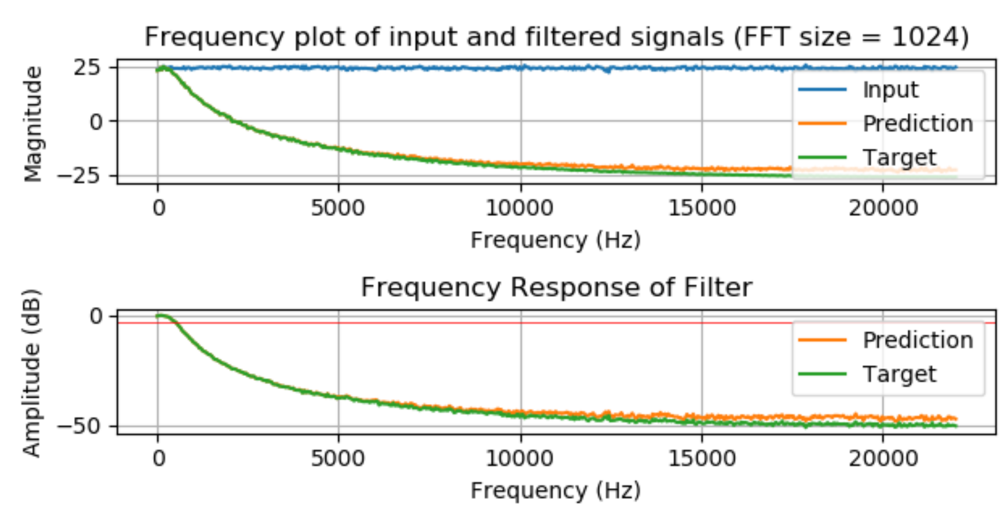
Modelling Filters Using Neural Networks

Kaushal Sali[1] and Alexander Lerch[1]
[1]Georgia Institute of Technology, Atlanta, GA, USA

Georgia Tech Center for Music Technology

INTRODUCTION

- **Replication of Audio Effects:** An easy way to reproduce an effect using only raw audio.
- Audio Equalization is an important effect used extensively in music production which manipulates frequency content by using filters.
- We try modelling filters using a multilayer perceptron. Try to extend previous architecture which modelled waveshapers.



RELATED WORK

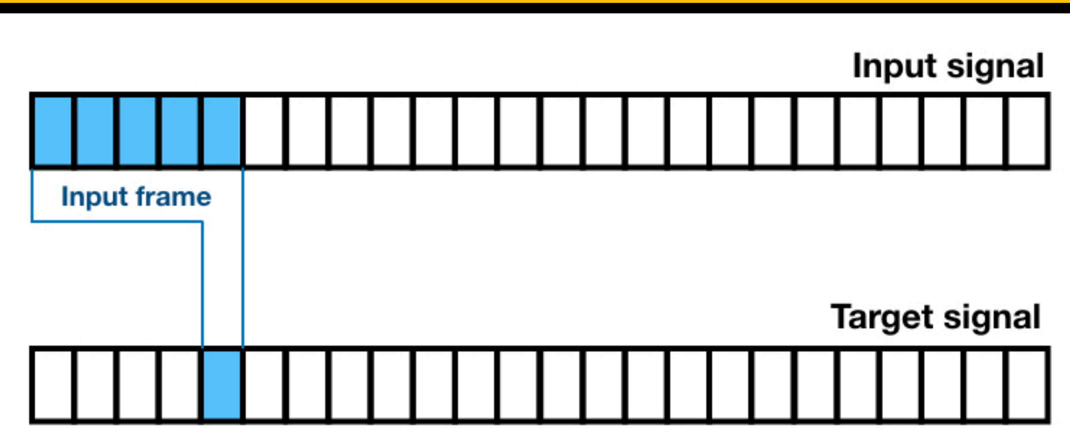
- Preliminary attempt: Mendoza.
  - Dataset: Whitenoise and sine sweeps. FIR highpass filter.
  - Model: Multilayer perceptron. Frame to frame prediction.
  - Results: Not able to model transfer function perfectly.
  - Conclusion: Whitenoise / sweeps not good dataset. Switch to real world music signals.
- Equalization using CNN: Martinez.
  - Dataset: Recorded piano notes. lowpass, highpass, peaking & shelving filters.
  - Model: Similar to an autoencoder with locally connected layers.
  - Results: MSE of the order of 1e-3.

