Analytics Vidhya

Understanding the Mel Spectrogram

Leland Roberts Follow 6 min read · Mar 5, 2020

2.4K 30

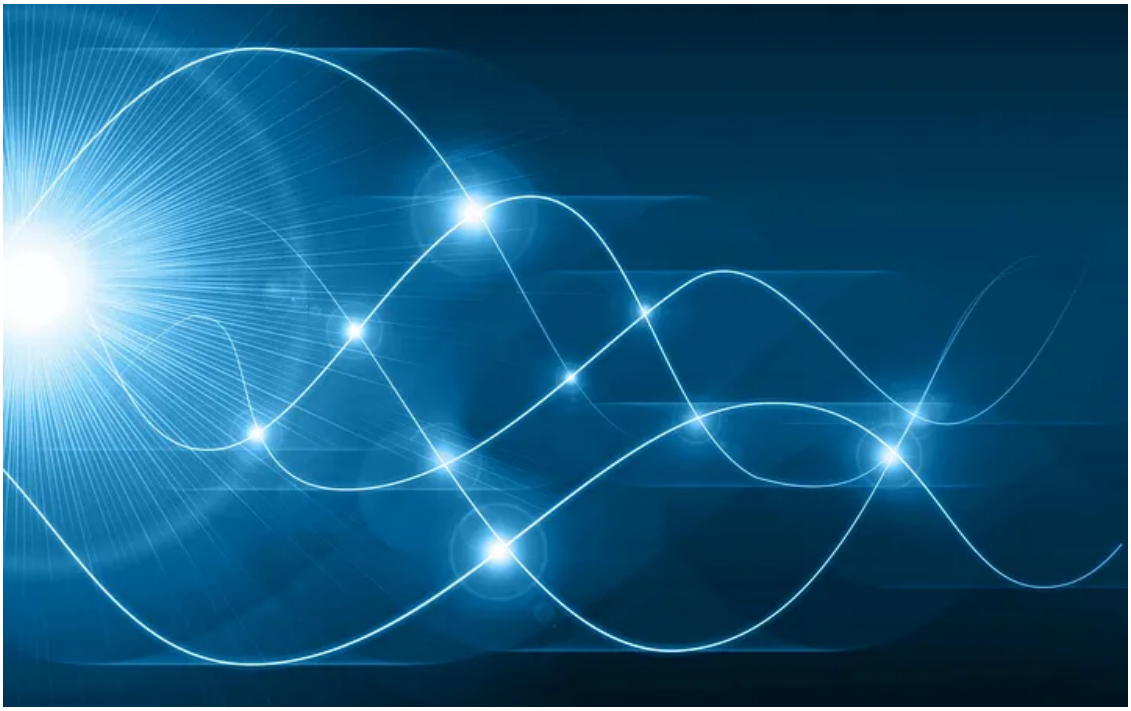


Image from Gradiom

If you are anything like me, trying to understanding the mel spectrogram has not been an easy task. You read an article only to be lead to another... and another... and another... on and on it goes. I hope this short post will clarify some of the confusion and explain the mel spectrogram from the ground up.

