Neural audio synthesis

WaveNet, SampleRNN, and the waveform domain

Sevag Hanssian

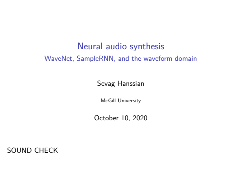
McGill University

October 10, 2020

SOUND CHECK

2020-10-10

Neural audio synthesis



Neural audio synthesis
WaveNet, SampleRNN, and the waveform domain

Sevag Hanssian

McGill University

October 10, 2020

SOUND CHECK

Physical sound waves

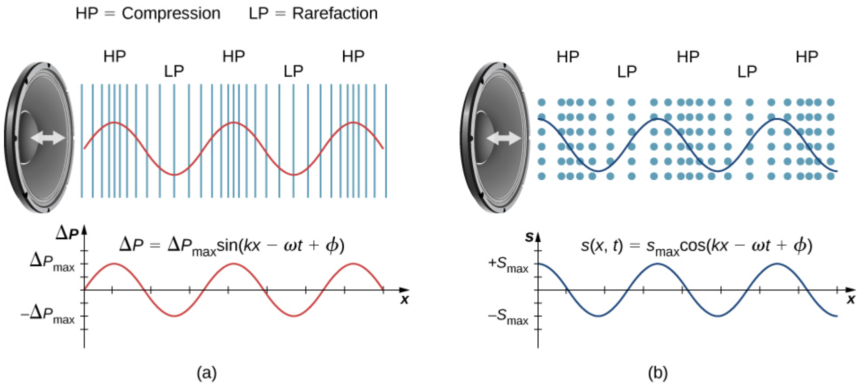


Figure: a) air pressure vs. distance, b) air molecule displacement[1]

[1] *17.1 Sound Waves*. URL: https://courses.lumenlearning.com/suny-osuniversityphysics/chapter/17-1-sound-waves/.

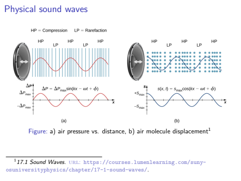
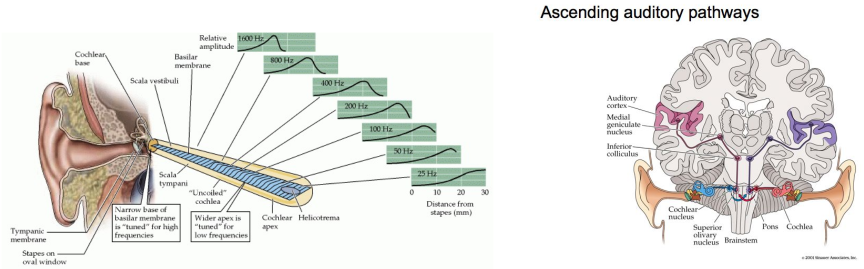

Physical sound waves

Figure: a) air pressure vs. distance, b) air molecule displacement[1]

[1] *17.1 Sound Waves*. URL: https://courses.lumenlearning.com/suny-osuniversityphysics/chapter/17-1-sound-waves/.

- Sound waves can be modelled by air pressure variations or air molecule displacement

Waveforms (human pov)



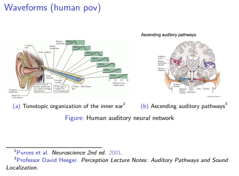
(a) Tonotopic organization of the inner ear[2]

(b) Ascending auditory pathways[3]

Figure: Human auditory neural network

[2] Purves et al. *Neuroscience 2nd ed.* 2001.

[3] Professor David Heeger. *Perception Lecture Notes: Auditory Pathways and Sound Localization.*


Waveforms (human pov)

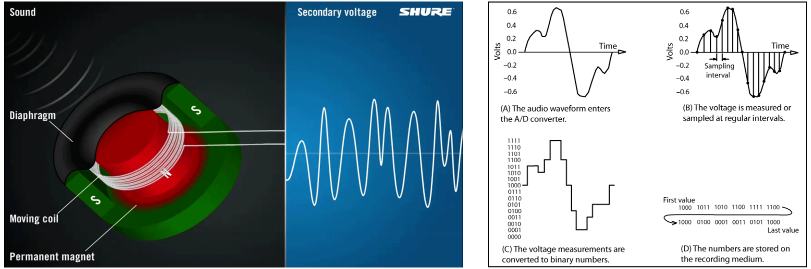
(a) Tonotopic organization of the inner ear[2] (b) Ascending auditory pathways[3]

Figure: Human auditory neural network

[2] Purves et al. *Neuroscience 2nd ed.* 2001.
[3] Professor David Heeger. *Perception Lecture Notes: Auditory Pathways and Sound Localization.*

- Hearing is when our ears detect sound pressure variations
- Vibrates our basilar membrane by frequency component
- Travels up the auditory nerve for further processing (so we can recognize and enjoy speech, music, etc.)
- An actual "neural network"

Waveforms (digital pov)

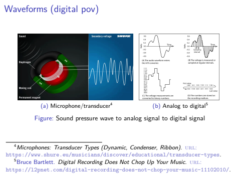


(a) Microphone/transducer[4] (b) Analog to digital[5]

Figure: Sound pressure wave to analog signal to digital signal

[4] *Microphones: Transducer Types (Dynamic, Condenser, Ribbon)*. URL: https://www.shure.eu/musicians/discover/educational/transducer-types.

[5] Bruce Bartlett. *Digital Recording Does Not Chop Up Your Music*. URL: https://l2pnet.com/digital-recording-does-not-chop-up-your-music-11102010/.


Waveforms (digital pov)
(a) Microphone/transducer[4] (b) Analog to digital[5]
Figure: Sound pressure wave to analog signal to digital signal
[4] Microphones: Transducer Types (Dynamic, Condenser, Ribbon). URL: https://www.shure.eu/musicians/discover/educational/transducer-types.
[5] Bruce Bartlett. Digital Recording Does Not Chop Up Your Music. URL: https://l2pnet.com/digital-recording-does-not-chop-up-your-music-11102010/.

- Microphones are transducers – convert sound pressure variations into an analog electric signal
- Analog signals are continuous in both time and range of values, so it must be sampled and quantized (pulse code modulation), Nyquist-Shannon sampling theorem – the fundamental bridge between continuous-time and discrete time
- Process is done in reverse for taking this digital signal and outputting it through speakers/headphones
- Wave to waveform

Waveform domain

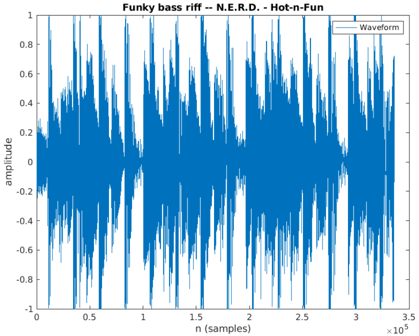


Figure: Bass riff waveform (CLICK2PLAY)[6]

How meaningful is it?

```
>> disp(x(50:128)); 0.0609, 0.0456, 0.0326, 0.0375, ...
```

[6] *N.E.R.D. - Hot-n-fun (Official Music Video) ft. Nelly Furtado – YouTube.* URL: https://www.youtube.com/watch?v=F6ZfA5QZDHY.

