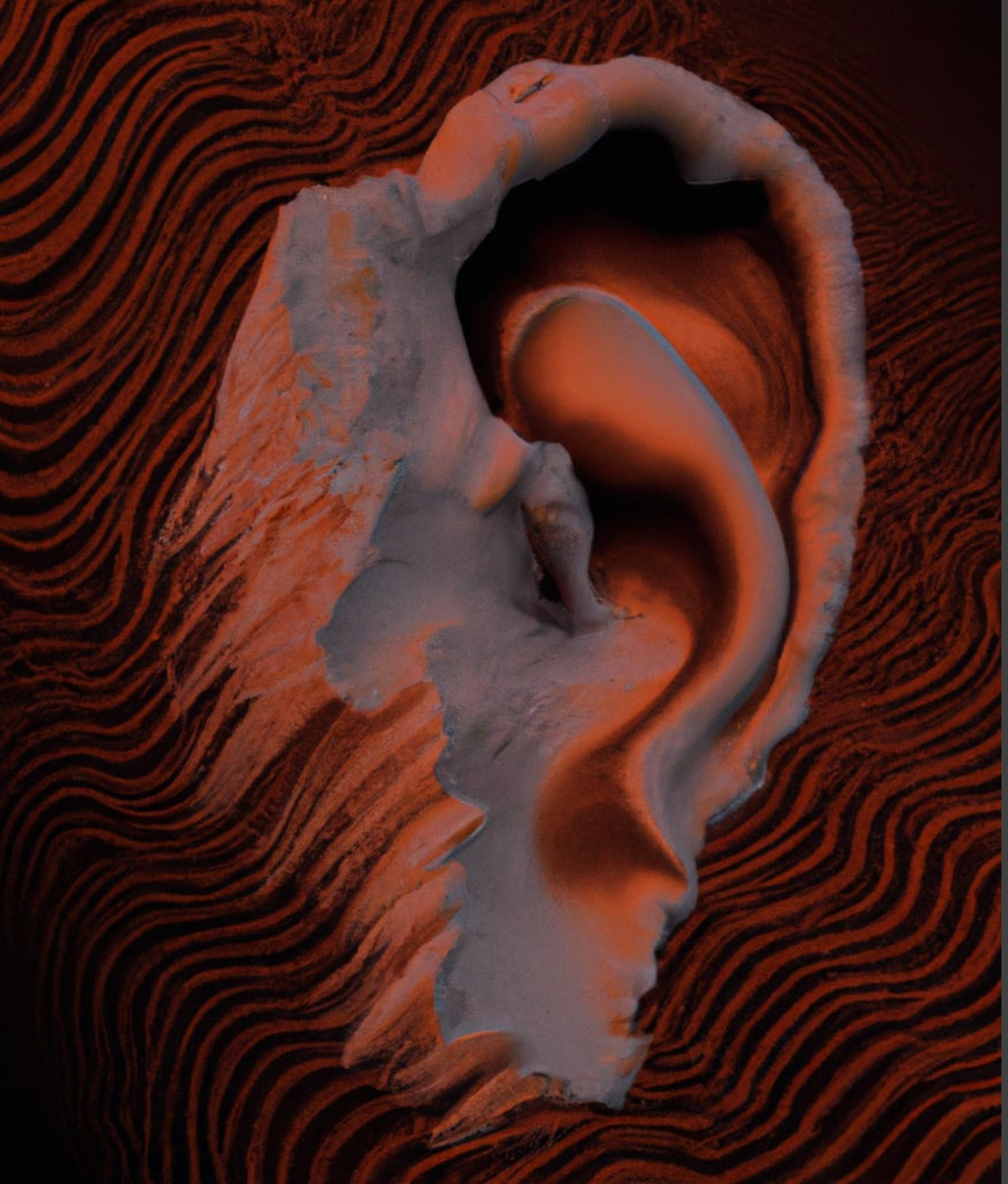


Audio and 1D Signals

Applied Machine Learning
Derek Hoiem

This class: Audio and 1D signals

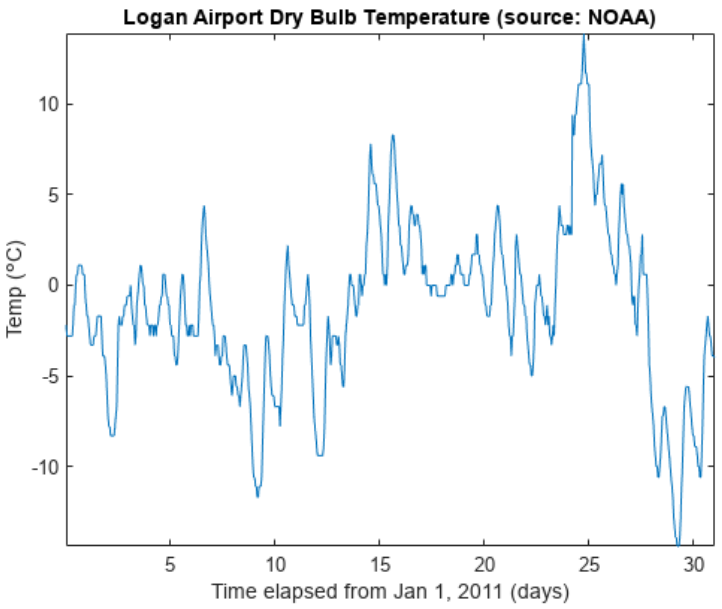
- Representing sound with frequencies
- Deep networks for audio
- Other 1D time series

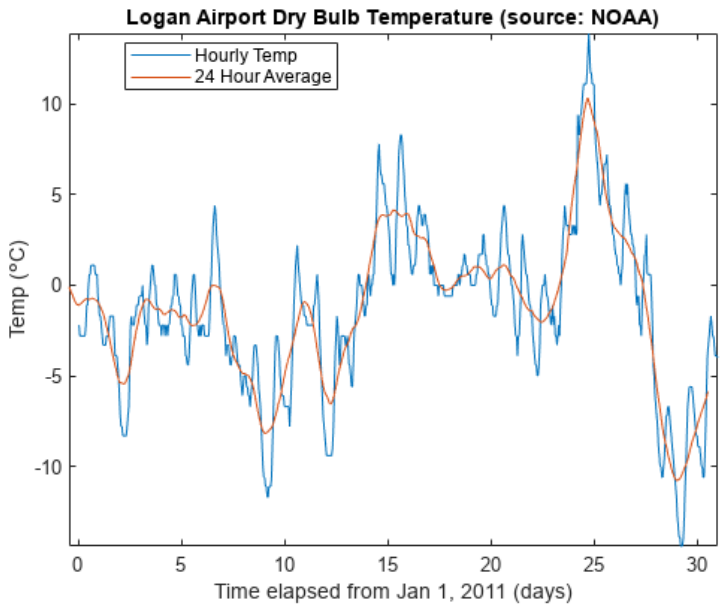
1D time series problems are common

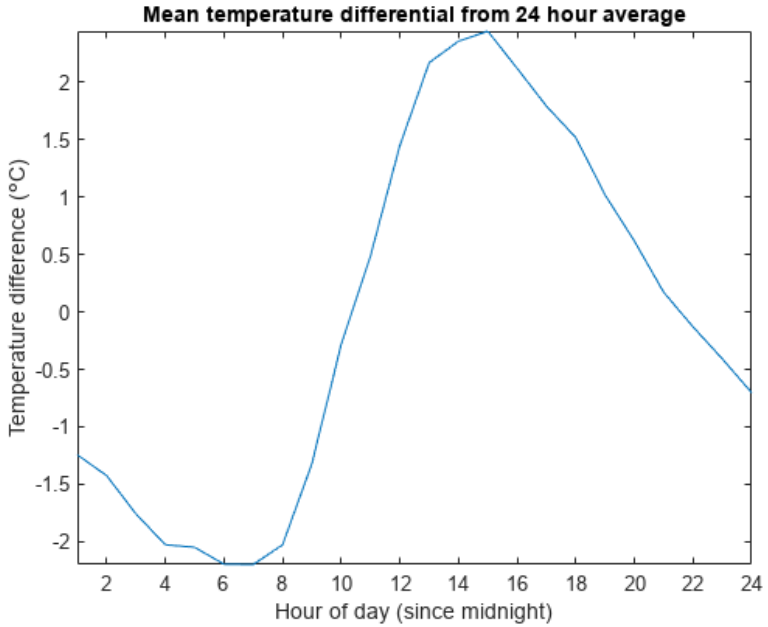
- Audio
  - Speech recognition, sound classification, source separation
- Stock prices, vibrations, popularity trends, temperature, …

Example: Temperature at Logan Airport

- Trends can occur at multiple time scales
- Smoothing and differencing are important forms of analysis







https://www.mathworks.com/help/signal/ug/signal-smoothing.html

How are 1D signals different from other problems?

1. Signals are often **periodic**, and can be analyzed in terms of slow and fast moving changes

2. Signals can have **arbitrary length**, and can be analyzed in terms of recent data points (windows) or some cumulative representation

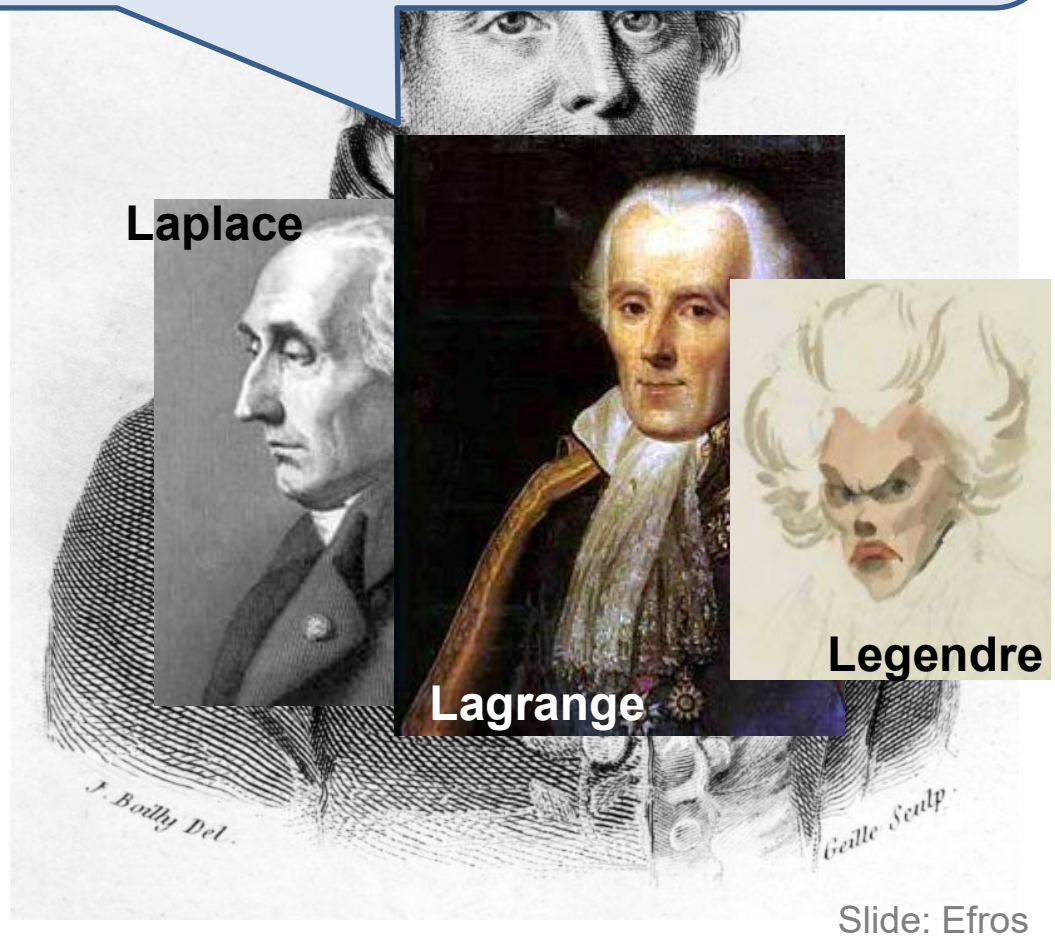
Thinking in terms of frequency

Jean Baptiste Joseph Fourier had a crazy idea in 1807

Any *univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies.*

- Don't believe it?
  - Neither did Lagrange, Laplace, Poisson and other big wigs
  - Not translated into English until 1878!
- But it's (mostly) true!
  - called Fourier Series
  - there are some subtle restrictions

...the manner in which the author arrives at these equations is not exempt of difficulties and...his analysis to integrate them still leaves something to be desired on the score of generality and even rigour.