PGR2026-00032
Krisp Technologies EX1034 Page 1

Signals

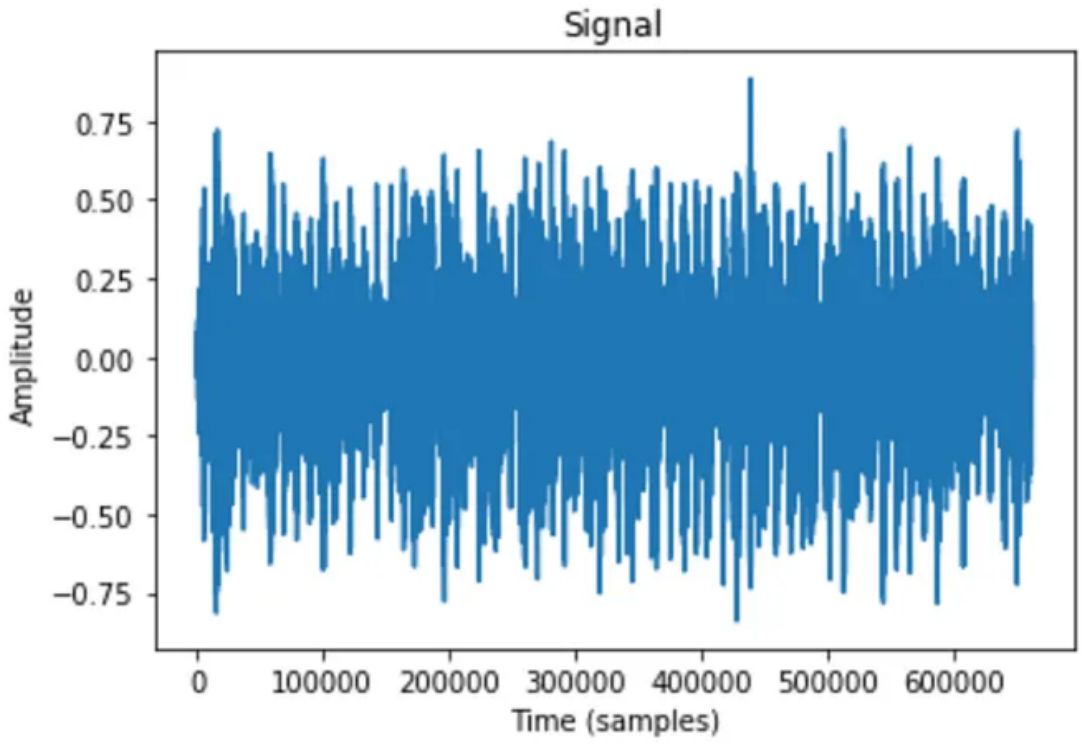
A **signal** is a variation in a certain quantity over time. For audio, the quantity that varies is air pressure. How do we capture this information digitally? We can take samples of the air pressure over time. The rate at which we sample the data can vary, but is most commonly 44.1kHz, or 44,100 samples per second. What we have captured is a **waveform** for the signal, and this can be interpreted, modified, and analyzed with computer software.

Top highlight

```
import librosa
import librosa.display
import matplotlib.pyplot as plt

y, sr = librosa.load('./example_data/blues.00000.wav')

plt.plot(y);
plt.title('Signal');
plt.xlabel('Time (samples)');
plt.ylabel('Amplitude');
```



This is great! We have a digital representation of an audio signal that we can

PGR2026-00032
Krisp Technologies EX1034 Page 2

work with. Welcome to the field of signal processing! You may be wondering though, how do we extract useful information from this? It looks like a jumbled mess. This is where our friend Fourier comes in.

The Fourier Transform

1

An audio signal is comprised of several single-frequency sound waves. When taking samples of the signal over time, we only capture the resulting amplitudes. The **Fourier transform** is a mathematical formula that allows us to decompose a signal into it's individual frequencies and the frequency's amplitude. In other words, it converts the signal from the time domain into the frequency domain. The result is called a **spectrum.**