DATASET

- Train set: Whitenoise
  - All frequencies: Flat magnitude spectrum.
  - All amplitudes: Samples drawn from uniform distribution.
- Test set:
  - Whitenoise.
  - Test suite (Short music excerpts and test signals): orchestra.wav, piano.wav, summer-violins-short.wav, sweep.wav, sweep-logarithmic.wav, multiSines.wav.
  - Test suite helps to see if network can generalize to real world audio and gives better insight through visualization.
- Filters:
  - IIR First order: lowpass.
  - IIR Second order: lowpass, highpass, lowshelf, highshelf, peaking, notch.

MODEL

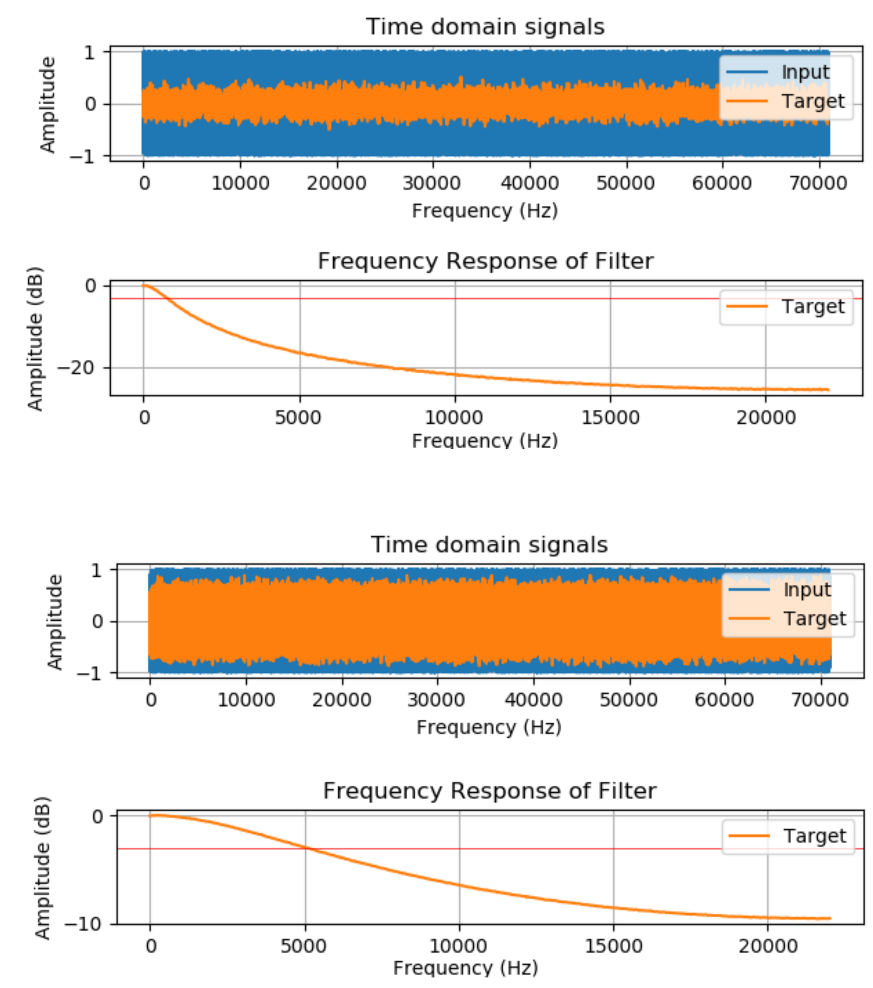
- Multi-layer perceptron:
  - Input size: 1024
  - Hidden layer size: 100
  - ReLU *
  - Output layer size: 1



- Input frame serves as context. Only previous samples provided since we are dealing with a causal system.
- Analogous to FIR system since no feedback provided to the network.
- Loss metric: Mean Squared Error
  - 16 bit audio; MSE of order 1e-10 ideal.