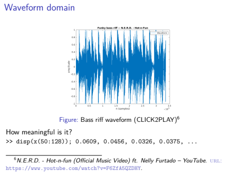
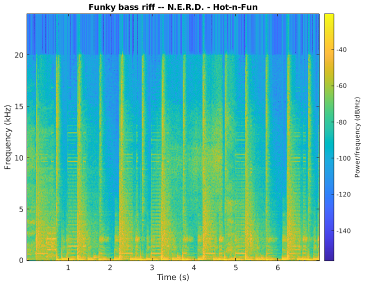

Waveform domain

Figure: Bass riff waveform (CLICK2PLAY)[6]

How meaningful is it?
>> disp(x(50:128)); 0.0609, 0.0456, 0.0326, 0.0375, ...

[6] N.E.R.D. - Hot-n-fun (Official Music Video) ft. Nelly Furtado – YouTube. URL: https://www.youtube.com/watch?v=F6ZfA5QZDHY.

Frequency domain

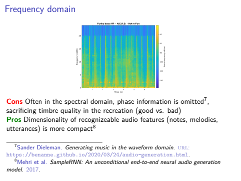


Cons Often in the spectral domain, phase information is omitted[7], sacrificing timbre quality in the recreation (good vs. bad)

Pros Dimensionality of recognizeable audio features (notes, melodies, utterances) is more compact[8]

[7] Sander Dieleman. *Generating music in the waveform domain*. URL: https://benanne.github.io/2020/03/24/audio-generation.html.

[8] Mehri et al. *SampleRNN: An unconditional end-to-end neural audio generation model*. 2017.


Frequency domain
Cons Often in the spectral domain, phase information is omitted[7], sacrificing timbre quality in the recreation (good vs. bad)
Pros Dimensionality of recognizeable audio features (notes, melodies, utterances) is more compact[8]
[7] Sander Dieleman. Generating music in the waveform domain. URL: https://benanne.github.io/2020/03/24/audio-generation.html.
[8] Mehri et al. SampleRNN: An unconditional end-to-end neural audio generation model. 2017.

- We discard the phase component when analyzing audio, because it is not informative for most of the things we could be interested in. phase is very important because it meaningfully affects our perception **when generating sound**. phase hard because
  - it is an angle between 0 and 2pi and wraps around (princarg)
  - it becomes effectively random as the magnitude tends towards 0, because noise starts to dominate;
  - absolute phase is less meaningful, but relative phase differences over time matter perceptually.
- Counterpoint is that modeling the waveform domain implicitly preserves phase information, perceptually important
- Counterpoint is that the waveform needs hundres of thousands of samples for simple audio features – from the sampleRNN paper, one of the primary methods i'll be introducing today

Symbolic domain

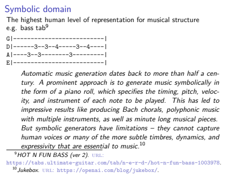
The highest human level of representation for musical structure
e.g. bass tab[9]

```
G|--------------------------|
D|------3--3--4-----3--4----|
A|----3--3--------3---------|
E|--------------------------|
```

> *Automatic music generation dates back to more than half a century. A prominent approach is to generate music symbolically in the form of a piano roll, which specifies the timing, pitch, velocity, and instrument of each note to be played. This has led to impressive results like producing Bach chorals, polyphonic music with multiple instruments, as well as minute long musical pieces. But symbolic generators have limitations – they cannot capture human voices or many of the more subtle timbres, dynamics, and expressivity that are essential to music.*[10]

[9] *HOT N FUN BASS (ver 2)*. URL: https://tabs.ultimate-guitar.com/tab/n-e-r-d-/hot-n-fun-bass-1003978.

[10] *Jukebox*. URL: https://openai.com/blog/jukebox/.



Symbolic domain
The highest human level of representation for musical structure
e.g. bass tab[9]
G|--------------------------|
D|------3--3--4-----3--4----|
A|----3--3--------3---------|
E|--------------------------|
Automatic music generation dates back to more than half a century. A prominent approach is to generate music symbolically in the form of a piano roll, which specifies the timing, pitch, velocity, and instrument of each note to be played. This has led to impressive results like producing Bach chorals, polyphonic music with multiple instruments, as well as minute long musical pieces. But symbolic generators have limitations – they cannot capture human voices or many of the more subtle timbres, dynamics, and expressivity that are essential to music.[10]
[9] HOT N FUN BASS (ver 2). URL: https://tabs.ultimate-guitar.com/tab/n-e-r-d-/hot-n-fun-bass-1003978.
[10] Jukebox. URL: https://openai.com/blog/jukebox/.

- Algorithmic music composition – MUMT 306 material
- I'm not even aware of how many of these exist. Tabs, sheet music, etc.

Interlude: machine learning in acoustics

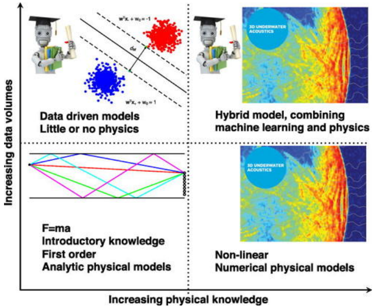


Figure: Analytic physical models (lower left) give basic insights about physical systems. More sophisticated models, reliant on computational methods (lower right), can model more complex phenomena. Whereas physical models are reliant on rules, which are updated by physical evidence (data), ML is purely data-driven (upper left). By augmenting ML methods with physical models to obtain hybrid models (upper right), a synergy of the strengths of physical intuition and data-driven insights can be obtained[11]

[11] Bianco et al. "Machine learning in acoustics: theory and applications". In: *Acoustical Society of America* (2019).

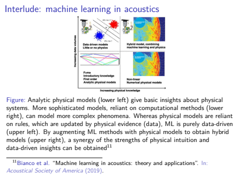

Interlude: machine learning in acoustics

Figure: Analytic physical models (lower left) give basic insights about physical systems. More sophisticated models, reliant on computational methods (lower right), can model more complex phenomena. Whereas physical models are reliant on rules, which are updated by physical evidence (data), ML is purely data-driven (upper left). By augmenting ML methods with physical models to obtain hybrid models (upper right), a synergy of the strengths of physical intuition and data-driven insights can be obtained[11]

[11] Bianco et al. "Machine learning in acoustics: theory and applications". In: *Acoustical Society of America* (2019).