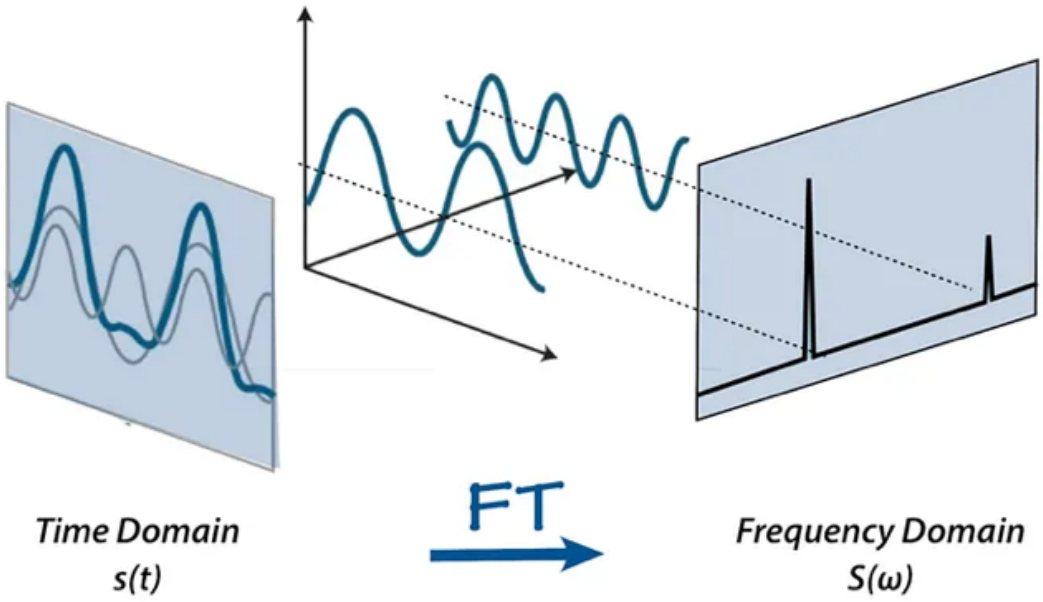


Image from Aavos International

This is possible because every signal can be decomposed into a set of sine and cosine waves that add up to the original signal. This is a remarkable theorem known as **Fourier's theorem.** Click here if you want a good intuition for why this theorems is true. There is also a phenomenal video by 3Blue1Brown on the Fourier Transform if you would like to learn more here.

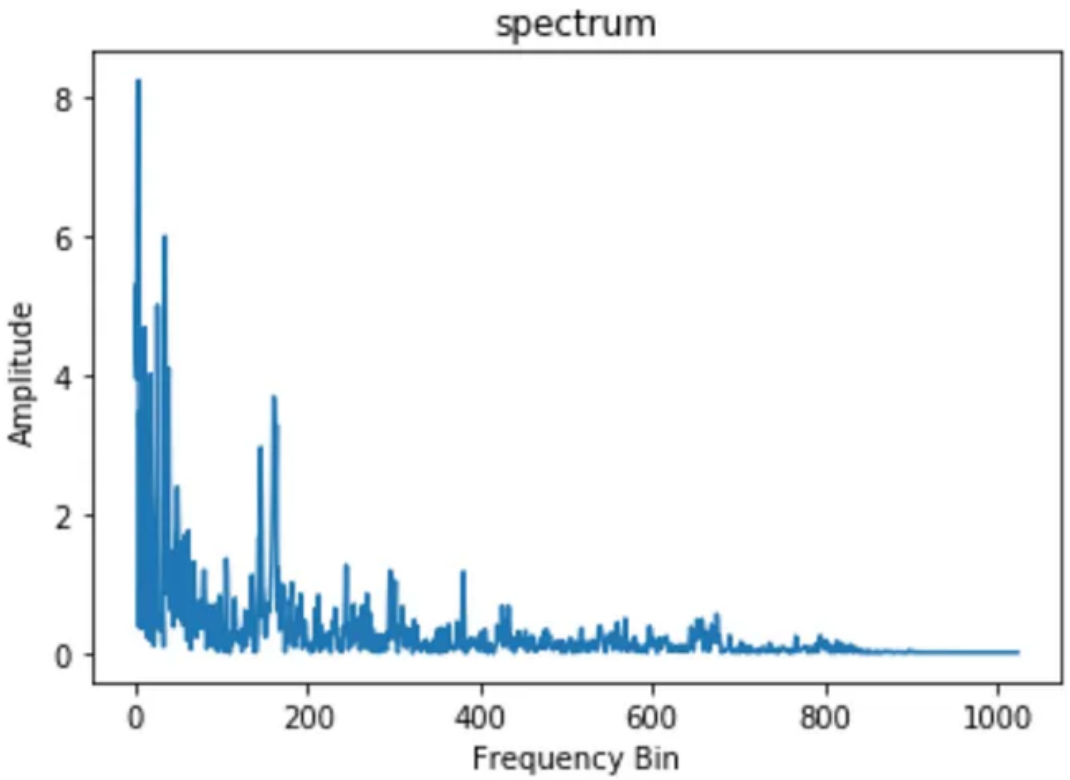
PGR2026-00032
Krisp Technologies EX1034 Page 3

The **fast Fourier transform (FFT)** is an algorithm that can efficiently compute the Fourier transform. It is widely used in signal processing. I will use this algorithm on a windowed segment of our example audio.

```
import numpy as np

n_fft = 2048
ft = np.abs(librosa.stft(y[:n_fft], hop_length = n_fft+1))

plt.plot(ft);
plt.title('Spectrum');
plt.xlabel('Frequency Bin');
plt.ylabel('Amplitude');
```



The Spectrogram

The fast Fourier transform is a powerful tool that allows us to analyze the frequency content of a signal, but what if our signal's frequency content varies over time? Such is the case with most audio signals such as music and speech. These signals are known as **non periodic** signals. We need a way to

PGR2026-00032
Krisp Technologies EX1034 Page 4

represent the spectrum of these signals as they vary over time. You may be thinking, “hey, can’t we compute several spectrums by performing FFT on several windowed segments of the signal?” Yes! This is exactly what is done, and it is called the **short-time Fourier transform.** The FFT is computed on overlapping windowed segments of the signal, and we get what is called the **spectrogram.** Wow! That’s a lot to take in. There’s a lot going on here. A good visual is in order.