EXPERIMENTS

- First order lowpass filter:



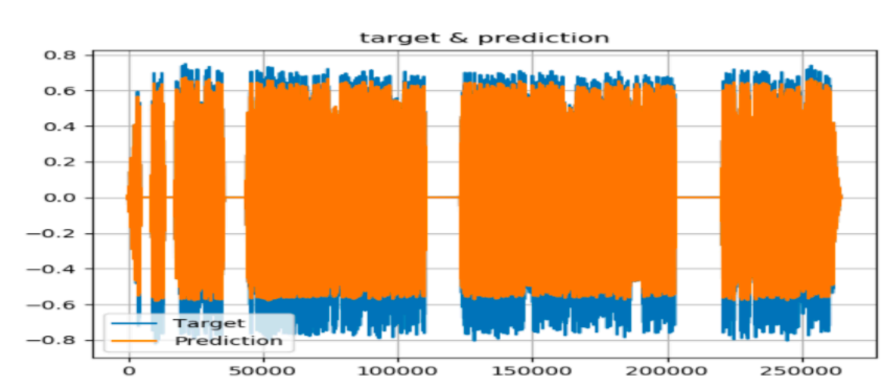
- **Input size tests:** (decay = 0.9)

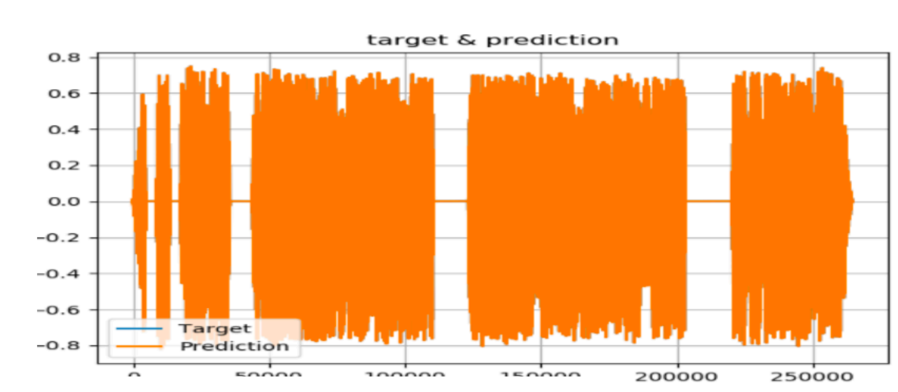
Input size	MSE
16	6.675e-4
32	2.186e-05
64	6.131e-07
128	1.675e-08
256	2.710e-08
512	6.317e-09
1024	4.782e-07

- Input size analogous to order of filter.
- Higher decays like 0.99 (steeper transfer functions) need larger input size.

Input size	MSE
128	1.030e-03
512	9.287e-07
1024	1.677e-10

- **Whitenoise Range:**
  - Values approx. above 0.5 and below -0.5 not being predicted.
  - Reason: Training data.
  - Solution: Increase range of input so that range of target values is close to -1 to 1.





File	MSE $(range = -1\ to\ 1)$	MSE $(range = -2\ to\ 2)$
piano.wav	7.502e-10	3.482e-10
orchestra.wav	3.772e-05	3.674e-10
summer-violins.wav	3.556e-06	3.208e-10
multiSines.wav	4.498e-04	4.858e-10
sweep.wav	5.886e-04	3.590e-10
sweep-logarithmic.wav	1.551e-02	1.020e-09

- **Removal of ReLU:**
  - Negative values lost. Only Output layer weights responsible for negative output.
  - Longer convergence time. (Almost instant without ReLU)
  - Filters are linear systems. No need of non-linearity in network.
  - But other types of effects may require non-linearity. So network becomes specific to filtering.

RESULTS

- Final network configuration:
  - Input size: 1024
  - No ReLU
- Results beat CNN based architecture.

Filter	f0 (Hz)	MSE
lowpass	10000	1.700e-13
	5000	1.279e-13
	1000	6.777e-14
	500	5.214e-14
	100	4.277e-10
	75	4.537e-08
highpass	10000	1.961e-13
	15000	1.550e-13
	21050	6.794e-14
	21550	8.626e-14
	21950	3.457e-10
	21975	4.697e-08

Filter	gain(dB)	Q	S	MSE
lowshelf	10	-	1	2.773e-13
highshelf	10	-	1	1.316e-12
peaking	10	0.707	-	3.096e-13
notch	-10	0.707	-	2.852e-13

FUTURE WORK

- One network to rule them all: Should be able to model both filtering and non-linear effects like waveshapers.
- Waveshapers do best with input size 1 and ReLU. How to bring the two architectures together?
- Challenge: What to do in case of chained effects? Don't want to learn separately.

CONTACT INFORMATION

Kaushal Sali, Music Informatics Group, Georgia Tech Center for Music Technology
ksali3@gatech.edu
Repository: https://bitbucket.org/kaushalsali/waveshaper/src/filter/