Software Project
AI1110: Probability and Random Variables
Indian Institute of Techonology Hyderabad

Mayank Parasramka*
AI22BTECH11018

CONTENTS

1 INTRODUCTION

The Python code provided implements a music player that allows users to play audio files from a specified directory. The program utilizes the Pygame library for audio playback and provides playback control functionalities such as pause, resume, skip to the next song, and quit.

2 CODE OVERVIEW

(i) The code starts by importing the required modules: os, numpy, and pygame.
(ii) The audio directory variable holds the path to the directory containing the audio files.
(iii) Pygame.mixer is initialized to set up the audio mixer.
(iv) The main program loop is executed indefinitely using a while True loop.
(v) Within the loop, the code retrieves a list of audio files from the specified directory, shuffles them, and iterates over the shuffled list.
(vi) For each audio file, the code loads it into the Pygame mixer and plays it.
(vii) During audio playback, the program waits for user commands.
(viii) The available commands are 'p' for pause, 'r' for resume, 'n' for next song, and 'q' for quitting the program.
(ix) The code also includes handling for the pause, resume, and quit commands.
(x) After playing the entire playlist, the audio files are reshuffled, and the loop continues

3 FUNCTIONALITY AND FEATURES

(i) The music player supports playing audio files from a specified directory.
(ii) The playlist is shuffled before each iteration to provide variety.
(iii) Users can control playback using the provided commands: pause, resume, skip to the next song, and quit the program.
(iv) The program displays the currently playing song name.

*The student is with the Department of Artificial Intelligence, Indian Institute of Technology, Hyderabad, 502285, India.
e-mail: ai22btech11018@iith.ac.in.

4 Usage

(i) Update the audio directory variable with the path to the directory containing your audio files.

(ii) Make sure to have the necessary audio files in the specified directory.

(iii) Run the Python script.

(iv) The program will start playing the audio files in a shuffled order.

(v) Use the command prompt to control the playback:

    (a) 'p': Pause the current song.

    (b) 'r': Resume playback.

    (c) 'n': Skip to the next song.

    (d) 'q': Quit the program.

(vi) The program will continue playing songs until manually terminated.

5 Conclusion

The provided Python code offers a basic music player with playback control features. It enables users to play audio files from a directory and provides commands for pause, resume, skipping to the next song, and quitting the program. The code can be further extended and customized to add more functionalities and improve the user experience.