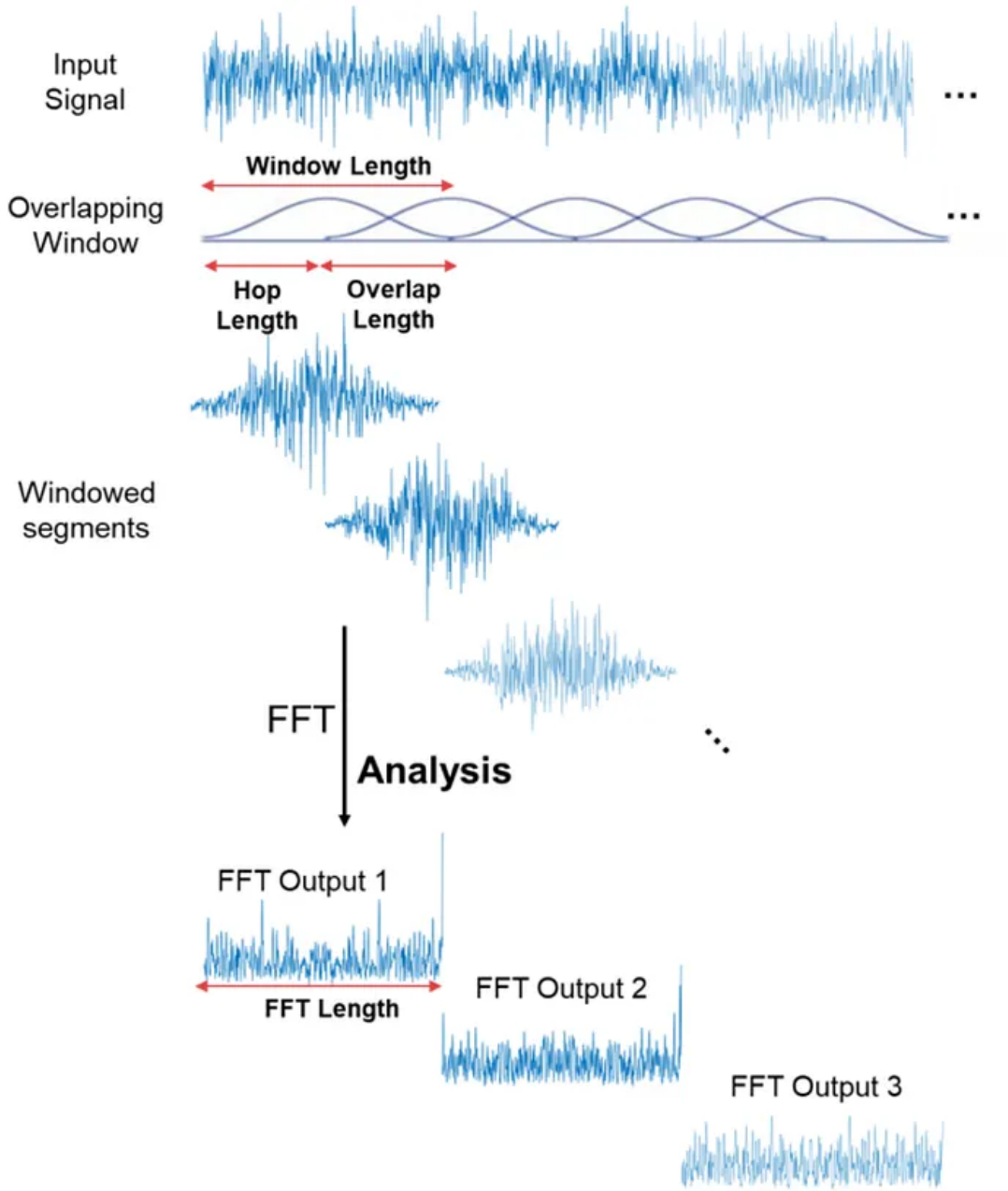


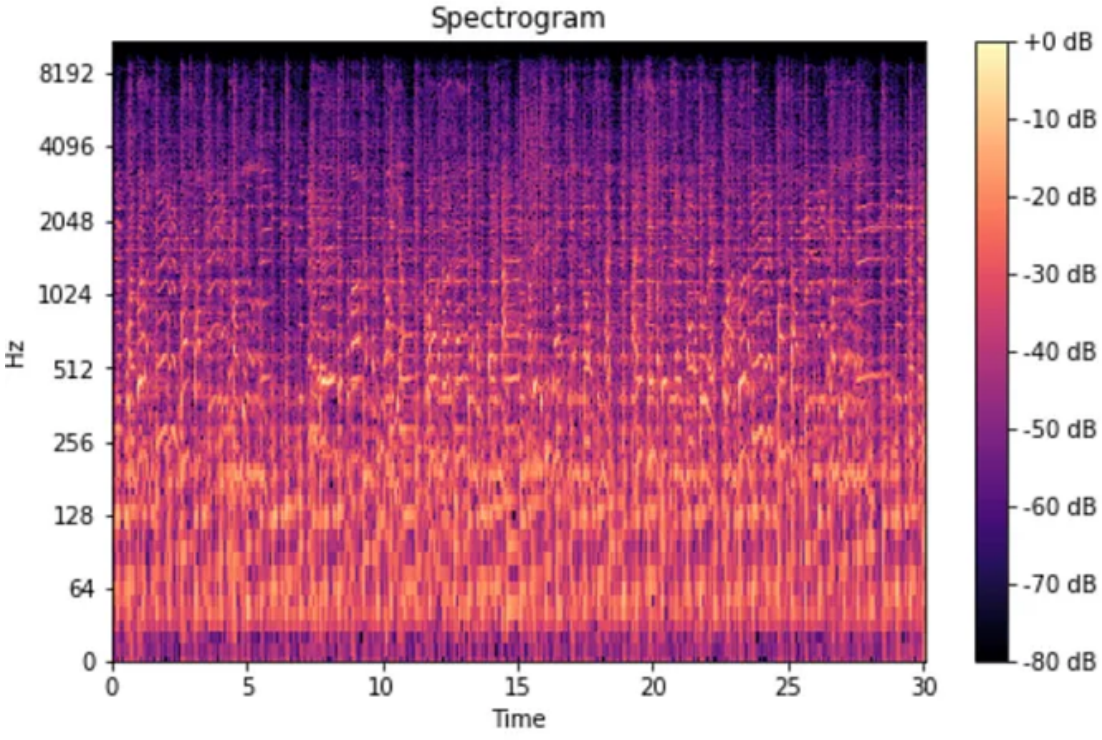
Image from MathWorks

PGR2026-00032
Krisp Technologies EX1034 Page 5

You can think of a spectrogram as a bunch of FFTs stacked on top of each other. It is a way to visually represent a signal's loudness, or amplitude, as it varies over time at different frequencies. There are some additional details going on behind the scenes when computing the spectrogram. The y-axis is converted to a log scale, and the color dimension is converted to decibels (you can think of this as the log scale of the amplitude). This is because humans can only perceive a very small and concentrated range of frequencies and amplitudes.

```
spec = np.abs(librosa.stft(y, hop_length=512))
spec = librosa.amplitude_to_db(spec, ref=np.max)

librosa.display.specshow(spec, sr=sr, x_axis='time', y_axis='log');
plt.colorbar(format='%+2.0f dB');
plt.title('Spectrogram');
```



Voila! With only a couple lines of code, we have created a spectrogram. Ok. We are almost there! We have a solid grasp on the "spectrogram" part, but what about "Mel." Who is he?

PGR2026-00032
Krisp Technologies EX1034 Page 6

The Mel Scale

Studies have shown that humans do not perceive frequencies on a linear scale. We are better at detecting differences in lower frequencies than higher frequencies. For example, we can easily tell the difference between 500 and 1000 Hz, but we will hardly be able to tell a difference between 10,000 and 10,500 Hz, even though the distance between the two pairs are the same.

Get Leland Roberts's stories in your inbox

Join Medium for free to get updates from this writer.