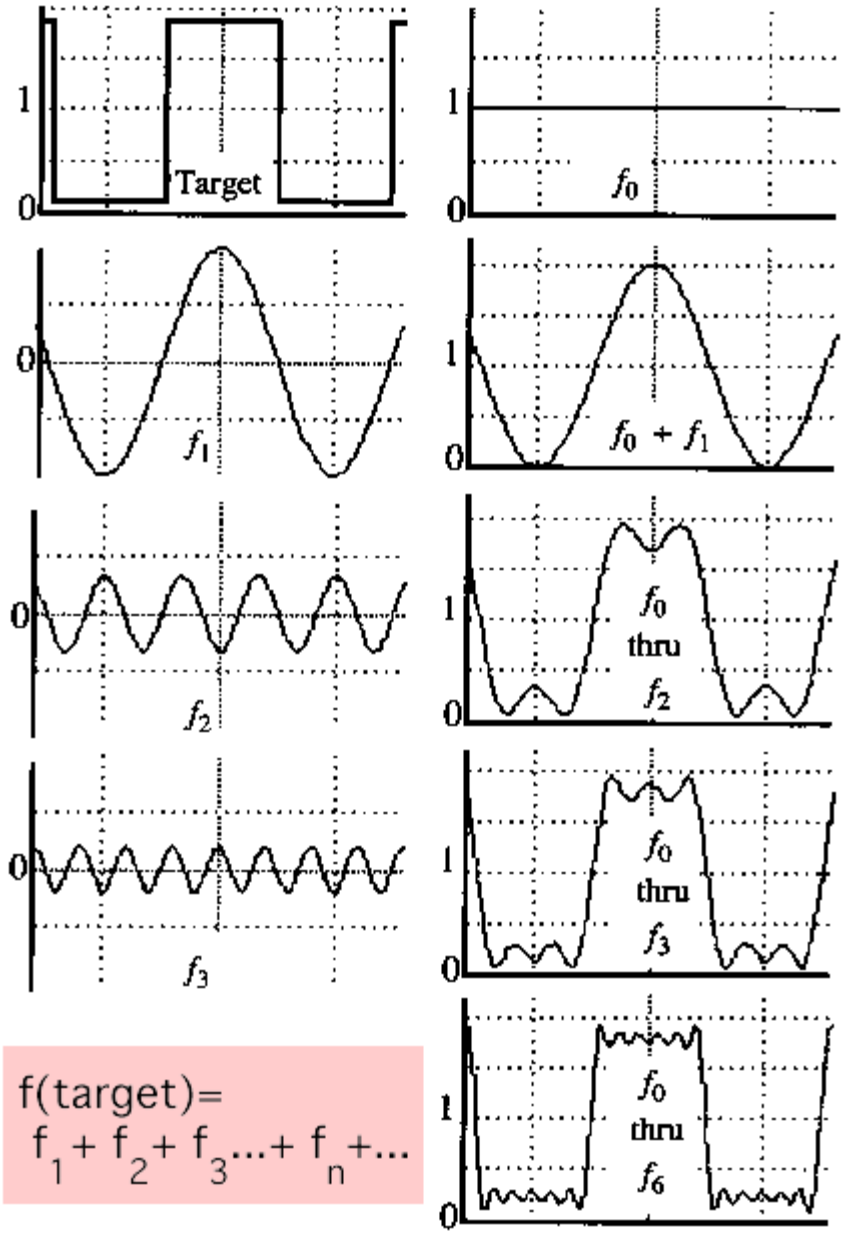
Slide: Efros

A sum of sines

Our building block:

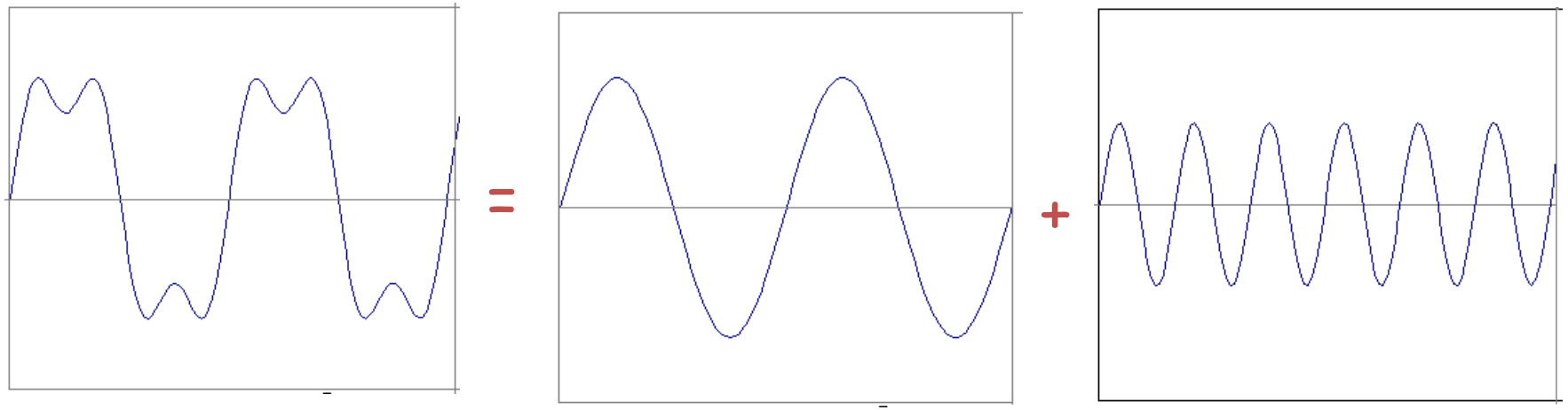
$$A\sin(\omega x + \phi)$$

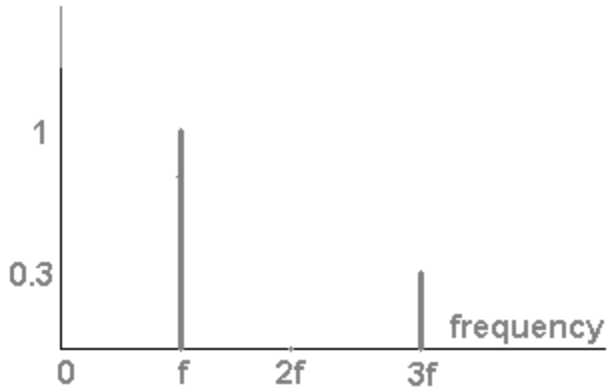
Add enough of them to get any signal *f(x)* you want!



Frequency Spectra

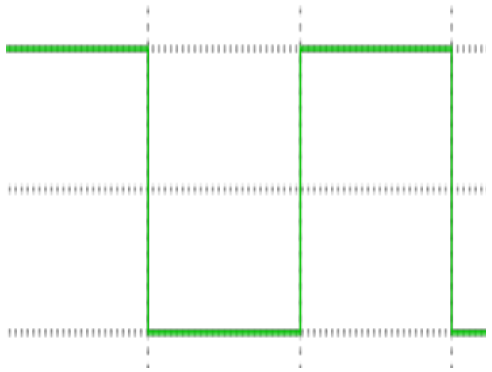
- example : $g(t) = \sin(2\pi f\, t) + (1/3)\sin(2\pi(3f)\, t)$



