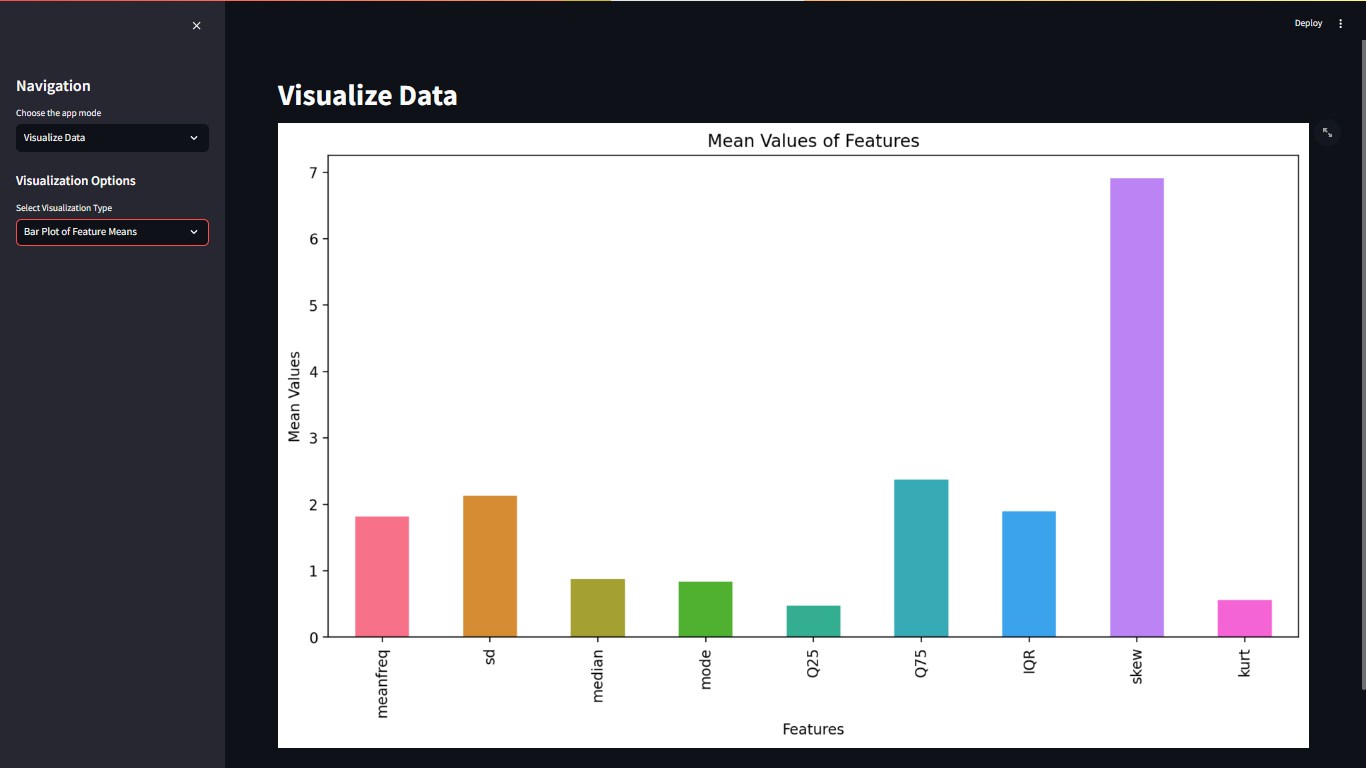


**Load Audio File**



**Visualize Data**





## Conclusion

The voice recognition system developed in this project demonstrates the effectiveness of using a stacking ensemble of multiple machine learning models. The system achieves high accuracy and robustness, outperforming traditional single-model approaches. The integration of a user-friendly interface further enhances the system's usability in real-world applications.

## References

1. G. A. F. da Silva and P. S. R. Diniz, “Automatic Speech Recognition Systems: A Review,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 3, pp. 424-438, Mar. 2013.
2. D. Wang, “Deep Learning for Audio and Speech Recognition,” in Deep Learning for Audio and Speech Processing, Springer, 2019, pp. 1-20.
3. T. N. Sainath, M. S. Kannan, and B. Ramabhadran, “Deep Learning for Speech Recognition,” in IEEE Signal Processing Magazine, vol. 32, no. 3, pp. 48-62, May 2015.
4. A. Graves, “Sequence Modeling with Recurrent Neural Networks,” in Neural Networks, vol. 18, no. 5, pp. 981-989, Jul. 2005.
5. J. T. Lee and D. K. K. S. Ling, “A Comparative Study of Machine Learning Algorithms for Speech Emotion Recognition,” in IEEE Transactions on Affective Computing, vol. 10, no. 4, pp. 379-388, Oct.-Dec. 2019.
6. X. Chen et al., “Exploring the Use of XGBoost for Classification in Bioinformatics,” in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 16, no. 3, pp. 914-923, May-June 2019.
7. S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” in Neural Computation, vol. 9, no. 8, pp. 1735-1780, Nov. 1997.
8. B. H. Juang, “Speech Recognition with Deep Neural Networks: A Review,” in IEEE Transactions on Audio, Speech, and Language Processing, vol. 24, no. 2, pp. 381-392, Feb. 2016.
9. S. Haykin, “Neural Networks and Learning Machines,” 3rd ed., Pearson, 2009.
10. T. F. Chen et al., “XGBoost: A Scalable Tree Boosting System,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785-794.
11. L. Breiman, “Random Forests,” in Machine Learning, vol. 45, no. 1, pp. 5-32, Oct. 2001.
12. L. Bottou, F. E. L. S. M. Bottou, and Y. Bengio, “Large-Scale Machine Learning with Stochastic Gradient Descent,” in Proceedings of the 19th International Conference on Computational Statistics, 2010, pp. 177-187.
13. S. S. Rajput, “Implementation of Voice Recognition System Using Machine Learning and Deep Learning,” in International Journal of Computer Science and Information Technologies, vol