Enter your email Subscribe

In 1937, Stevens, Volkmann, and Newmann proposed a unit of pitch such that equal distances in pitch sounded equally distant to the listener. This is called the **mel scale.** We perform a mathematical operation on frequencies to convert them to the mel scale.

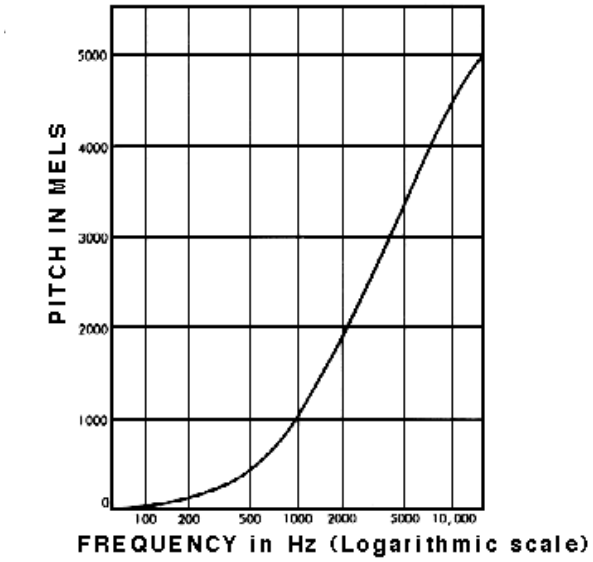


Image from Simon Fraser University

The Mel Spectrogram

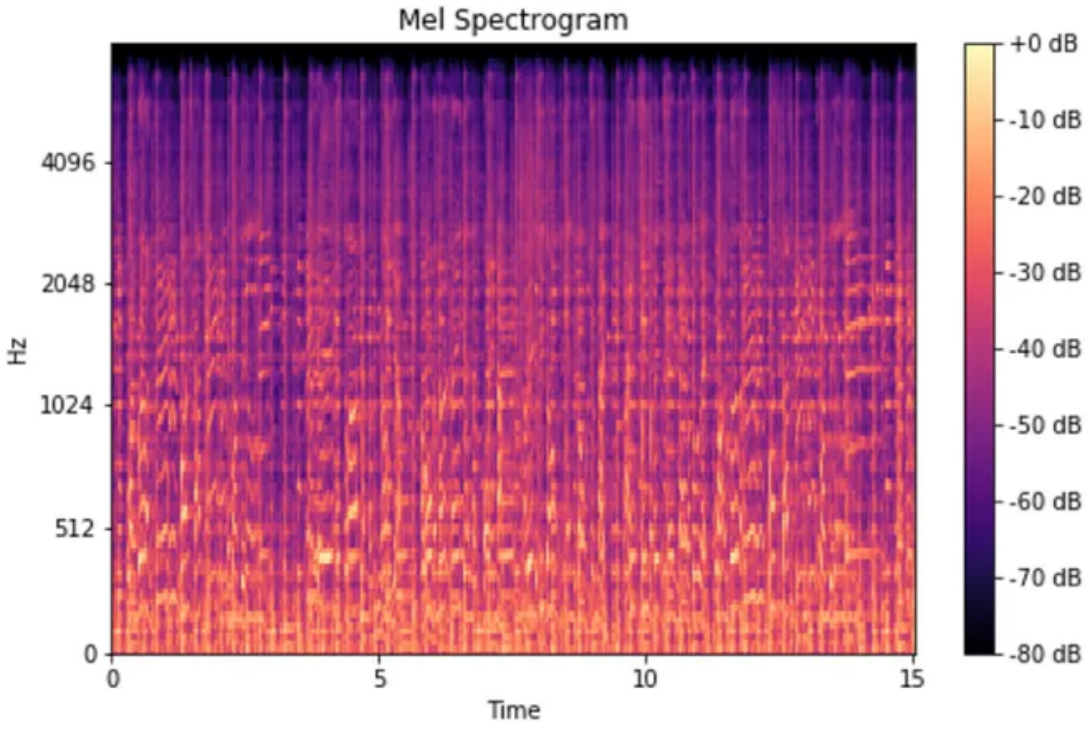
PGR2026-00032
Krisp Technologies EX1034 Page 7