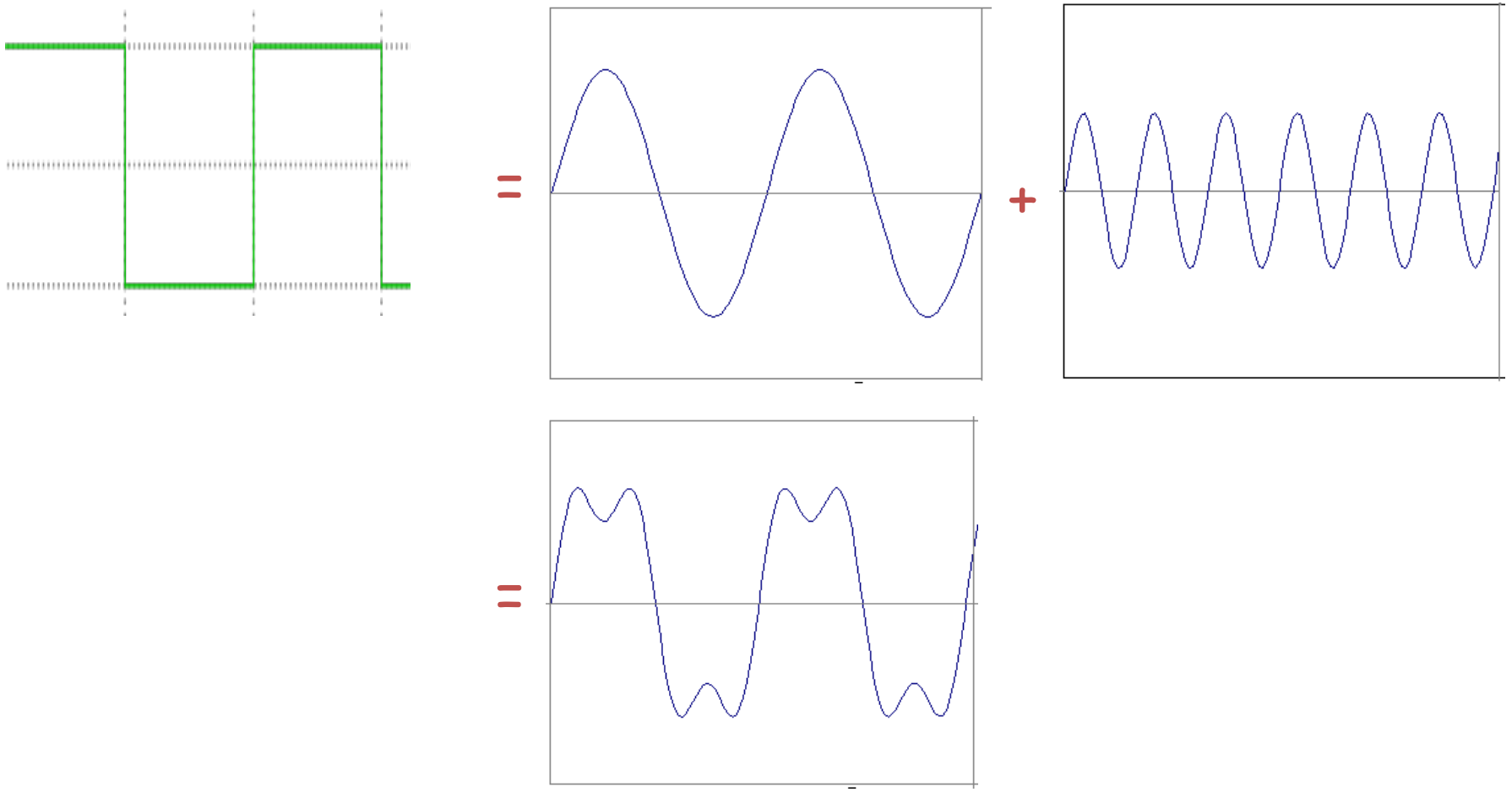


Slides: Efros

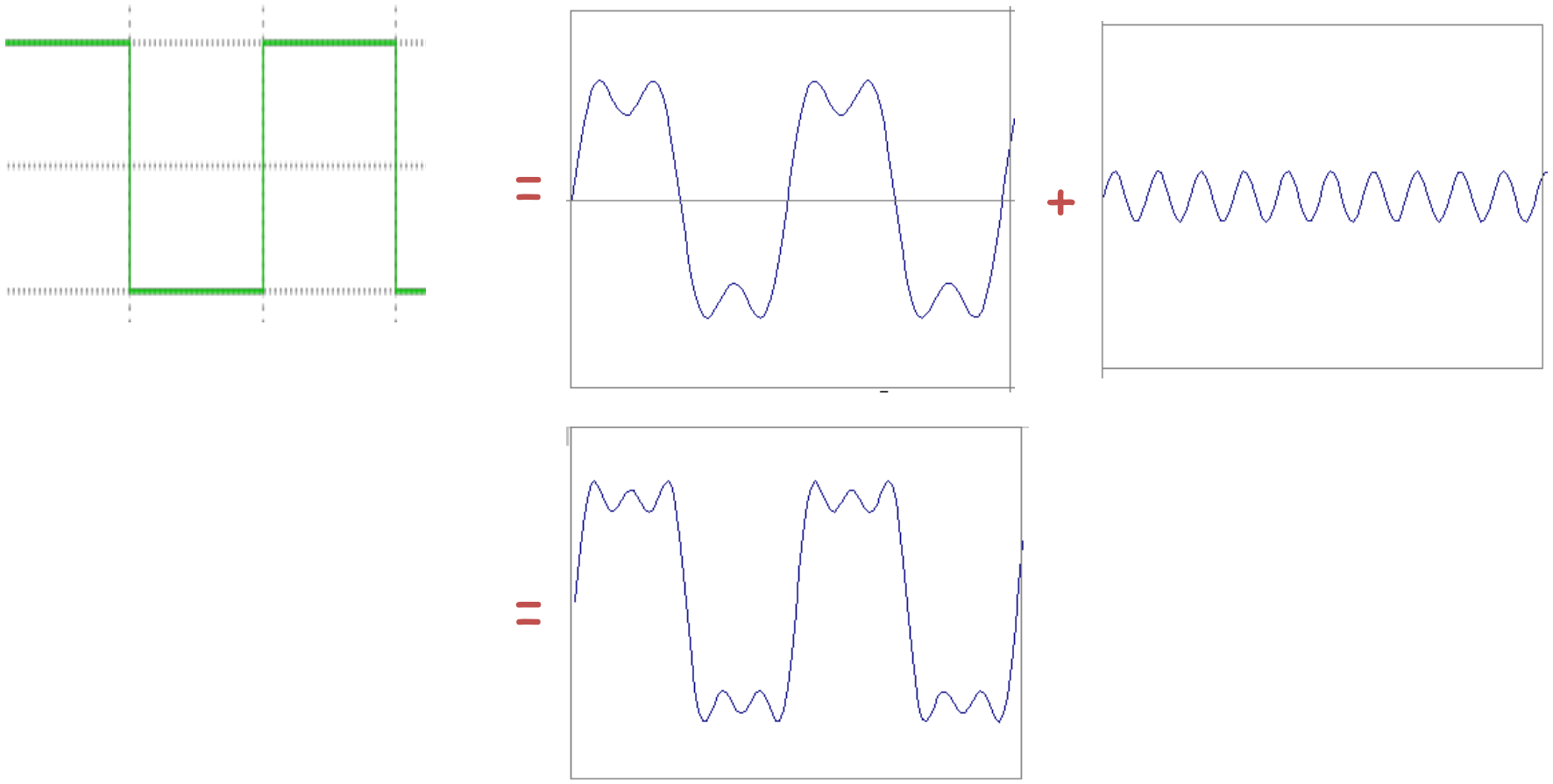
Frequency Spectra



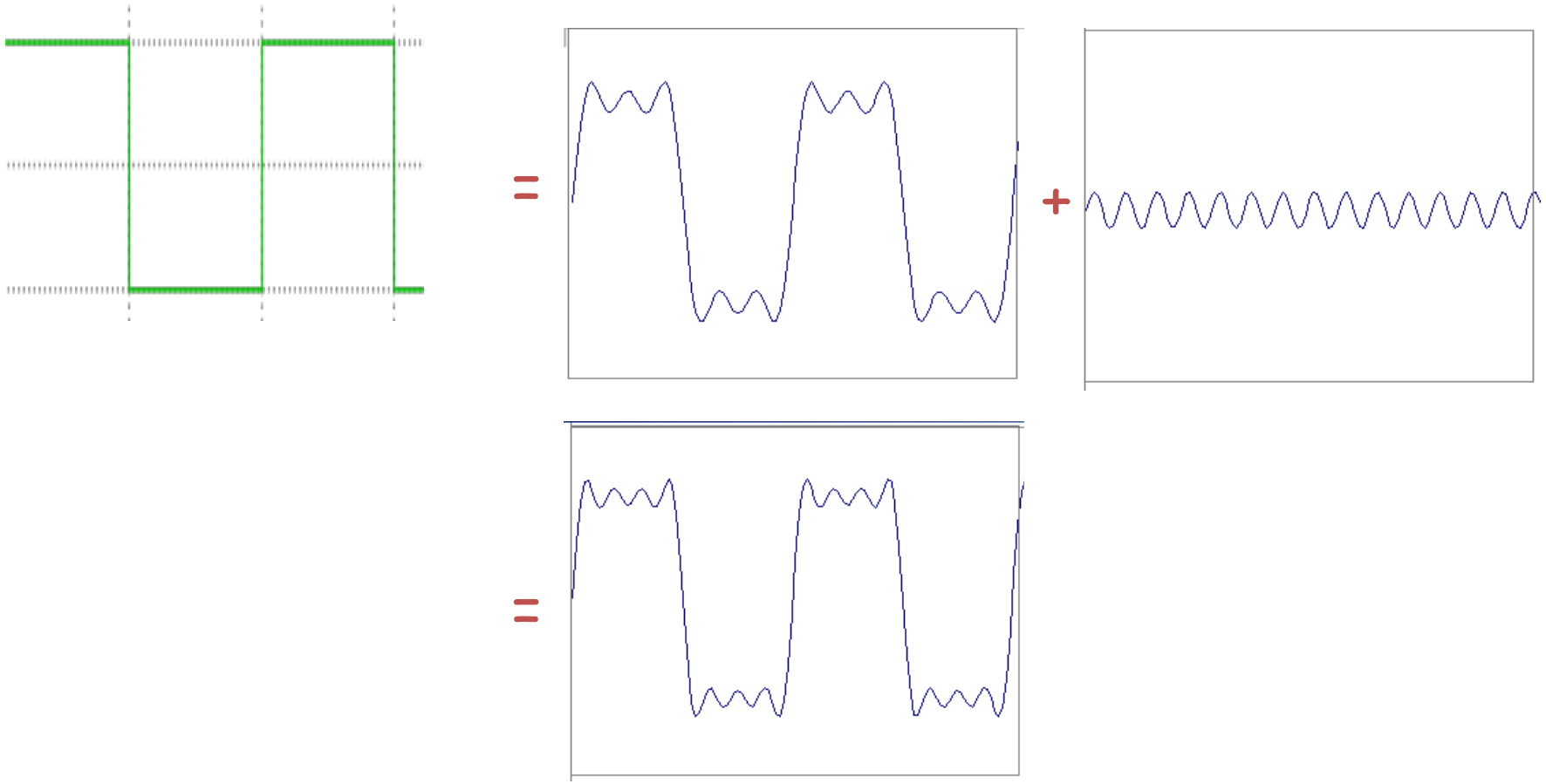
Frequency Spectra



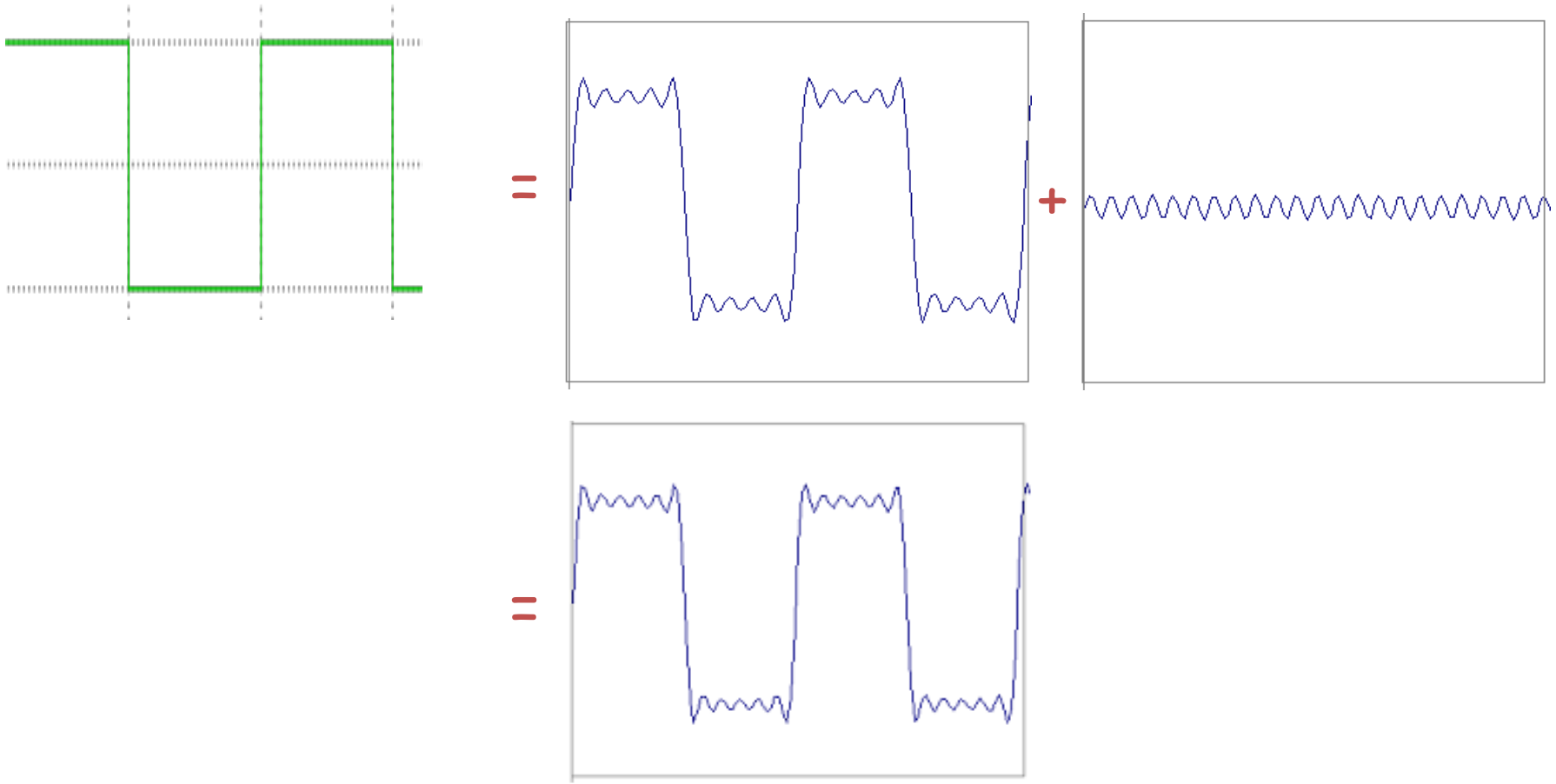
Frequency Spectra



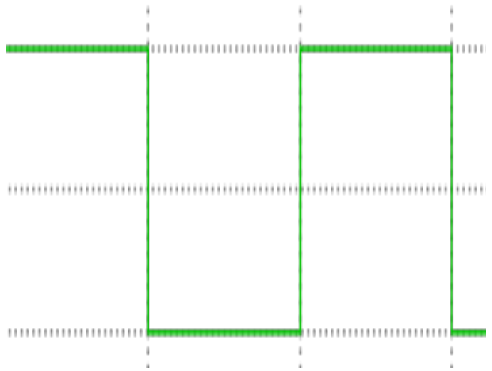
Frequency Spectra

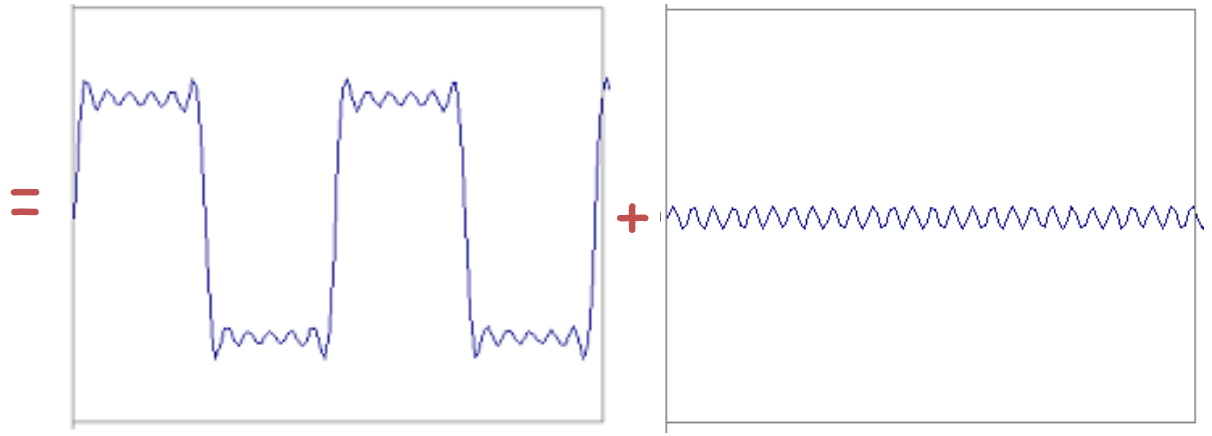


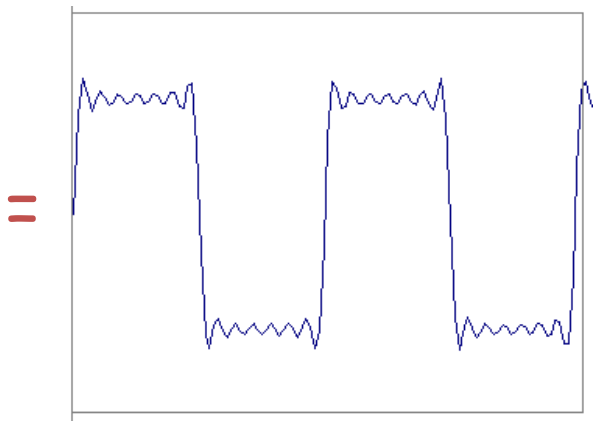
Frequency Spectra