- First two papers i'll discuss, purely neural audio synthesis, are from the top left. Unstructured, whacky, figure out what you can from the provided audio
- Lastly I'll mention an alternative approach, top right – mixing our known acoustical physical models with machine learning and deep learning to train on real audio and discover optimal parameters

Sound synthesis – wavetables and sinusoidal oscillators

Wavetable synthesis is perhaps the oldest technique for creating sounds with computers. It involves **the storage of a single period of a periodic waveform** in a circular buffer. By varying the “speed” with which a read pointer is advanced through the buffer, one can achieve output waveforms of different frequencies[12].

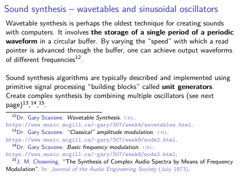
Sound synthesis algorithms are typically described and implemented using primitive signal processing “building blocks” called **unit generators**. Create complex synthesis by combining multiple oscillators (see next page)[13],[14],[15].

[12] Dr. Gary Scavone. *Wavetable Synthesis*. URL: https://www.music.mcgill.ca/~gary/307/week4/wavetables.html.

[13] Dr. Gary Scavone. *“Classical” amplitude modulation*. URL: https://www.music.mcgill.ca/~gary/307/week9/node2.html.

[14] Dr. Gary Scavone. *Basic frequency modulation*. URL: https://www.music.mcgill.ca/~gary/307/week9/node2.html.

[15] J. M. Chowning. “The Synthesis of Complex Audio Spectra by Means of Frequency Modulation”. In: *Journal of the Audio Engineering Society* (July 1973).



Sound synthesis – wavetables and sinusoidal oscillators

Wavetable synthesis is perhaps the oldest technique for creating sounds with computers. It involves **the storage of a single period of a periodic waveform** in a circular buffer. By varying the “speed” with which a read pointer is advanced through the buffer, one can achieve output waveforms of different frequencies[12].

Sound synthesis algorithms are typically described and implemented using primitive signal processing “building blocks” called **unit generators**. Create complex synthesis by combining multiple oscillators (see next page)[13],[14],[15].

[12] Dr. Gary Scavone. *Wavetable Synthesis*. URL: https://www.music.mcgill.ca/~gary/307/week4/wavetables.html.
[13] Dr. Gary Scavone. *“Classical” amplitude modulation*. URL: https://www.music.mcgill.ca/~gary/307/week9/node2.html.
[14] Dr. Gary Scavone. *Basic frequency modulation*. URL: https://www.music.mcgill.ca/~gary/307/week9/node2.html.
[15] J. M. Chowning. “The Synthesis of Complex Audio Spectra by Means of Frequency Modulation”. In: *Journal of the Audio Engineering Society* (July 1973).

307 material – sound synthesis

- Can simulate more complex sound with multiple wavetables, envelopes, unit generators, oscillators.
- So, the computer is not *creating* a new waveform – it’s modifying one already supplied to create outputs with different frequencies.
- Similarly, sine wave generators generate a simple basic waveform on which operations are applied and combined to create complex sounds
- Or using FFT synthesis (summation of sines)
- We’re either using an existing waveform or combining simple waveforms, through frequency domain principles, to generate complex waveforms

Sound synthesis – sinusoidal oscillators

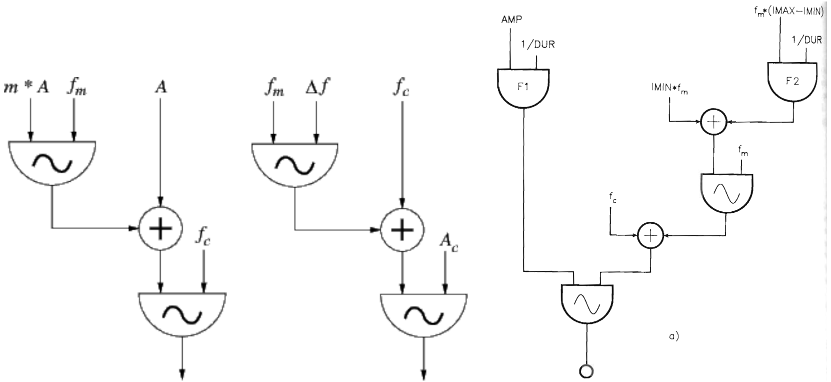
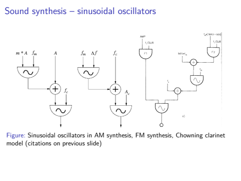


Figure: Sinusoidal oscillators in AM synthesis, FM synthesis, Chowning clarinet model (citations on previous slide)

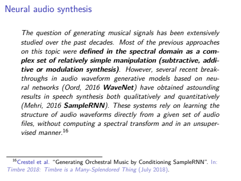

Sound synthesis – sinusoidal oscillators
Figure: Sinusoidal oscillators in AM synthesis, FM synthesis, Chowning clarinet model (citations on previous slide)

In 1967, Chowning realized that complex sounds could be generated using only two oscillators when the output of one oscillator is connected to the frequency input of a second oscillator. Essentially, the first oscillator generates a pure tone that modulates the frequency of the second oscillator in a way that produces a complex tone, like a string vibrating.

Neural audio synthesis

> *The question of generating musical signals has been extensively studied over the past decades. Most of the previous approaches on this topic were* ***defined in the spectral domain as a complex set of relatively simple manipulation (subtractive, additive or modulation synthesis)****. However, several recent breakthroughs in audio waveform generative models based on neural networks (Oord, 2016* ***WaveNet****) have obtained astounding results in speech synthesis both qualitatively and quantitatively (Mehri, 2016* ***SampleRNN****). These systems rely on learning the structure of audio waveforms directly from a given set of audio files, without computing a spectral transform and in an unsupervised manner.*[16]

[16] Crestel et al. "Generating Orchestral Music by Conditioning SampleRNN". In: *Timbre 2018: Timbre is a Many-Splendored Thing* (July 2018).



Neural audio synthesis

The question of generating musical signals has been extensively studied over the past decades. Most of the previous approaches on this topic were ***defined in the spectral domain as a complex set of relatively simple manipulation (subtractive, additive or modulation synthesis)****. However, several recent breakthroughs in audio waveform generative models based on neural networks (Oord, 2016* ***WaveNet****) have obtained astounding results in speech synthesis both qualitatively and quantitatively (Mehri, 2016* ***SampleRNN****). These systems rely on learning the structure of audio waveforms directly from a given set of audio files, without computing a spectral transform and in an unsupervised manner.*[16]

[16] Crestel et al. "Generating Orchestral Music by Conditioning SampleRNN". In: *Timbre 2018: Timbre is a Many-Splendored Thing* (July 2018).