A **mel spectrogram** is a spectrogram where the frequencies are converted to the mel scale. I know, right? Who would've thought? What's amazing is that after going through all those mental gymnastics to try to understand the mel spectrogram, it can be implemented in only a couple lines of code.

```
mel_spect = librosa.feature.melspectrogram(y=y, sr=sr, n_fft=2048,
hop_length=1024)
mel_spect = librosa.power_to_db(spect, ref=np.max)

librosa.display.specshow(mel_spect, y_axis='mel', fmax=8000,
x_axis='time');
plt.title('Mel Spectrogram');
plt.colorbar(format='%+2.0f dB');
```

1



PGR2026-00032
Krisp Technologies EX1034 Page 8

In Summary

That was a lot of information to take in, especially if you are new to signal processing like myself. However, if you continue to review the concepts laid out in this post (and spend enough time staring at the corner of a wall thinking about them), it'll begin to make sense! Let's briefly review what we have done.

1. We took samples of air pressure over time to digitally represent an audio **signal**
2. We mapped the audio signal from the time domain to the frequency domain using the **fast Fourier transform**, and we performed this on overlapping windowed segments of the audio signal.
3. We converted the y-axis (frequency) to a log scale and the color dimension (amplitude) to decibels to form the **spectrogram**.
4. We mapped the y-axis (frequency) onto the **mel scale** to form the **mel spectrogram**.

That's it! Sounds easy, right? Well, not quite, but I hope this post made the mel spectrogram a little less intimidating. It took me quite a while to understand it. At the end of the day though, I found out that Mel wasn't so standoffish.

> *If you would like to see a cool application of this topic, check out my post on musical genre classification where I use mel spectrograms to train a convolutional neural network to predict the genre of a song. How does it do? Find out here!*

Signal Processing | Librosa | Spectrogram | Python | Fourier Transform

PGR2026-00032
Krisp Technologies EX1034 Page 9

Published in Analytics Vidhya

77K followers · Last published Dec 15, 2025

Follow

Analytics Vidhya is a community of Generative AI and Data Science professionals. We are building the next-gen data science ecosystem https://www.analyticsvidhya.com

Written by Leland Roberts

293 followers · 27 following

Follow

Data Storytelling | Math | Driven by Curiosity

PGR2026-00032
Krisp Technologies EX1034 Page 10

Responses (30)

Write a response

What are your thoughts?

monkeyman
Dec 17, 2020 (edited)

For your code on the Mel Spectrogram section, the line

mel_spect = librosa.power_to_db(spect, ref=np.max)...

should the code be

..._db(mel_spect,...??

9 1 reply Reply

Edward Leardi
Oct 6, 2020

> An audio signal is comprised of several single-frequency sound waves. When taking samples of the signal over time, we only capture the resulting amplitudes. The Fourier transform is a m...

This is possible because every signal can be decomposed into a set of sine and cosine waves that add up to the original signal. This is a remarkable theorem known as Fourier's theorem.

57 Reply

Please dont bother me with accounts
Sep 22, 2022

Clearest explanation not only on mel spectrogram but regarding audio/signal I have seen. Incredibly well done. Thank you!

6 Reply

See all responses

PGR2026-00032
Krisp Technologies EX1034 Page 11

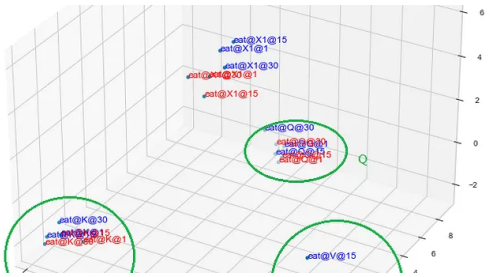
Medium Search Write Sign up Sign in

In TDS Archive by Leland Roberts

Musical Genre Classification with Convolutional Neural Networks

A Practical Application of the Mel Spectrogram

Mar 14, 2020 148 2

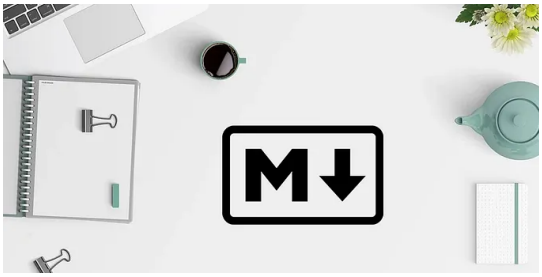


In Analytics Vidhya by mustafac

Understanding Q,K,V In Transformer(Self Attention)

Here I assume you already know what is Transformer and just want to understand...