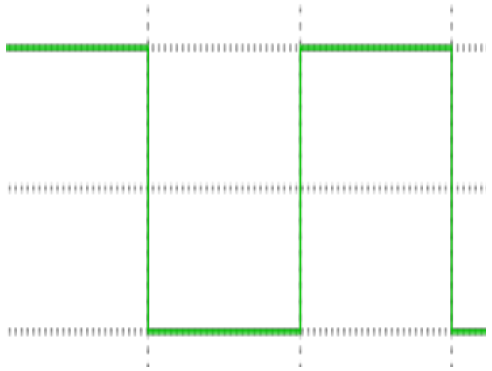
Frequency Spectra



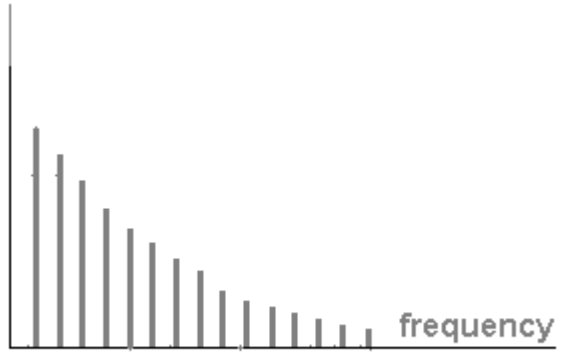




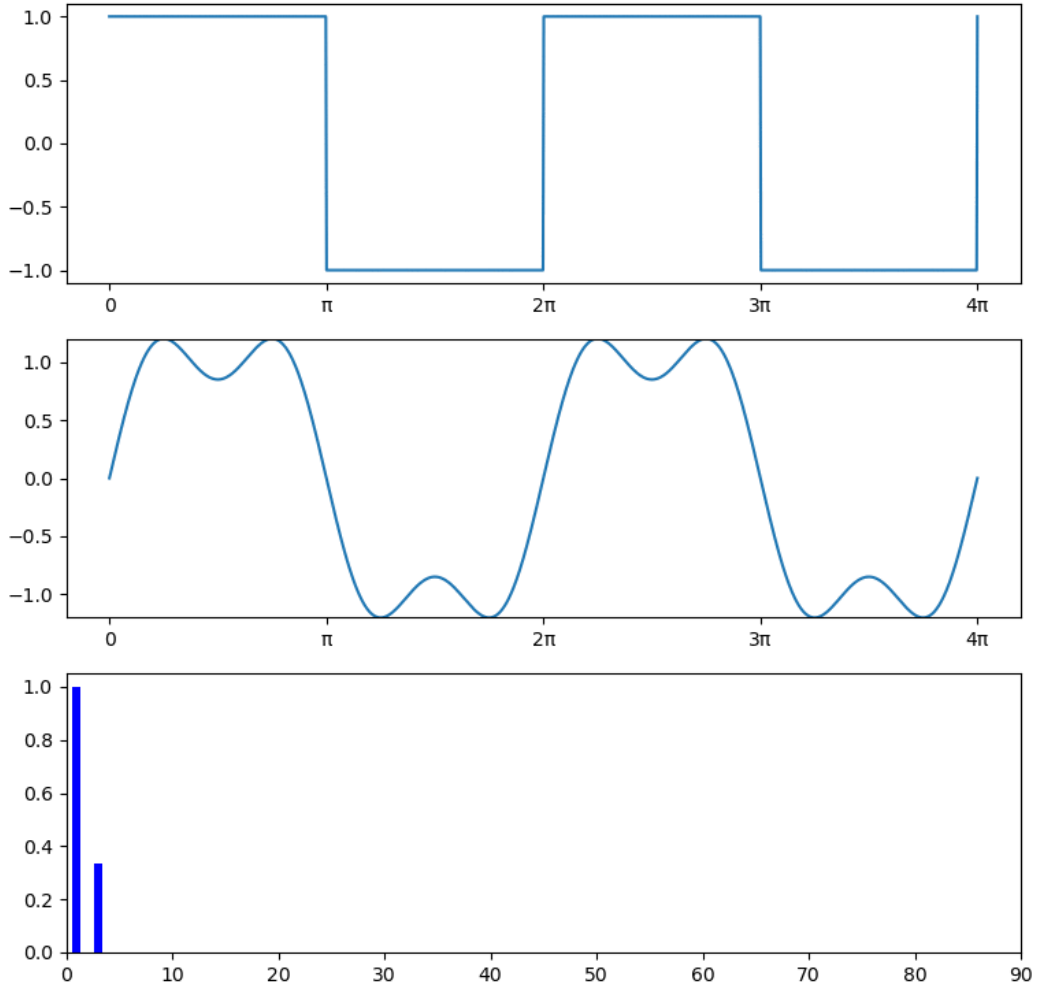
Frequency Spectra



$$= A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kt)$$



Frequency Spectra



Sound

- Sound is composed of overlaid sinusoidal functions with varying frequency and amplitude
- Digital sound is recorded by sampling the sound signal with high sampling frequency

