## August

Contact Brian Oberlin, mandolinist in the Grand Rapids area, for permission to use his music in this project and listened to a selection of his pieces on YouTube.

### Hours: 1

## September

Researched the Fast Fourier Transform for digital signal processing. Downloaded and converted a C Major Chord from YouTube to mp3 and wav formats. Used a python script that uses FFT to display the frequencies for the audio file. Debugged inputting wav files and larger mp3 files.

Found mandolin songs on YouTube and converted videos into mp3 files. Researched several machine learning music videos on YouTube and read the documentation on LibROSA, a python library for audio analysis.

### Hours: 7

## October

Installed PyCharm, a development environment for Python, and installed Python 2.7 and 3.7. Installed the libROSA library. There were problems using the mandolin mp3 files in the FFT, so more research was conducted on converting mp3 files into datasets and different libROSA functions.

Convert music files from mp3 to wav. Created a data preprocessing function.

Further research with audio classification using FastAI, how to convert mp3/wav to a dataframe, and how to save the dataframe to a file. Used the Pip tool to install several libraries. Learned about the pandas dataframe and numpy array data types and their methods. Ran the FTT with the mp3 songs but could not save the entire dataframes into a csv.

Package management with Pip.

Researching machine learning models (LSTM, RNN) and found several code samples on GitHub for music generation. Converted audio files to MIDI because the file structure for MIDIs seemed like they were easier to work with. Researched Keras for music generation and music21 for wav files.

Install Tensorflow with Pip. However, there were issues where tensorflow would not detect the computer's graphics card. Installed NVIDIA's CUDA Toolkit 10.1 and cuDNN to try to remedy this issue but the cards still were not found.

Hours: 37

## November

Look up GPU support for Tensorflow 1.15 and 2.0. Upgraded Tensorflow to 2.0 but ran into issues with an HDF5 version mismatch which is tied to the H5PY library. Looked up how to use GPUs with Keras.

Ran the program but the result was an audio that was similar to the input except fuzzier.

Contacted Dr. Wolffe for access to the research computer and created an anaconda virtual environment on my account. Used conda to install necessary libraries. The connection to the research computer regularly dropped and I was not able to reconnect due to a timeout error.

Since the research computer was unavailable, I continued the work on my own computer. I experimented with the program using different numbers of iterations and higher look back values to try to get better results. When the lookback value was at least 2000 and sample size was at least 20000, there would be a memory error that would end the computations on my personal computer.

Ran experiments with various number of neural network layers and activation functions. Predicted results of a MSLE loss function.

Create a journal and final report.

Hours: 17

Hours Training on research computer: 12

Hours Training on personal computer: 28

## Total Hours: 102

## Lessons Learned

There were learning opportunities throughout the entire progression of this project. I learned the fundamentals of mp3 and wav data and how it relates to audio, which was what the Fast Fourier Transform and the model would depend on. Prior to this project, I only used Python for small projects. However, machine learning in Python has an abundance of libraries that are commonly used, and I learned how to create virtual environments as well as package management. These two skills made it possible to work on multiple projects that may not use the same versions of libraries. I also learned about the different forms of activation and loss functions through experimentation. Since there was a lot of experimentation, each run should have been committed to the GitHub repository with the resulting audio file to better document the trials that were already completed. I first learned of the NVIDIA CUDA platform and its use in allowing GPUs to be used for normal processing through this project. Lastly, I noticed that LSTMs require a lot of calculations and there may be more efficient ways to configure this current model or use a different model.