- In traditional synthesis we create complex waveforms by combining simple ones
- In 2018, we have the computational power and models to make sense of the waveform directly

WaveNet

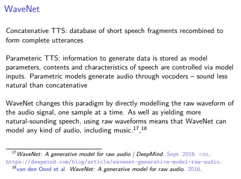
Concatenative TTS: database of short speech fragments recombined to form complete utterances

Parameteric TTS: information to generate data is stored as model parameters, contents and characteristics of speech are controlled via model inputs. Parametric models generate audio through vocoders – sound less natural than concatenative

WaveNet changes this paradigm by directly modelling the raw waveform of the audio signal, one sample at a time. As well as yielding more natural-sounding speech, using raw waveforms means that WaveNet can model any kind of audio, including music.[17],[18]

[17] *WaveNet: A generative model for raw audio | DeepMind*. Sept. 2016. URL: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio.

[18] van den Oord et al. *WaveNet: A generative model for raw audio*. 2016.



WaveNet

Concatenative TTS: database of short speech fragments recombined to form complete utterances

Parameteric TTS: information to generate data is stored as model parameters, contents and characteristics of speech are controlled via model inputs. Parametric models generate audio through vocoders – sound less natural than concatenative

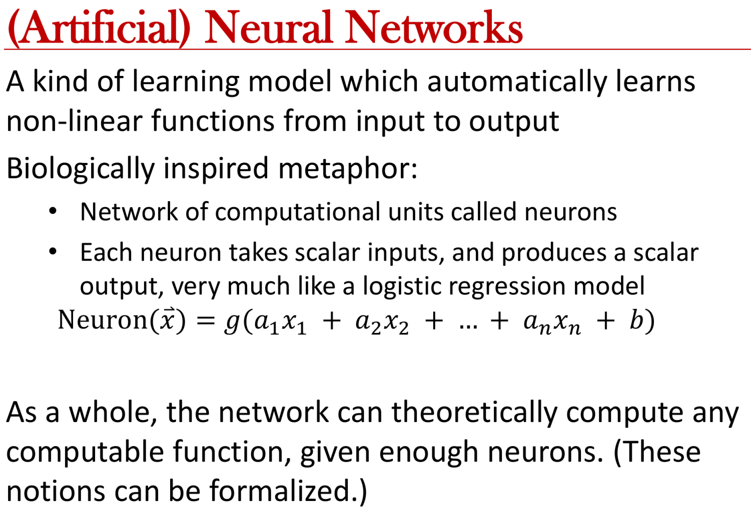
WaveNet changes this paradigm by directly modelling the raw waveform of the audio signal, one sample at a time. As well as yielding more natural-sounding speech, using raw waveforms means that WaveNet can model any kind of audio, including music.[17],[18]

[17] *WaveNet: A generative model for raw audio | DeepMind*. Sept. 2016. URL: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio.

[18] van den Oord et al. *WaveNet: A generative model for raw audio*. 2016.

- Introduced by DeepMind London (Google) in September 2016.
- Derives from PixelCNN – similarity with audio, large arrays of numbers with patterns
- TTS = text-to-speech
- difficult to modify the voice (for example switching to a different speaker, or altering the emphasis or emotion of their speech) without recording a whole new database

Artificial neural networks



(Artificial) Neural Networks

A kind of learning model which automatically learns non-linear functions from input to output

Biologically inspired metaphor:

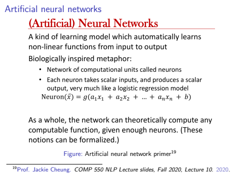
- Network of computational units called neurons
- Each neuron takes scalar inputs, and produces a scalar output, very much like a logistic regression model

$$\text{Neuron}(\vec{x}) = g(a_1x_1 + a_2x_2 + \dots + a_nx_n + b)$$

As a whole, the network can theoretically compute any computable function, given enough neurons. (These notions can be formalized.)

Figure: Artificial neural network primer[19]

[19] Prof. Jackie Cheung. *COMP 550 NLP Lecture slides, Fall 2020, Lecture 10.* 2020.



Artificial neural networks
(Artificial) Neural Networks
A kind of learning model which automatically learns non-linear functions from input to output
Biologically inspired metaphor:
• Network of computational units called neurons
• Each neuron takes scalar inputs, and produces a scalar output, very much like a logistic regression model
Neuron($\vec{x}$) = $g(a_1x_1 + a_2x_2 + \dots + a_nx_n + b)$
As a whole, the network can theoretically compute any computable function, given enough neurons. (These notions can be formalized.)
Figure: Artificial neural network primer[19]
[19] Prof. Jackie Cheung. COMP 550 NLP Lecture slides, Fall 2020, Lecture 10. 2020.

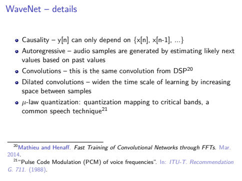
- I can't tell you its meaningful, but my brain certainly knows it.
- Psychoacoustics – after the inputs to the system (basilar membrane etc.) are activated, the end result is that sound creates a bunch of firing neurons that travel "up the
- Unsupervised learning – you only had to listen to music to figure out patterns. Nobody had to tell you what patterns to look for
- Contrast to FEM – did a very light bit of research, not enough...

WaveNet – details

- Causality – y[n] can only depend on {x[n], x[n-1], ...}
- Autoregressive – audio samples are generated by estimating likely next values based on past values
- Convolutions – this is the same convolution from DSP[20]
- Dilated convolutions – widen the time scale of learning by increasing space between samples
- μ-law quantization: quantization mapping to critical bands, a common speech technique[21]

[20] Mathieu and Henaff. *Fast Training of Convolutional Networks through FFTs*. Mar. 2014.

[21] "Pulse Code Modulation (PCM) of voice frequencies". In: *ITU-T. Recommendation G. 711.* (1988).



WaveNet – details

- Causality – y[n] can only depend on {x[n], x[n-1], ...}
- Autoregressive – audio samples are generated by estimating likely next values based on past values
- Convolutions – this is the same convolution from DSP[20]
- Dilated convolutions – widen the time scale of learning by increasing space between samples
- μ-law quantization: quantization mapping to critical bands, a common speech technique[21]

[20] Mathieu and Henaff. *Fast Training of Convolutional Networks through FFTs*. Mar. 2014.

[21] "Pulse Code Modulation (PCM) of voice frequencies". In: *ITU-T. Recommendation G. 711.* (1988).