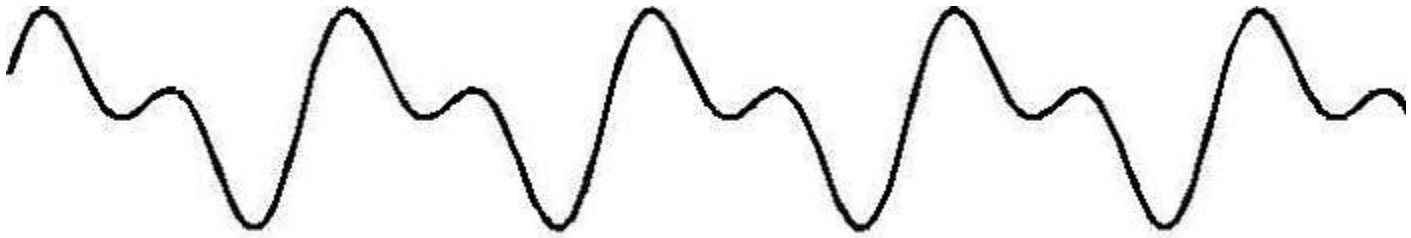


Fig src

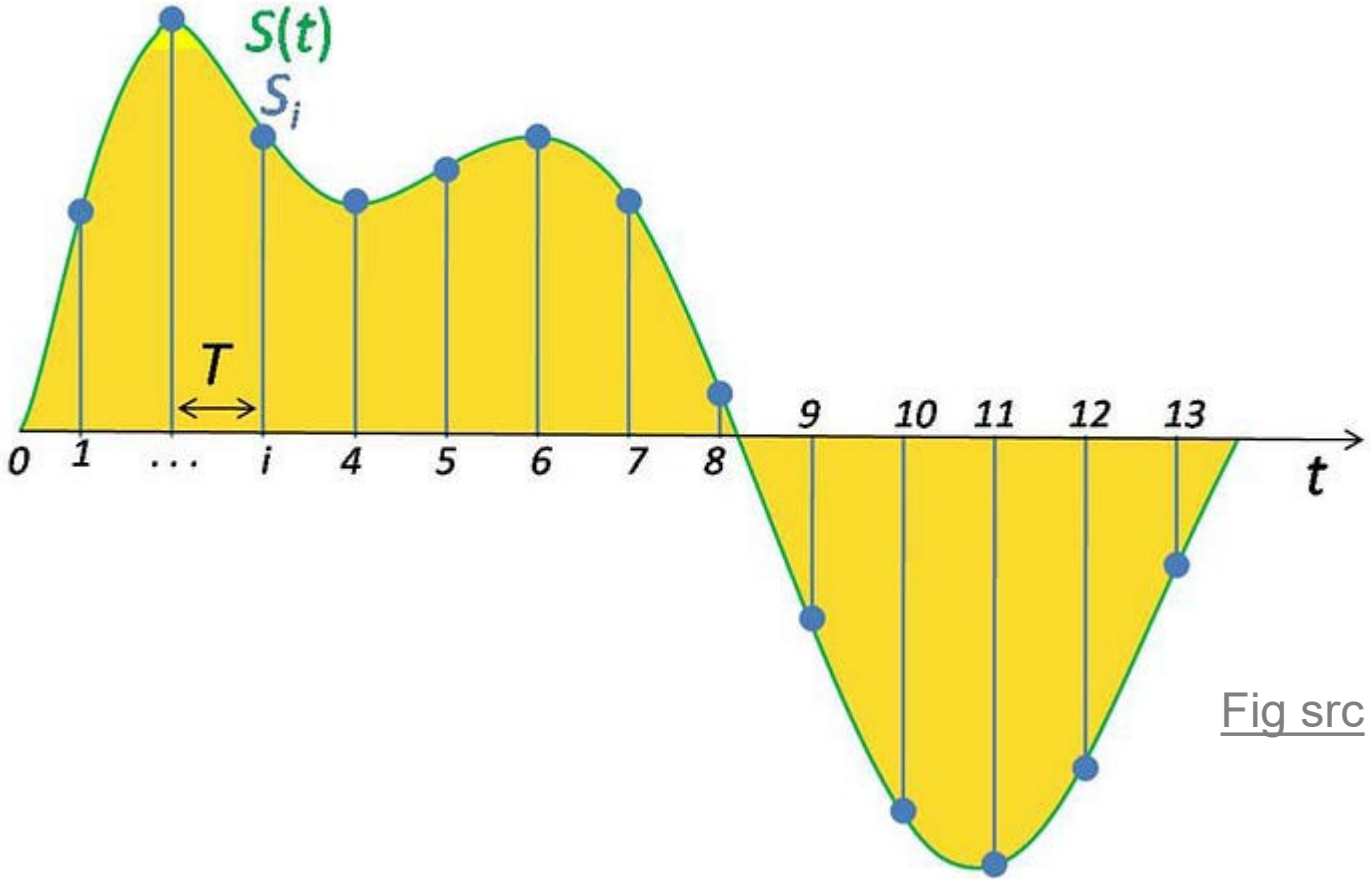


Fig src

Slide content source: TDS

Spectrogram

- Sound is a series of sinusoids with varying amplitudes, frequencies, and phases

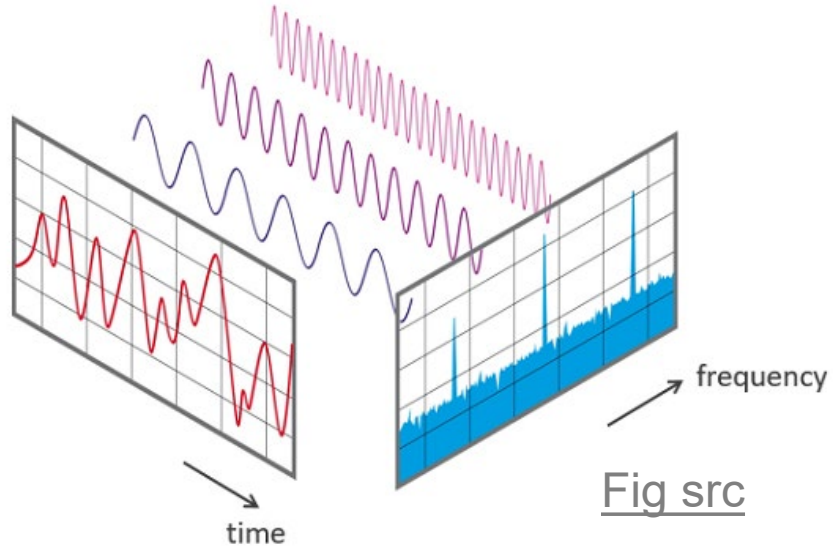
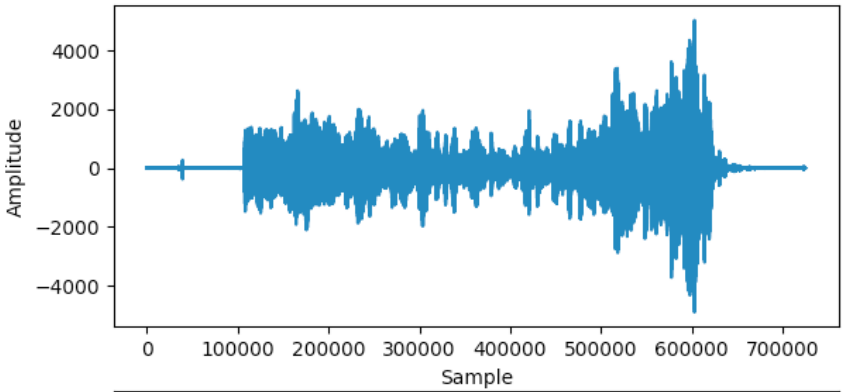
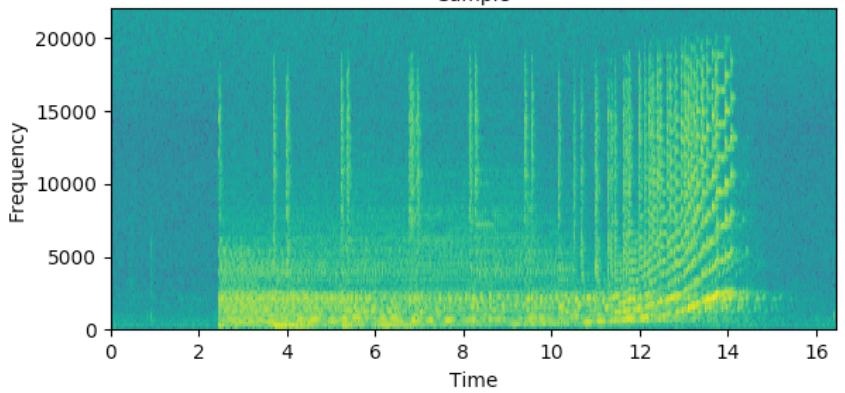


Fig src

- Total amplitude tells us how loud the sound is at some point, but that's not very informative



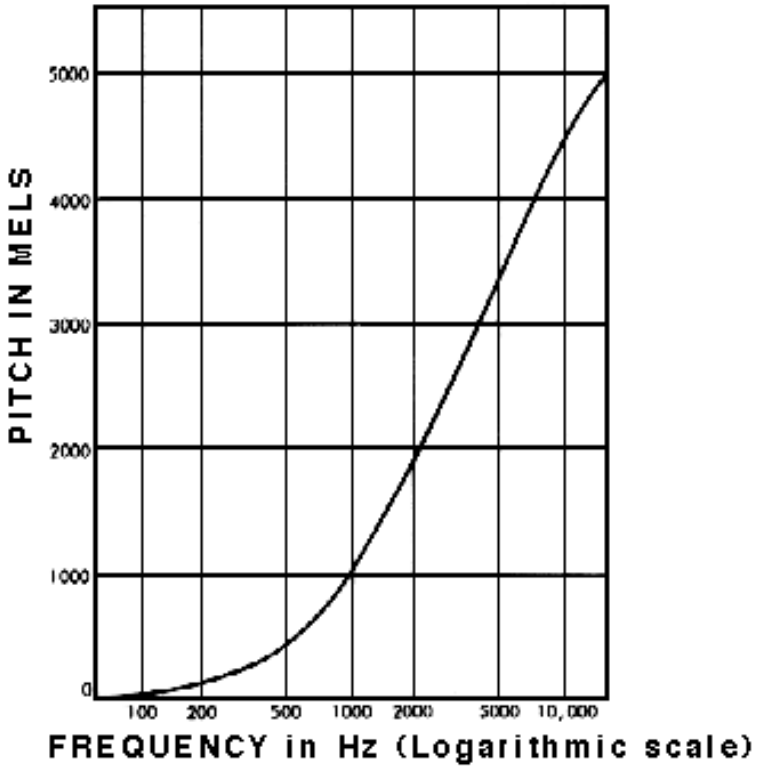
- The spectrogram tells us the power in each frequency over some time window



Slide content source: TDS

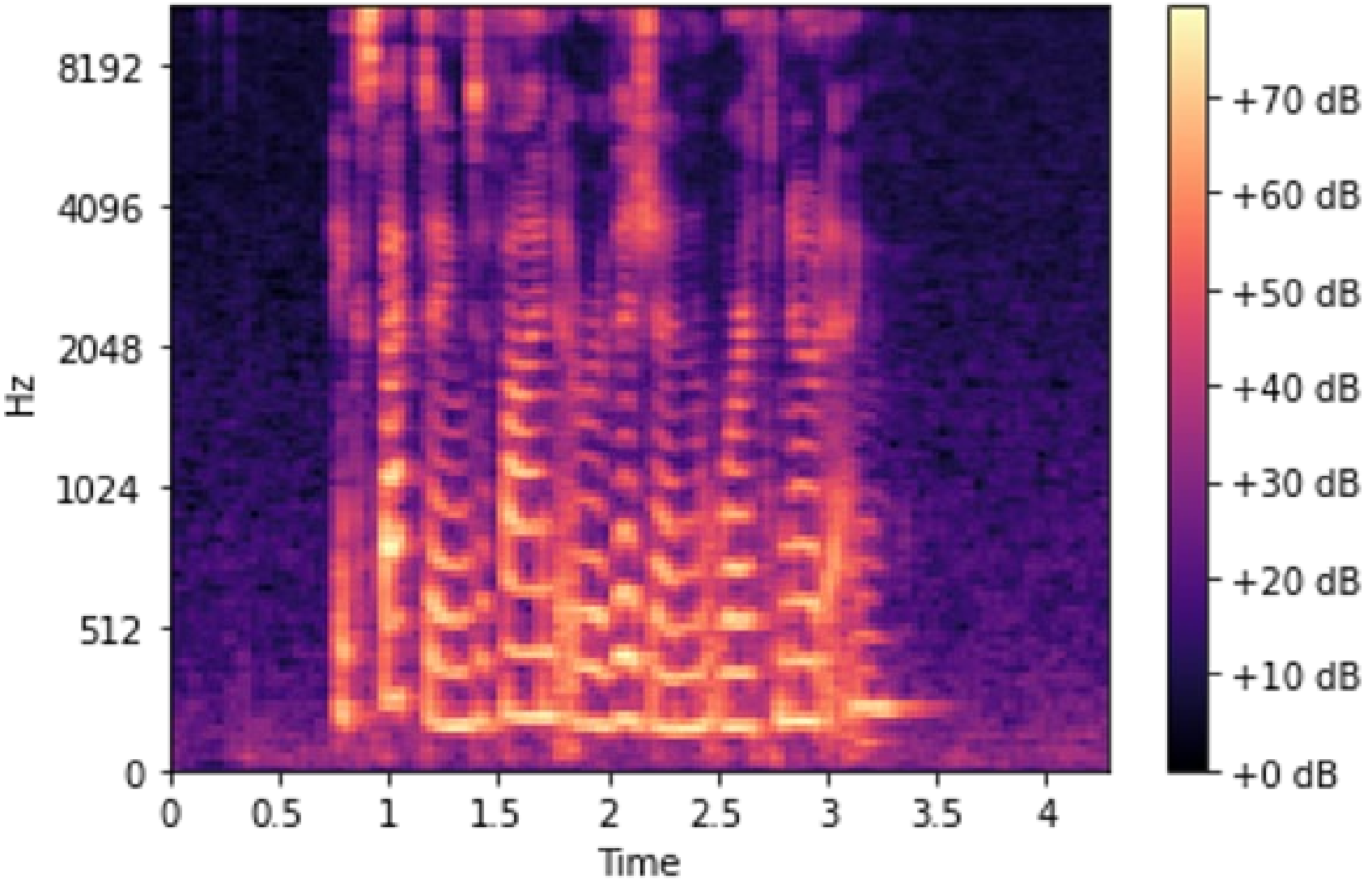
Mel Spectrograms

- Humans perceive frequency on a logarithmic scale, e.g. each octave in music doubles the frequency
- Mel scale maps frequency to human perception of pitch
- Mel scale records amplitude in decibels (logarithmic base 10)



