GP Assignment – Markov Analysis and Algorithmic Composition

Name: Jordan Bailey

Abstract: This is the first GP assignment, focusing on creating state transition *count*, *probability*, and *distribution* matrices for the tunes, “Ode to Joy”, by Beethoven, and Turk’s “March”. After manually running the algorithm presented earlier in the course, the new 30 note musical tunes created will sound eerily similar to the respective original song.

Beethoven Task

*State Transition **Count** Matrix*

From/To	E	F	G	D	C	EQ	DI	DH	DQ	CI	CH
E	**2**	**2**	**0**	**2**	**0**	**1**	**0**	**0**	**1**	**0**	**0**
F	**2**	**0**	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
G	**0**	**2**	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
D	**2**	**0**	**0**	**0**	**2**	**0**	**0**	**0**	**0**	**0**	**0**
C	**0**	**0**	**0**	**2**	**2**	**0**	**0**	**0**	**0**	**0**	**0**
EQ	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**
DI	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**
DH	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
DQ	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
CI	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**
CH	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

State Transition ***Probability*** *Matrix*

From/To	E	F	G	D	C	EQ	DI	DH	DQ	CI	CH
E	.25	.25	0	.25	0	.125	0	0	.125	0	0
F	*.5*	0	*.5*	0	0	0	0	0	0	0	0
G	0	*.5*	*.5*	0	0	0	0	0	0	0	0
D	*.5*	0	0	0	*.5*	0	0	0	0	0	0
C	0	0	0	*.5*	*.5*	0	0	0	0	0	0
EQ	0	0	0	0	0	0	1.00	0	0	0	0
DI	0	0	0	0	0	0	0	1.00	0	0	0
DH	1.00	0	0	0	0	0	0	0	0	0	0
DQ	0	0	0	0	0	0	0	0	0	1.00	0
CI	0	0	0	0	0	0	0	0	0	0	1.00
CH	1.00	0	0	0	0	0	0	0	0	0	0

State Transition ***Probability Distribution*** *Matrix*

From/To	**E**	**F**	**G**	**D**	**C**	**EQ**	**DI**	**DH**	**DQ**	**CI**	**CH**
E	**.25**	**.50**	***.50***	**.75**	***.75***	**.875**	***.875***	***.875***	**1.00**	**1.00**	**1.00**
F	***.50***	**.50**	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***
G	**0**	***.50***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***
D	***.50***	**.50**	**.50**	**.50**	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***
C	**0**	**0**	**0**	***.50***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***
EQ	**0**	**0**	**0**	**0**	**0**	**0**	**1.00**	***1.00***	***1.00***	***1.00***	***1.00***
DI	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1.00**	***1.00***	***1.00***	***1.00***
DH	**1.00**	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***
DQ	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1.00**	***1.00***
CI	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1.00**
CH	**1.00**	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***	***1.00***

New Sequence of Notes Produced:

E D C C D E F E EQ DI DH E E D C D C D C C D E E E E E F G G F

Turk Task

*State Transition **Count** Matrix*

From/To	C	G	E	D	EH	CH	FH	DH
C	2	1	0	1	0	0	0	0
G	0	0	1	0	1	0	1	0
E	1	0	3	1	0	0	0	0
D	0	2	0	5	0	1	0	0
EH	0	0	0	0	0	1	0	0
CH	1	0	0	1	0	1	0	0
FH	0	0	0	0	0	0	0	1
DH	0	0	1	0	0	0	0	0

*State Transition **Probability** Matrix*

From/To	**C**	**G**	**E**	**D**	**EH**	**CH**	**FH**	**DH**
C	**.50**	**.25**	**0**	**.25**	**0**	**0**	**0**	**0**
G	**0**	**0**	**.33**	**0**	**.33**	**0**	**.33**	**0**
E	**.20**	**0**	**.60**	**.20**	**0**	**0**	**0**	**0**
D	**0**	**.25**	**0**	**.625**	**0**	**.125**	**0**	**0**
EH	**0**	**0**	**0**	**0**	**0**	**1.00**	**0**	**0**
CH	**.33**	**0**	**0**	**.33**	**0**	**.33**	**0**	**0**
FH	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1.00**
DH	**0**	**0**	**1.00**	**0**	**0**	**0**	**0**	**0**

State Transition ***Probability Distribution*** *Matrix*

From/To	**C**	**G**	**E**	**D**	**EH**	**CH**	**FH**	**DH**
C	**.50**	**.75**	**.75**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
G	**0**	**0**	**.33**	**.33**	**.66**	**.66**	**1.00**	**1.00**
E	**.20**	**.20**	**.80**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
D	**0**	**.25**	**.25**	**.875**	**.875**	**1.00**	**1.00**	**1.00**
EH	**0**	**0**	**0**	**0**	**0**	**1.00**	**1.00**	**1.00**
CH	**.33**	**.33**	**.33**	**.66**	**.66**	**1.00**	**1.00**	**1.00**
FH	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1.00**
DH	**0**	**0**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

New Sequence of Notes Produced:

C C D D CH CH C C G EH CH D D D G E E E E E E E D G FH DH

Short Essay

Within this assignment, “Ode to Joy”, by Beethoven, and Turk’s “March”, were the two songs being experimented on. The two songs have clear salient differences, which can be used to capture a particular “spirit” of the song. “Ode to Joy” tends to gradually increase in pitch, while Turk’s “March”, tends to have repetition in its notes before raising or lowering the pitch. Another difference between the two songs that sets them apart from one another would be the duration of the notes themselves. While “Ode to Joy”, tends to have quicker notes (half notes), Turk’s “March”, uses whole notes more often. These salient differences between the songs are able to be captured through the use of the *State Transition Probability Matrix*. For “Ode to Joy”, its STPM is able to capture the higher probability of “step-like” movements for the notes and gives a higher probability of landing on half notes, rather than whole notes. The fragment created from Ode to Joy’s STPM captures the gradual movements of the notes well, especially in the beginning of the song, and uses half notes more frequently than whole notes. The new song created for Turk’s March also keeps the spirit of its original song intact through the STPM. The “fake” version of March is accurately able to mimic both the prolonged notes, and the repetition of the notes before changing pitch. These fake versions of the songs are very simple examples of the incredible surface level mimicking that *Markov Processes* are able to produce. Markov Processes are memoryless, stochastic processes that are able to make predictions for the future of a particular process based on its present state. To generate these melodies, the open-source programming library titled “Jfugue”, was used as reference. Jfugue is a musical knowledge representation that uses inherent conventions such as pitch classes, pitch identifiers, and letters following the note identifiers to indicate the duration of the notes (W, H, Q, I, S). To render the sequence of notes for the faked melodies, the notes and notation within Jfugue were used as reference, and an online sequencer was used (onlinesequencer.net), to replicate the tune, and to save as a .mp3 file. While Jfugue was done with programming, onlinesequencer.net gives a user interface, and affords the user to drag/drop notes, and lengthen/shorten them at will. The process as a whole can be considered as an improvisational process in the sense that the notes created are, “randomized”. While the songs recreated

within this activity may not be melodically satisfying, the amount of characteristics kept from the original songs are impressive considering the simplicity of the matrices, and it effectively portrays the importance of Markov processes for replication and generation purposes.