- makes sense for audio which moves forward in time
- Convolution and correlation are related. The goal is to find repeating strong patterns in waveform data by doing convolutions somewhere in the neural network's hidden layers
- Mu-law isnt used in sampleRNN, only wavenet

WaveNet diagram

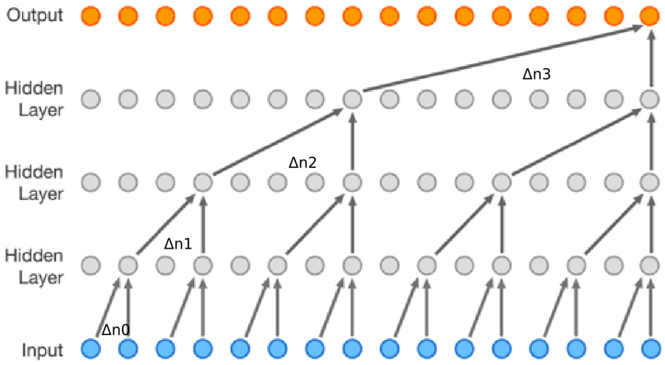


Figure: How wavenet computes samples of output y[n] from historical samples {x[n], x[n-1], ...}

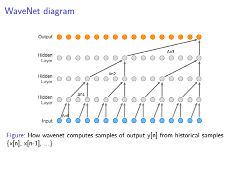

WaveNet diagram

Figure: How wavenet computes samples of output y[n] from historical samples {x[n], x[n-1], ...}

- Same diagram as wavenet, i added the delta t annotations
- delta t0 = 1 sample timescale (patterns in 1/fs = dT)
- delta t1 = 2 sample timescale, etc.

Inference in a FF Neural Network

Perform computations forwads through the graph:

$$\mathbf{h^1} = g^1(\mathbf{xW^1} + \mathbf{b^1})$$
$$\mathbf{h^2} = g^2(\mathbf{h^1W^2} + \mathbf{b^2})$$
$$\mathbf{y} = \mathbf{h^2W^3}$$

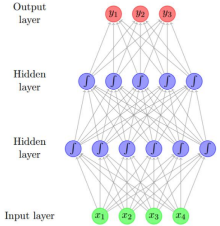


Figure 2: Feed-forward neural network with two hidden layers.

Note that we are now representing each layer as a vector; combining all of the weights in a layer across the units into a weight matrix

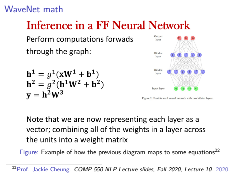
Figure: Example of how the previous diagram maps to some equations[22]

[22] Prof. Jackie Cheung. *COMP 550 NLP Lecture slides, Fall 2020, Lecture 10.* 2020.

Neural audio synthesis

2020-10-10

└WaveNet math


WaveNet math

Inference in a FF Neural Network

Perform computations forwads through the graph:

$\mathbf{h^1} = g^1(\mathbf{xW^1} + \mathbf{b^1})$
$\mathbf{h^2} = g^2(\mathbf{h^1W^2} + \mathbf{b^2})$
$\mathbf{y} = \mathbf{h^2W^3}$

Note that we are now representing each layer as a vector; combining all of the weights in a layer across the units into a weight matrix

Figure: Example of how the previous diagram maps to some equations[22]

[22] Prof. Jackie Cheung. *COMP 550 NLP Lecture slides, Fall 2020, Lecture 10.* 2020.

WaveNet examples

All examples are from WaveNet blog post[23]

- Unconditioned speech (i.e. let the machine do whatever it wants): CLICK TO PLAY
- Conditioned speech, English (i.e. train machine to learn a specific phrase*): CLICK TO PLAY
- Conditioned speech, Mandarin: CLICK TO PLAY
- Unconditional music**: CLICK TO PLAY
- Unconditional music: CLICK TO PLAY

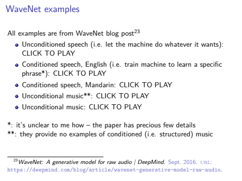
*: it's unclear to me how – the paper has precious few details
**: they provide no examples of conditioned (i.e. structured) music

[23] *WaveNet: A generative model for raw audio | DeepMind*. Sept. 2016. URL: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio.

Neural audio synthesis

2020-10-10

└WaveNet examples



WaveNet examples

All examples are from WaveNet blog post[23]

- Unconditioned speech (i.e. let the machine do whatever it wants): CLICK TO PLAY
- Conditioned speech, English (i.e. train machine to learn a specific phrase*): CLICK TO PLAY
- Conditioned speech, Mandarin: CLICK TO PLAY
- Unconditional music**: CLICK TO PLAY
- Unconditional music: CLICK TO PLAY

*: it's unclear to me how – the paper has precious few details
**: they provide no examples of conditioned (i.e. structured) music

[23] *WaveNet: A generative model for raw audio | DeepMind*. Sept. 2016. URL: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio.

SampleRNN

Use RNNs to learn long-term patterns:

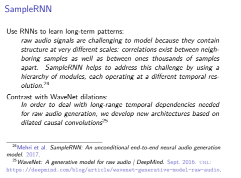
> *raw audio signals are challenging to model because they contain structure at very different scales: correlations exist between neighboring samples as well as between ones thousands of samples apart. SampleRNN helps to address this challenge by using a hierarchy of modules, each operating at a different temporal resolution.*[24]

Contrast with WaveNet dilations:

> *In order to deal with long-range temporal dependencies needed for raw audio generation, we develop new architectures based on dilated causal convolutions*[25]

[24] Mehri et al. *SampleRNN: An unconditional end-to-end neural audio generation model*. 2017.

[25] *WaveNet: A generative model for raw audio | DeepMind*. Sept. 2016. URL: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio.



SampleRNN

Use RNNs to learn long-term patterns:
raw audio signals are challenging to model because they contain structure at very different scales: correlations exist between neighboring samples as well as between ones thousands of samples apart. SampleRNN helps to address this challenge by using a hierarchy of modules, each operating at a different temporal resolution.[24]

Contrast with WaveNet dilations:
In order to deal with long-range temporal dependencies needed for raw audio generation, we develop new architectures based on dilated causal convolutions[25]

[24] Mehri et al. *SampleRNN: An unconditional end-to-end neural audio generation model*. 2017.
[25] *WaveNet: A generative model for raw audio | DeepMind*. Sept. 2016. URL: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio.

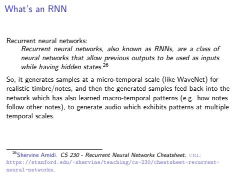
What's an RNN

Recurrent neural networks:

> *Recurrent neural networks, also known as RNNs, are a class of neural networks that allow previous outputs to be used as inputs while having hidden states.*[26]

So, it generates samples at a micro-temporal scale (like WaveNet) for realistic timbre/notes, and then the generated samples feed back into the network which has also learned macro-temporal patterns (e.g. how notes follow other notes), to generate audio which exhibits patterns at multiple temporal scales.

[26] Shervine Amidi. *CS 230 - Recurrent Neural Networks Cheatsheet*. URL: https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks.