Slide content source: TDS

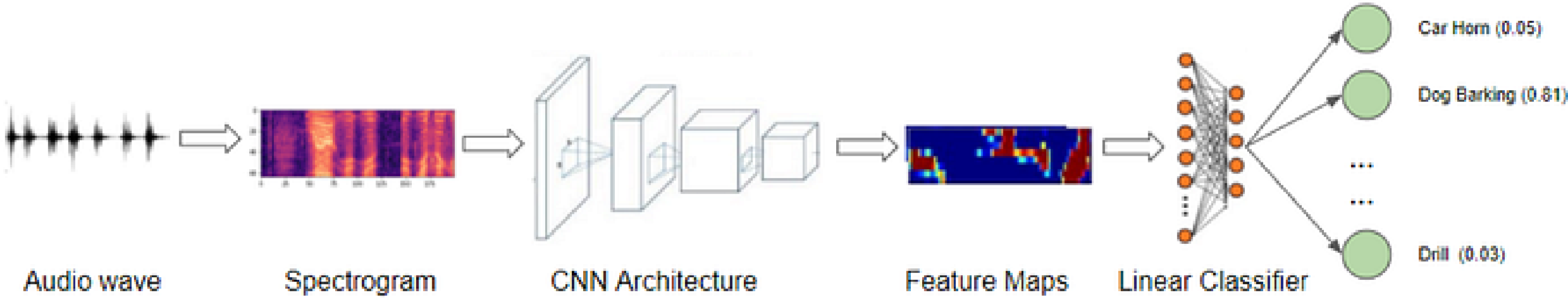
Mel Spectrogram



Slide content source: TDS See this link for code to generate Mel Spectrogram

Audio Classification

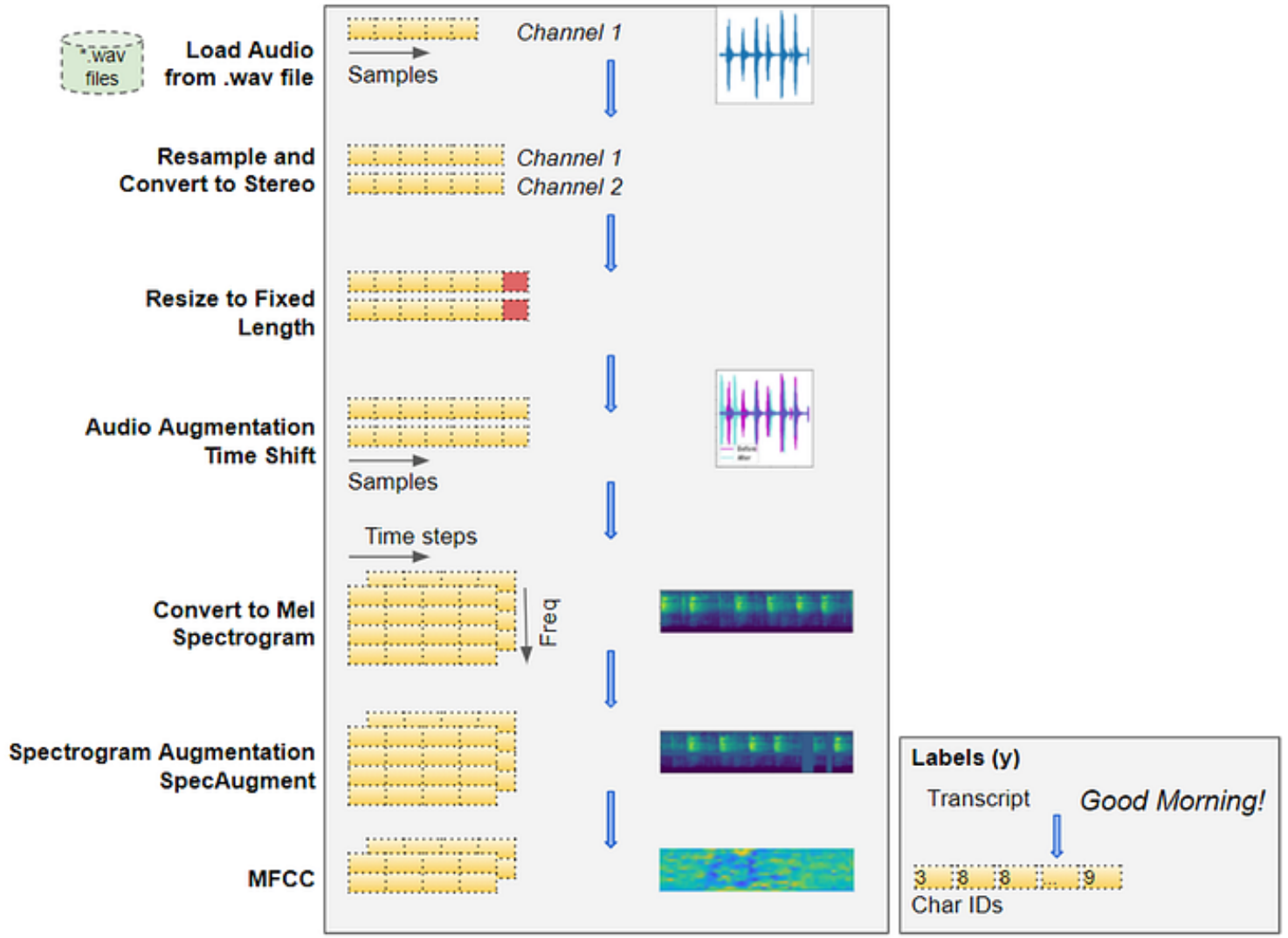
1. Pre-process into Mel Spectogram Image
2. Apply vision-based architectures to classify
   - Data augmentation can include time shift on audio wave and time/frequency masking on spectrogram



Details and code here: https://towardsdatascience.com/audio-deep-learning-made-simple-sound-classification-step-by-step-cebc936bbe5

Audio to Speech (ASR)

- Process data, similar to audio classification
- MFCC processes Mel Spectrogram with DCT to get a compressed representation that focuses on speech-related frequencies

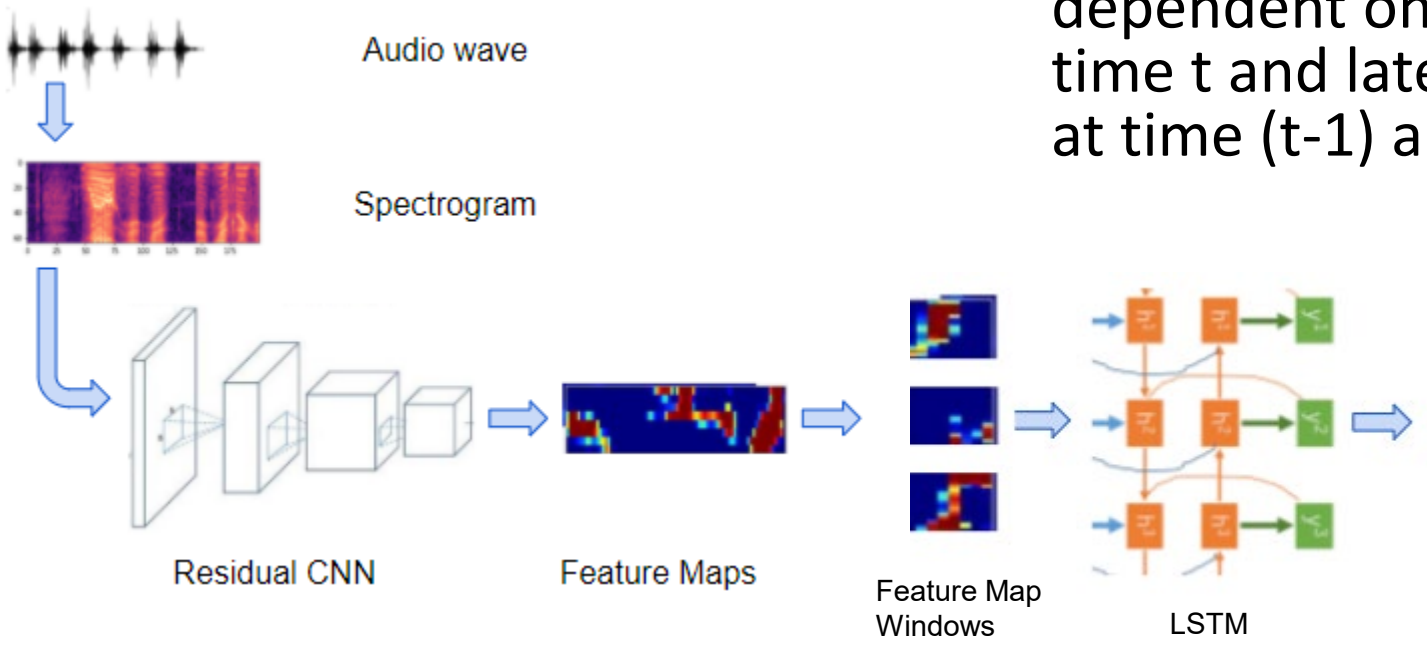


Content src

ASR

Deep network with vision architecture extracts features

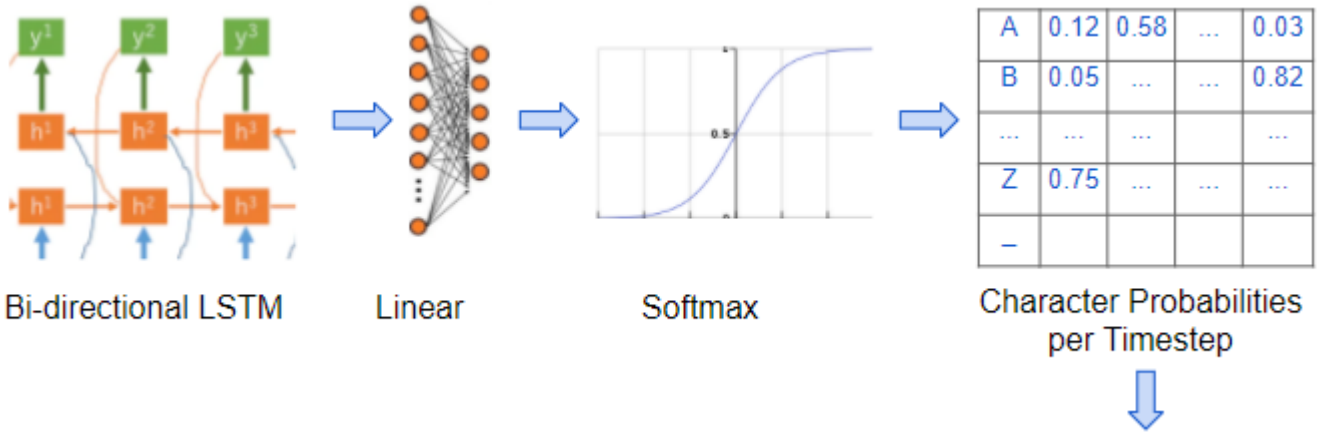
Recurrent network (e.g. LSTM) models each output at time t as dependent on the features at time t and latent representations at time (t-1) and/or (t+1)