What's an RNN

Recurrent neural networks:

Recurrent neural networks, also known as RNNs, are a class of neural networks that allow previous outputs to be used as inputs while having hidden states.[26]

So, it generates samples at a micro-temporal scale (like WaveNet) for realistic timbre/notes, and then the generated samples feed back into the network which has also learned macro-temporal patterns (e.g. how notes follow other notes), to generate audio which exhibits patterns at multiple temporal scales.

[26] Shervine Amidi. *CS 230 - Recurrent Neural Networks Cheatsheet*. URL: https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks.

- Different scales of patterns – that's music exactly
- The main insight behind SampleRNN is that audio in particular exhibits salient patterns at multiple timescales, and that these patterns and features are composed hierarchically. For example, timbre is shaped by patterns over very short timescales, while musical events and gestures like bowing or striking an instrument happen over longer timescales. Melodies are composed of a series of musical events, which are often grouped into compositional structures like bars or phrases, which are further grouped into sections and then full songs. (Karl Hiner)

SampleRNN diagram

Note how similar it is to WaveNet. WaveNet “dilates” the convolutions to go from a small time step to a large one – SampleRNN “upsamples” the vector to bump up the temporal resolution

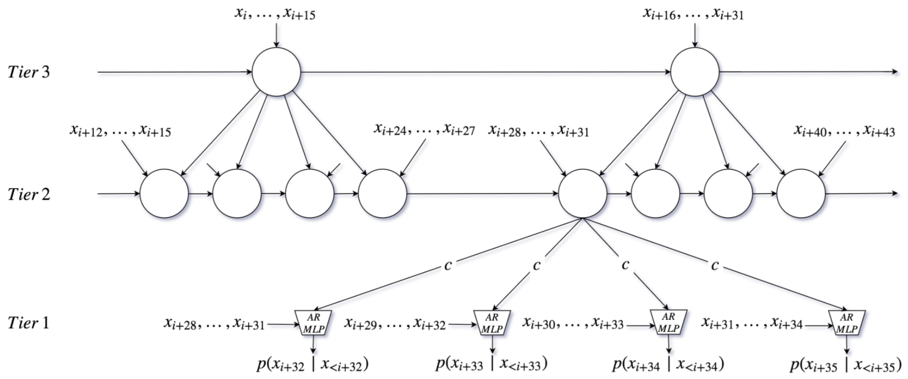

$x_i, \ldots, x_{i+15}$
$x_{i+16}, \ldots, x_{i+31}$
Tier 3
$x_{i+12}, \ldots, x_{i+15}$
$x_{i+24}, \ldots, x_{i+27}$
$x_{i+28}, \ldots, x_{i+31}$
$x_{i+40}, \ldots, x_{i+43}$
Tier 2
c
c
c
c
Tier 1
$x_{i+28}, \ldots, x_{i+31}$
AR MLP
$x_{i+29}, \ldots, x_{i+32}$
AR MLP
$x_{i+30}, \ldots, x_{i+33}$
AR MLP
$x_{i+31}, \ldots, x_{i+34}$
AR MLP
$p(x_{i+32} \mid x_{<i+32})$
$p(x_{i+33} \mid x_{<i+33})$
$p(x_{i+34} \mid x_{<i+34})$
$p(x_{i+35} \mid x_{<i+35})$

Figure: How SampleRNN computes samples of output y[n] from historical samples {x[n], x[n-1], ...}

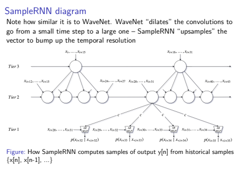

SampleRNN diagram
Note how similar it is to WaveNet. WaveNet “dilates” the convolutions to go from a small time step to a large one – SampleRNN “upsamples” the vector to bump up the temporal resolution

Figure: How SampleRNN computes samples of output y[n] from historical samples {x[n], x[n-1], ...}

- For me, dilation (spreading out the input signal) is the same as upsampling (spreading out the input signal)
- Notice how the diagram shows the broad similarity in their architecture

SampleRNN in practise

So what does SampleRNN actually look like? Up-to-date implementation, PRiSM SampleRNN based on Tensorflow 2[27],[28]. Your best bet at getting up and running (works for me).

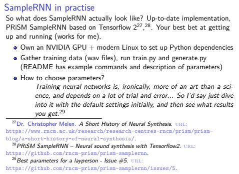
- Own an NVIDIA GPU + modern Linux to set up Python dependencies
- Gather training data (wav files), run train.py and generate.py (README has example commands and description of parameters)
- How to choose parameters?

> *Training neural networks is, ironically, more of an art than a science, and depends on a lot of trial and error... So I'd say just dive into it with the default settings initially, and then see what results you get.*[29]

[27] Dr. Christopher Melen. *A Short History of Neural Synthesis.* URL: https://www.rncm.ac.uk/research/research-centres-rncm/prism/prism-blog/a-short-history-of-neural-synthesis/.

[28] *PRiSM SampleRNN – Neural sound synthesis with Tensorflow2.* URL: https://github.com/rncm-prism/prism-samplernn.

[29] *Best parameters for a layperson - Issue #5.* URL: https://github.com/rncm-prism/prism-samplernn/issues/5.



SampleRNN in practise
So what does SampleRNN actually look like? Up-to-date implementation, PRiSM SampleRNN based on Tensorflow 2[27],[28]. Your best bet at getting up and running (works for me).
- Own an NVIDIA GPU + modern Linux to set up Python dependencies
- Gather training data (wav files), run train.py and generate.py (README has example commands and description of parameters)
- How to choose parameters?
Training neural networks is, ironically, more of an art than a science, and depends on a lot of trial and error... So I'd say just dive into it with the default settings initially, and then see what results you get.[29]
[27] Dr. Christopher Melen. A Short History of Neural Synthesis. URL: https://www.rncm.ac.uk/research/research-centres-rncm/prism/prism-blog/a-short-history-of-neural-synthesis/.
[28] PRiSM SampleRNN – Neural sound synthesis with Tensorflow2. URL: https://github.com/rncm-prism/prism-samplernn.
[29] Best parameters for a layperson - Issue #5. URL: https://github.com/rncm-prism/prism-samplernn/issues/5.

- Science vs. art?
- What's the line between science, art, and magic?
- Is it art if I have no clue what's going on?

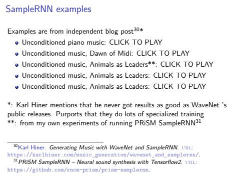
SampleRNN examples

Examples are from independent blog post[30]*

- Unconditioned piano music: CLICK TO PLAY
- Unconditioned music, Dawn of Midi: CLICK TO PLAY
- Unconditioned music, Animals as Leaders**: CLICK TO PLAY
- Unconditioned music, Animals as Leaders: CLICK TO PLAY
- Unconditioned music, Animals as Leaders: CLICK TO PLAY

*: Karl Hiner mentions that he never got results as good as WaveNet 's public releases. Purports that they do lots of specialized training
**: from my own experiments of running PRiSM SampleRNN[31]

[30] Karl Hiner. *Generating Music with WaveNet and SampleRNN*. URL: https://karlhiner.com/music_generation/wavenet_and_samplernn/.
[31] *PRiSM SampleRNN – Neural sound synthesis with Tensorflow2*. URL: https://github.com/rncm-prism/prism-samplernn.



SampleRNN examples

Examples are from independent blog post[30]*

- Unconditioned piano music: CLICK TO PLAY
- Unconditioned music, Dawn of Midi: CLICK TO PLAY
- Unconditioned music, Animals as Leaders**: CLICK TO PLAY
- Unconditioned music, Animals as Leaders: CLICK TO PLAY
- Unconditioned music, Animals as Leaders: CLICK TO PLAY

*: Karl Hiner mentions that he never got results as good as WaveNet 's public releases. Purports that they do lots of specialized training
**: from my own experiments of running PRiSM SampleRNN[31]

[30] Karl Hiner. *Generating Music with WaveNet and SampleRNN*. URL: https://karlhiner.com/music_generation/wavenet_and_samplernn/.
[31] *PRiSM SampleRNN – Neural sound synthesis with Tensorflow2*. URL: https://github.com/rncm-prism/prism-samplernn.

- The non-reproduceability problem personified?
- Hidden trade secrets?

Traits of WaveNet and SampleRNN

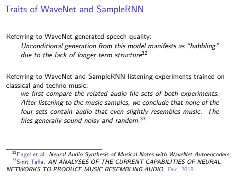
Referring to WaveNet generated speech quality:

> *Unconditional generation from this model manifests as "babbling" due to the lack of longer term structure*[32]

Referring to WaveNet and SampleRNN listening experiments trained on classical and techno music:

> *we first compare the related audio file sets of both experiments. After listening to the music samples, we conclude that none of the four sets contain audio that even slightly resembles music. The files generally sound noisy and random.*[33]

[32] Engel et al. *Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders*.
[33] Sinit Tafla. *AN ANALYSES OF THE CURRENT CAPABILITIES OF NEURAL NETWORKS TO PRODUCE MUSIC-RESEMBLING AUDIO*. Dec. 2018.



Traits of WaveNet and SampleRNN

Referring to WaveNet generated speech quality:
Unconditional generation from this model manifests as "babbling" due to the lack of longer term structure[32]

Referring to WaveNet and SampleRNN listening experiments trained on classical and techno music:
we first compare the related audio file sets of both experiments. After listening to the music samples, we conclude that none of the four sets contain audio that even slightly resembles music. The files generally sound noisy and random.[33]

[32] Engel et al. *Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders*.
[33] Sinit Tafla. *AN ANALYSES OF THE CURRENT CAPABILITIES OF NEURAL NETWORKS TO PRODUCE MUSIC-RESEMBLING AUDIO*. Dec. 2018.

- WaveNet and SampleRNN are good at learning on a micro-temporal level (up to individual instrument onsets, notes, timbres). Needs further conditioning to enforce longer-term structure (melody, song structure, etc.)
- Incoherent medium/longer-term structure:
- In that first paper, they start discussing ways of training WaveNet by enforcing longer-term structure

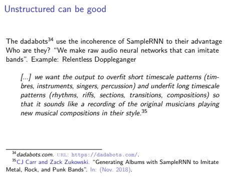
Unstructured can be good

The dadabots[34] use the incoherence of SampleRNN to their advantage Who are they? "We make raw audio neural networks that can imitate bands". Example: Relentless Doppleganger

> *[...] we want the output to overfit short timescale patterns (timbres, instruments, singers, percussion) and underfit long timescale patterns (rhythms, riffs, sections, transitions, compositions) so that it sounds like a recording of the original musicians playing new musical compositions in their style.*[35]

[34] *dadabots.com*. URL: https://dadabots.com/.

[35] CJ Carr and Zack Zukowski. "Generating Albums with SampleRNN to Imitate Metal, Rock, and Punk Bands". In: (Nov. 2018).



Unstructured can be good

The dadabots[34] use the incoherence of SampleRNN to their advantage Who are they? "We make raw audio neural networks that can imitate bands". Example: Relentless Doppleganger

[...] we want the output to overfit short timescale patterns (timbres, instruments, singers, percussion) and underfit long timescale patterns (rhythms, riffs, sections, transitions, compositions) so that it sounds like a recording of the original musicians playing new musical compositions in their style.[35]

[34] *dadabots.com*. URL: https://dadabots.com/.
[35] CJ Carr and Zack Zukowski. "Generating Albums with SampleRNN to Imitate Metal, Rock, and Punk Bands". In: (Nov. 2018).

This is my project idea. It'll be heavier on the literature – I'll need to deep dive to understand the innards of the machine learning/neural networks, but it's too soon to do a significant contribution or novel work – I'm new to machine learning. I want to create a fake artist with fake music using one or many machine learning-based approaches.

Towards structure

Jukebox[36],[37] addresses the long input problem with an autoencoder that compresses raw audio to a lower-dimensional space by discarding perceptually irrelevant information. Then trains a model to generate audio in this compressed space, and upsample back to the raw audio space.

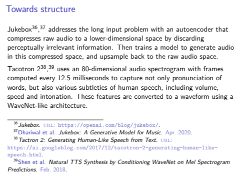
Tacotron 2[38],[39] uses an 80-dimensional audio spectrogram with frames computed every 12.5 milliseconds to capture not only pronunciation of words, but also various subtleties of human speech, including volume, speed and intonation. These features are converted to a waveform using a WaveNet-like architecture.

[36] *Jukebox*. URL: https://openai.com/blog/jukebox/.

[37] Dhariwal et al. *Jukebox: A Generative Model for Music*. Apr. 2020.

[38] *Tactron 2: Generating Human-Like Speech from Text*. URL: https://ai.googleblog.com/2017/12/tacotron-2-generating-human-like-speech.html.

[39] Shen et al. *Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions*. Feb. 2018.



Towards structure

Jukebox[36],[37] addresses the long input problem with an autoencoder that compresses raw audio to a lower-dimensional space by discarding perceptually irrelevant information. Then trains a model to generate audio in this compressed space, and upsample back to the raw audio space.

Tacotron 2[38],[39] uses an 80-dimensional audio spectrogram with frames computed every 12.5 milliseconds to capture not only pronunciation of words, but also various subtleties of human speech, including volume, speed and intonation. These features are converted to a waveform using a WaveNet-like architecture.