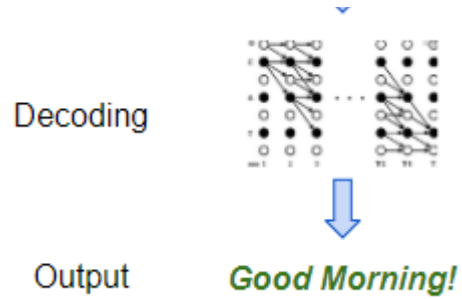
Based on Baidu's Deep Speech model, as described here

ASR

- Each time step predicts a probability for each character



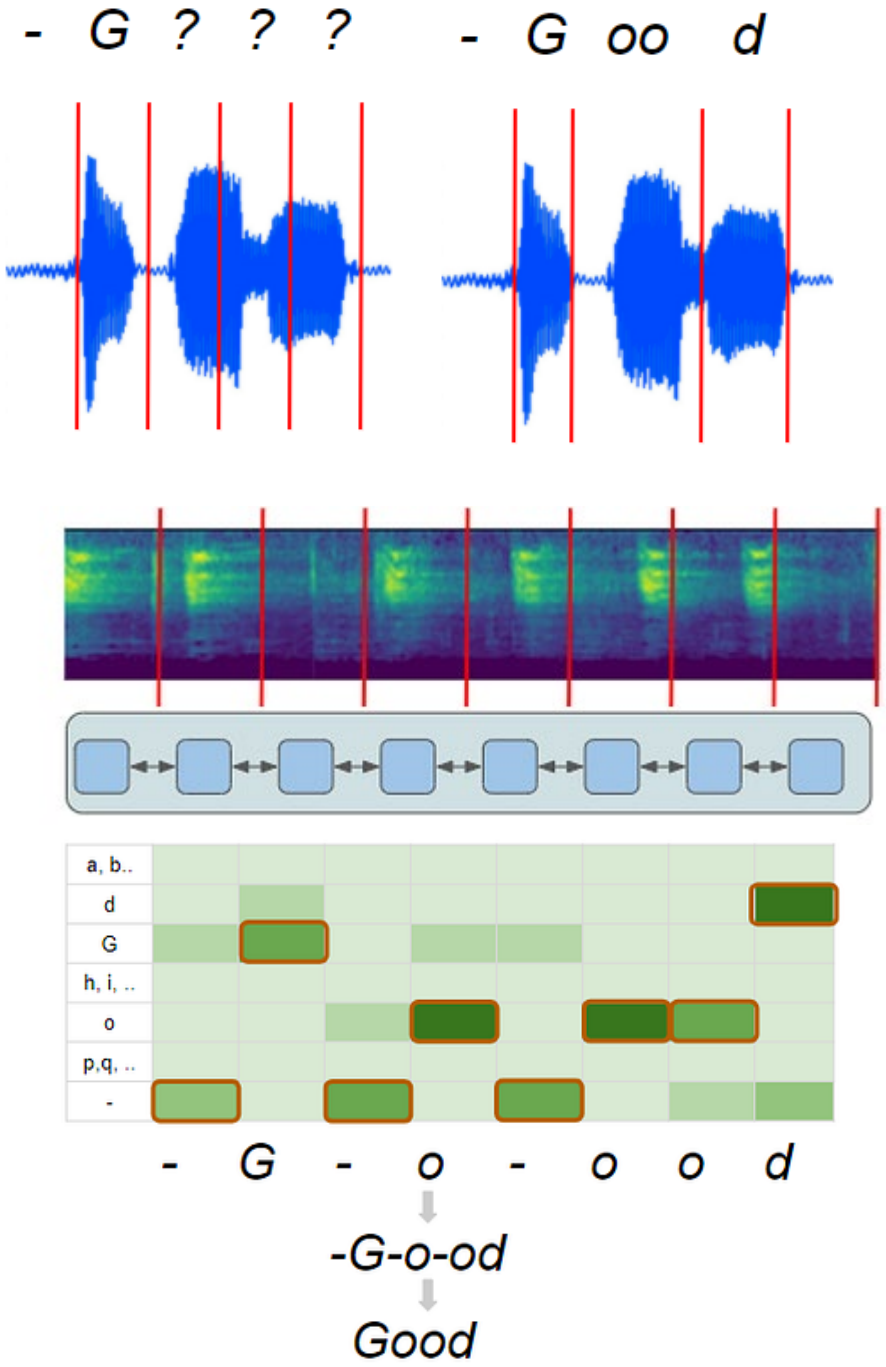
- Character probabilities are decoded into text output



Based on Baidu's Deep Speech model, as described here

ASR

- A challenge of ASR is temporal spacing between characters
- Audio is sliced uniformly and fed into RNN
- RNN predicts character probabilities, merges repeated characters, and removes blanks

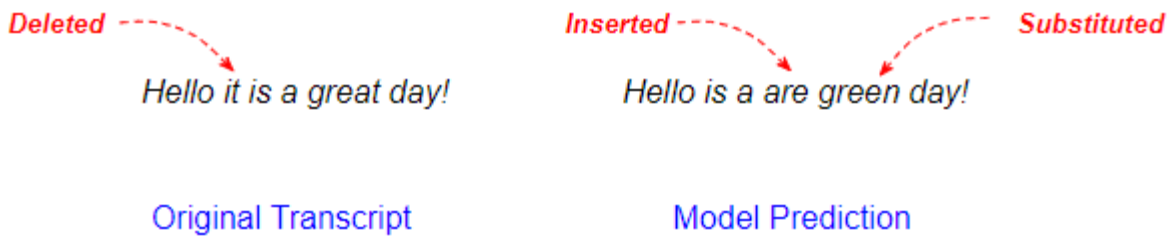


Based on Baidu's Deep Speech model, as described here

Other details

- Loss based on likelihood of true character sequence

- Error is often reported as Word Error Rate or Character Error Rate



- A language model and/or beam search can be used to improve output

As described here

Popular libraries for audio processing

- Librosa: https://librosa.org/doc/latest/index.html
- Torch Audio: https://pytorch.org/audio/stable/index.html
- Others: https://wiki.python.org/moin/Audio/
- HuggingFace Models: https://huggingface.co/docs/transformers/tasks/audio_classification

And now, this…

GPT-4: Write lyrics for a short, catchy Irish tune about the best deep learning techniques….
Suno: A lively, Irish folk tune with a cheery rhythm
https://suno.com/song/8d22437b-579a-410d-b3cc-730b55ee5f07

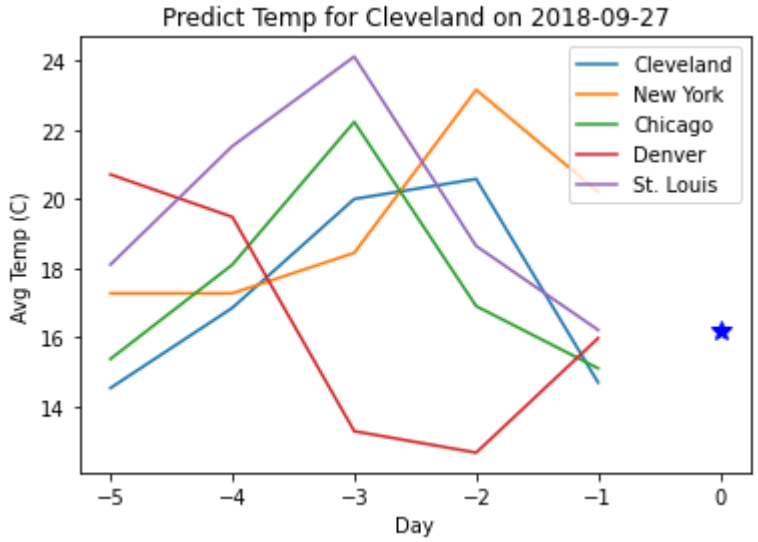
GPT-4 + Suno: Power up the code!
https://suno.com/song/3bea1938-bbcf-4c98-aff6-99bed9711f8b

Smarter Better Faster (my parody lyrics)
https://suno.com/song/eb2a33d3-aa01-466f-8d4b-a8ebcb49ced4

It’s my loss (my lyrics)
https://suno.com/song/2cdf6a66-7a1a-4b2f-afff-507d8554b610

1D Time Series more generally

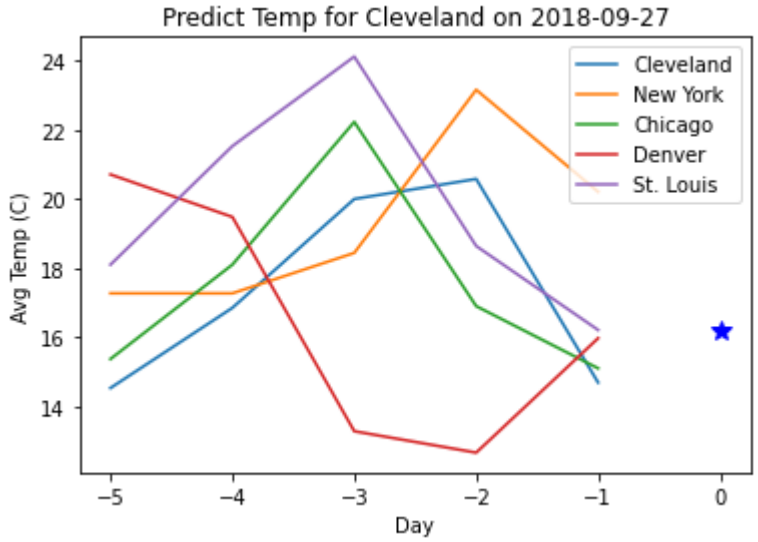
Recall HW 1: predict temperature in Cleveland based on temperatures from US cities in previous days



What did we do to handle this time series?

1D Time Series more generally

Recall HW 1: predict temperature in Cleveland based on temperatures from US cities in previous days

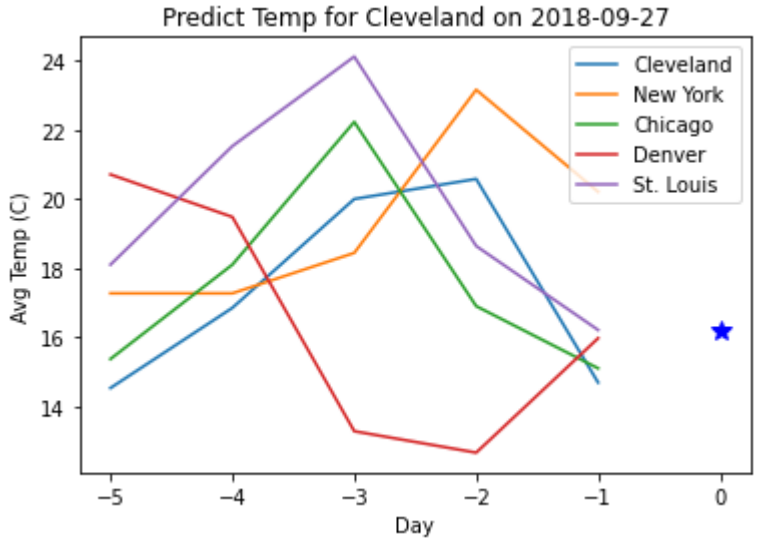


What did we do to handle this time series?