[36] Jukebox. URL: https://openai.com/blog/jukebox/.
[37] Dhariwal et al. Jukebox: A Generative Model for Music. Apr. 2020.
[38] Tactron 2: Generating Human-Like Speech from Text. URL: https://ai.googleblog.com/2017/12/tacotron-2-generating-human-like-speech.html.
[39] Shen et al. Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions. Feb. 2018.

- Note the 2 ways of dealing with wavenet shortcomings
- Jukebox relies on operating on a compressed version of the waveform, so it can become computationally feasible to learn longer-term (minutes) musical structure. Opens itself up to suffering from imperfect recreation - e.g. Griffin-Lim phase reconstruction
- Tactron 2 combines spectral features for long-term structure (utterances, words) and wavenet for the synthesis to the waveform. Best of both worlds

Recall

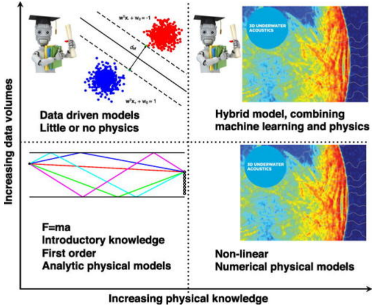


Figure: By augmenting ML methods with physical models to obtain hybrid models (upper right), a synergy of the strengths of physical intuition and data-driven insights can be obtained[40]

[40] Bianco et al. "Machine learning in acoustics: theory and applications". In: *Acoustical Society of America* (2019).

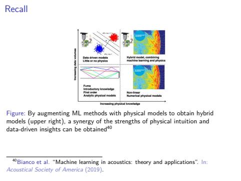

Recall

Figure: By augmenting ML methods with physical models to obtain hybrid models (upper right), a synergy of the strengths of physical intuition and data-driven insights can be obtained[40]

[40] Bianco et al. "Machine learning in acoustics: theory and applications". In: *Acoustical Society of America* (2019).

Middle ground

Traditional synthesis: structured building blocks based on frequency and/or symbolic representation. Fiddly parameters, imperfect recreations

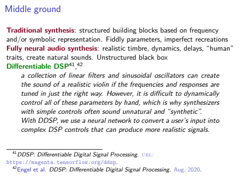
Fully neural audio synthesis: realistic timbre, dynamics, delays, “human” traits, create natural sounds. Unstructured black box

Differentiable DSP[41],[42]

> *a collection of linear filters and sinusoidal oscillators can create the sound of a realistic violin if the frequencies and responses are tuned in just the right way. However, it is difficult to dynamically control all of these parameters by hand, which is why synthesizers with simple controls often sound unnatural and “synthetic”.*
> *With DDSP, we use a neural network to convert a user’s input into complex DSP controls that can produce more realistic signals.*

[41] *DDSP: Differentiable Digital Signal Processing*. URL: https://magenta.tensorflow.org/ddsp.

[42] Engel et al. *DDSP: Differentiable Digital Signal Processing*. Aug. 2020.



Middle ground

Traditional synthesis: structured building blocks based on frequency and/or symbolic representation. Fiddly parameters, imperfect recreations
Fully neural audio synthesis: realistic timbre, dynamics, delays, “human” traits, create natural sounds. Unstructured black box
Differentiable DSP[41],[42]
a collection of linear filters and sinusoidal oscillators can create the sound of a realistic violin if the frequencies and responses are tuned in just the right way. However, it is difficult to dynamically control all of these parameters by hand, which is why synthesizers with simple controls often sound unnatural and “synthetic”.
With DDSP, we use a neural network to convert a user’s input into complex DSP controls that can produce more realistic signals.

[41] *DDSP: Differentiable Digital Signal Processing*. URL: https://magenta.tensorflow.org/ddsp.
[42] Engel et al. *DDSP: Differentiable Digital Signal Processing*. Aug. 2020.

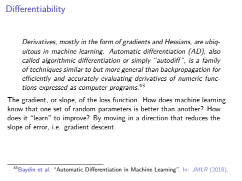
- Learn complex parameters to traditional DSP building blocks, rather than throwing out the baby with the bathwater.

Differentiability

> *Derivatives, mostly in the form of gradients and Hessians, are ubiquitous in machine learning. Automatic differentiation (AD), also called algorithmic differentiation or simply "autodiff", is a family of techniques similar to but more general than backpropagation for efficiently and accurately evaluating derivatives of numeric functions expressed as computer programs.*[43]

The gradient, or slope, of the loss function. How does machine learning know that one set of random parameters is better than another? How does it "learn" to improve? By moving in a direction that reduces the slope of error, i.e. gradient descent.

[43] Baydin et al. "Automatic Differentiation in Machine Learning". In: *JMLR* (2018).



Differentiability

Derivatives, mostly in the form of gradients and Hessians, are ubiquitous in machine learning. Automatic differentiation (AD), also called algorithmic differentiation or simply "autodiff", is a family of techniques similar to but more general than backpropagation for efficiently and accurately evaluating derivatives of numeric functions expressed as computer programs.[43]

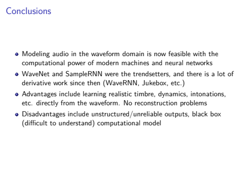
The gradient, or slope, of the loss function. How does machine learning know that one set of random parameters is better than another? How does it "learn" to improve? By moving in a direction that reduces the slope of error, i.e. gradient descent.

[43] Baydin et al. "Automatic Differentiation in Machine Learning". In: *JMLR* (2018).

- Differentiability is not new. Relevant in many forms of numerical computation e.g. fem. Just very generalized in neural networks
- "Differentiable" – machine learning is often done with gradients – gradient descent. The machine "learns" by reducing the error through the gradient. As DSP building blocks are differentiable, they can thus be machine learned (or more accurately, parameters can be estimated, and from the differentiability, the resulting quality improvement can be measured e.g. compared to a real violin sound).

Conclusions

- Modeling audio in the waveform domain is now feasible with the computational power of modern machines and neural networks
- WaveNet and SampleRNN were the trendsetters, and there is a lot of derivative work since then (WaveRNN, Jukebox, etc.)
- Advantages include learning realistic timbre, dynamics, intonations, etc. directly from the waveform. No reconstruction problems
- Disadvantages include unstructured/unreliable outputs, black box (difficult to understand) computational model



Conclusions

- Modeling audio in the waveform domain is now feasible with the computational power of modern machines and neural networks
- WaveNet and SampleRNN were the trendsetters, and there is a lot of derivative work since then (WaveRNN, Jukebox, etc.)
- Advantages include learning realistic timbre, dynamics, intonations, etc. directly from the waveform. No reconstruction problems
- Disadvantages include unstructured/unreliable outputs, black box (difficult to understand) computational model

- It’s taken me 3 weeks of training so far to get even really bad outputs. Computational power is a real limitation
- We can’t compete with BigCos and their own special sauce (computation, storage, in-house expertise)