1. Windowed prediction: Consider temperature given five preceding days
2. Various prediction methods
    1. Linear regression
    2. Nearest neighbor
    3. Random forest

1D Time Series more generally

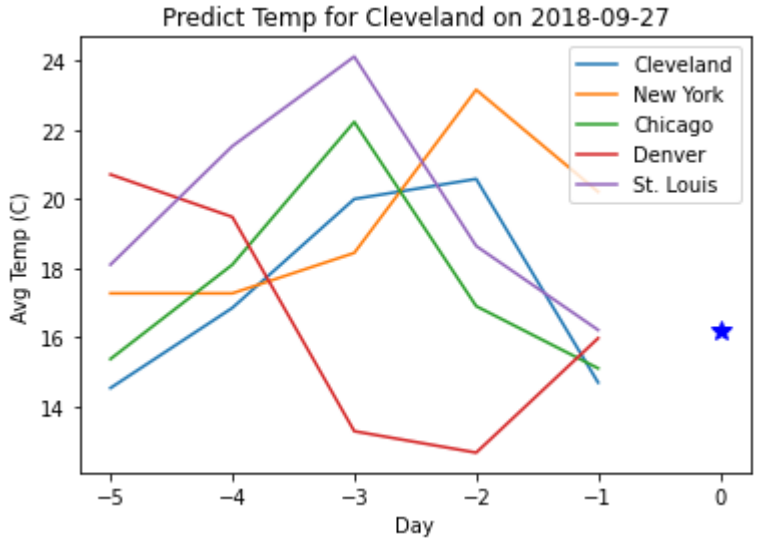
Recall HW 1: predict temperature in Cleveland based on temperatures from US cities in previous days



What could we do to improve prediction?

1D Time Series more generally

Recall HW 1: predict temperature in Cleveland based on temperatures from US cities in previous days



What could we do to improve prediction?

1. Take into account other features: time of year, other atmospheric effects
2. Learn better representations, e.g. use multitask learning to regress difference of temperature for each city from previous day in a deep neural net

1D Series more generally

- Common to apply 1-D filter operations to smooth (e.g. 1D Gaussian) or highlight differences (e.g. difference filter [-1 1 0])
- Signal is often converted into fixed length by windowing and/or padding
- Predictions by dynamic models can also be used as features, e.g. accounting for position, velocity, and acceleration
- Continuity of features can be achieved using convolutional networks, LSTMs, or other recurrent networks

Deep networks for time series forecasting

- The success of deep networks in vision, language, and audio is due to ability to train pre-trained models on large corpora and transfer representations to target tasks

- But time series problems seem quite different, e.g. predicting the price of commodities, the temperature, or the favorability polling of a political candidate

- Can we pre-train a model that adapts well to target time series forecasting tasks?

Zero-shot time-series forecasting, N-BEATS

- Train a network on diverse time series forecasting tasks
  - 100K+ time series sequences
  - Predict next value given preceding values
- Apply it zero shot to new time series forecasting tasks

https://arxiv.org/abs/2002.02887
https://arxiv.org/abs/1905.10437

Zero-shot time-series forecasting, N-BEATS

- Input is preceding window of time series data
- Each block updates the forecast based on unexplained data
  1. Receives delta of input and “backcast”
  2. Applies MLP
  3. Outputs “backcast”, the estimate of input signal
  4. Outputs delta to forecast
- Residual connections between blocks and for each stack allow training gradients to flow throughout the network
- Output is total forecast

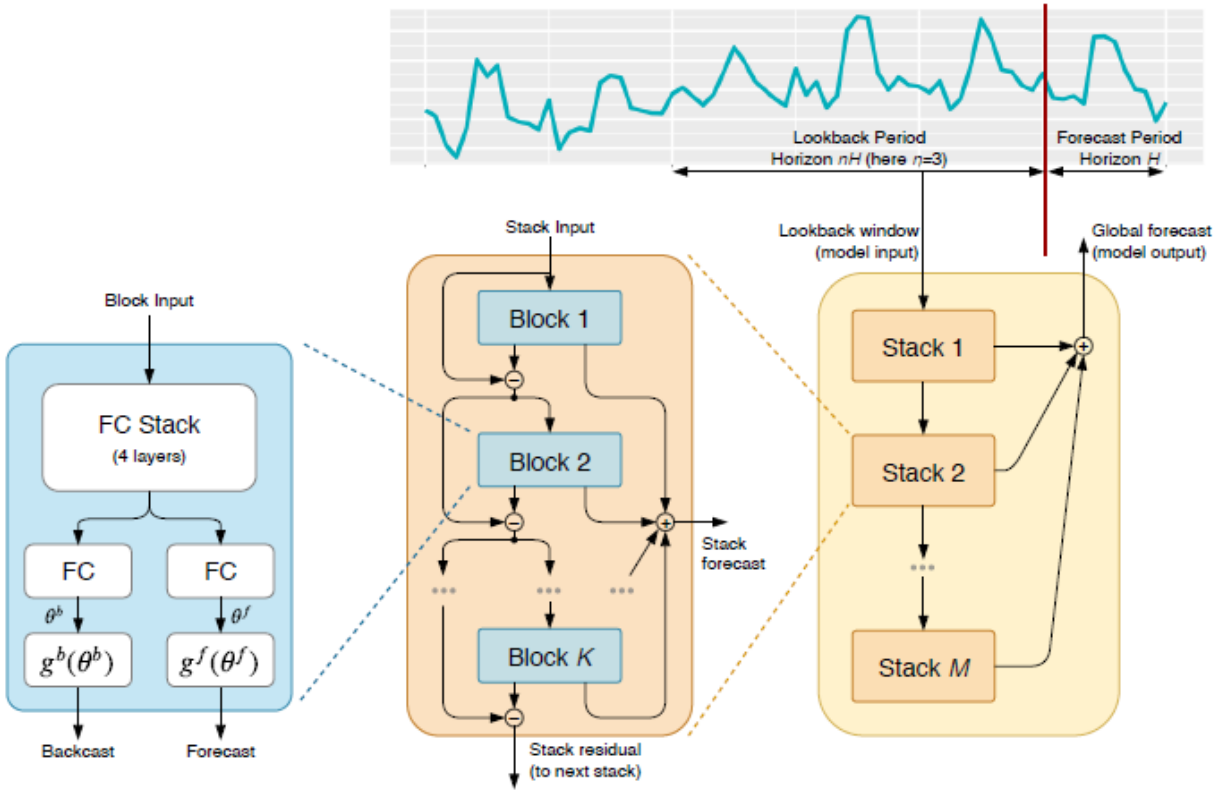
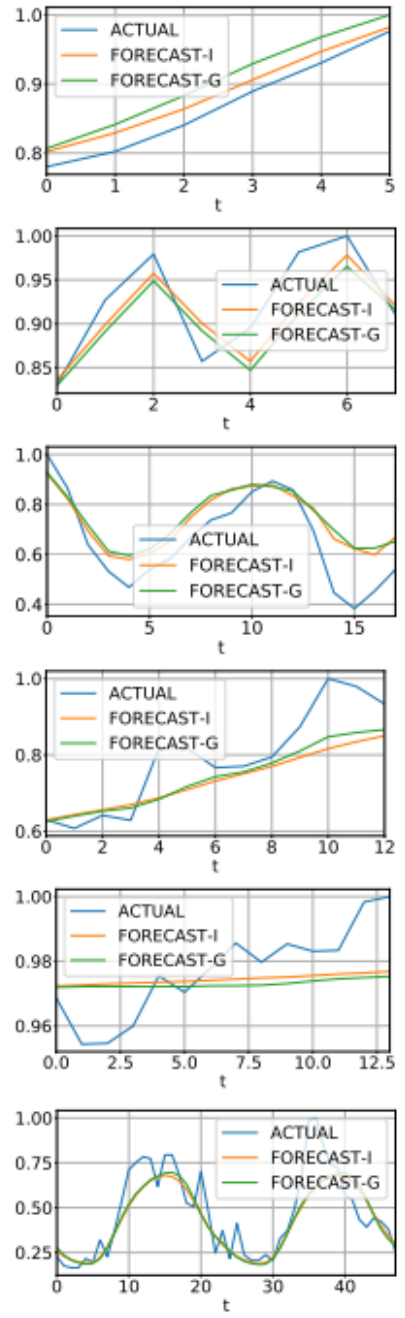


Figure 1: Proposed architecture. The basic building block is a multi-layer FC network with RELU nonlinearities. It predicts basis expansion coefficients both forward, θ^f, (forecast) and backward, θ^b, (backcast). Blocks are organized into stacks using doubly residual stacking principle. A stack may have layers with shared g^b and g^f. Forecasts are aggregated in hierarchical fashion. This enables building a very deep neural network with interpretable outputs.

https://arxiv.org/abs/2002.02887 https://arxiv.org/abs/1905.10437

More accurate than other ML and statistical models, some of which take into account seasonality and other time-series behaviors



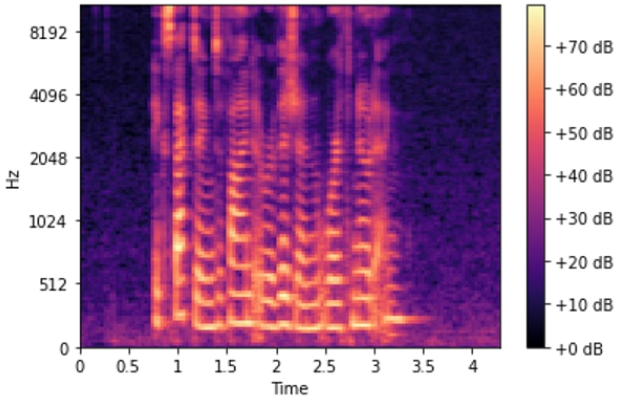
(a) Combined

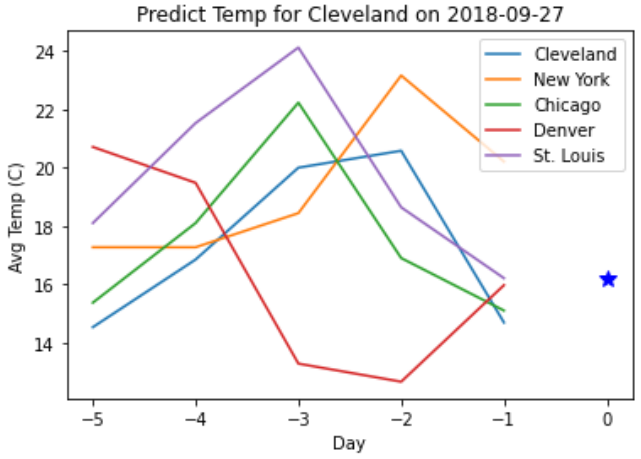
Table 1: Performance on the M4, M3, TOURISM test sets, aggregated over each dataset. Evaluation metrics are specified for each dataset; lower values are better. The number of time series in each dataset is provided in brackets.

M4 Average (100,000)			M3 Average (3,003)		TOURISM Average (1,311)	
	sMAPE	OWA		sMAPE		MAPE
Pure ML	12.894	0.915	Comb S-H-D	13.52	ETS	20.88
Statistical	11.986	0.861	ForecastPro	13.19	Theta	20.88
ProLogistica	11.845	0.841	Theta	13.01	ForePro	19.84
ML/TS combination	11.720	0.838	DOTM	12.90	Stratometrics	19.52
DL/TS hybrid	11.374	0.821	EXP	12.71	LeeCBaker	19.35
N-BEATS-G	11.168	0.797		12.47		**18.47**

Things to remember

- Audio is best represented in terms of amplitudes of frequency ranges over time
- Audio models use vision-based architectures on Mel Spectrograms
- 1D Time classification or forecasting methods often take a windowed approach, making a prediction based on data from a fixed length of time
- Though relatively unstudied, effective approaches for time series forecasting have been demonstrated with deep networks





Next week

- Reinforcement learning (by Josh Levine)
- ML Problems from 441 Students