Jan 1, 2021 208 1

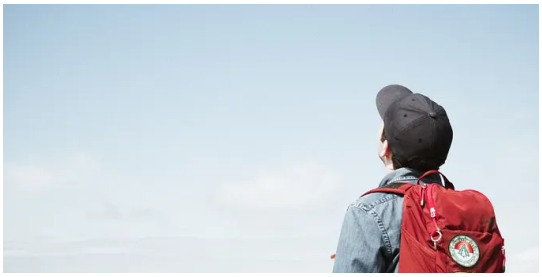


In Analytics Vidhya by Hannan Satopay

The Ultimate Markdown Guide (for Jupyter Notebook)

An in-depth guide for Markdown syntax usage for Jupyter Notebook

Nov 18, 2019 2.5K 16



In TDS Archive by Leland Roberts

The Power of Data: My Journey Into the Field

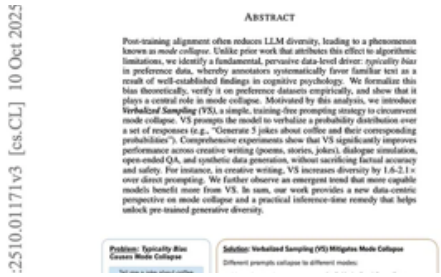
Data Science Discovered in an Unexpected Place

Mar 23, 2020 76

See all from Leland Roberts See all from Analytics Vidhya

PGR2026-00032
Krisp Technologies EX1034 Page 12

Recommended from Medium

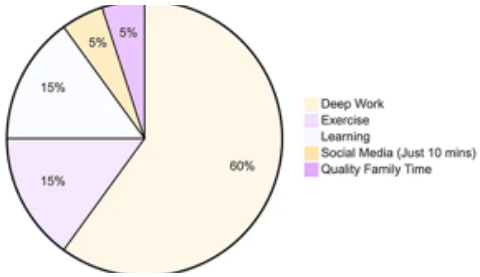


In Generative AI by Adham Khaled

Stanford Just Killed Prompt Engineering With 8 Words (And I...

ChatGPT keeps giving you the same boring response? This new technique unlocks 2×...

Oct 19, 2025 24K 632



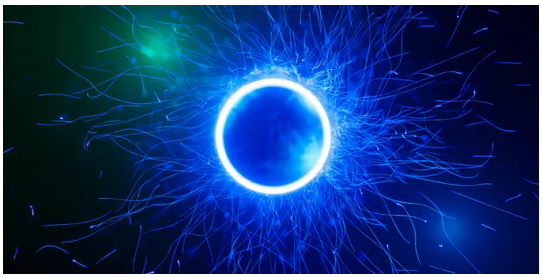
In Level Up Coding by Teja Kusireddy

I Stopped Using ChatGPT for 30 Days. What Happened to My Brai...

91% of you will abandon 2026 resolutions by January 10th. Here's how to be in the 9% who...

Dec 28, 2025 6.8K 274

PGR2026-00032
Krisp Technologies EX1034 Page 13

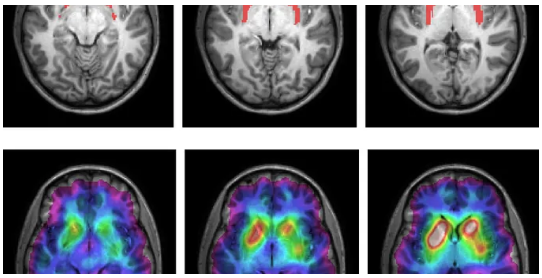


Will Lockett

The AI Bubble Is About To Burst, But The Next Bubble Is Already...

Techbros are preparing their latest bandwagon.

Sep 14, 2025 22K 964

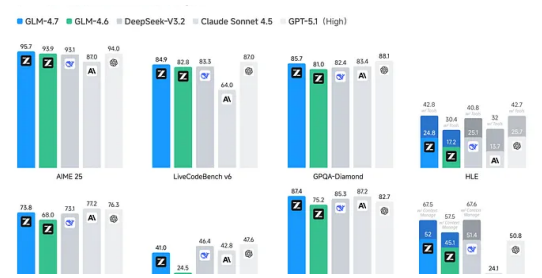


In Write A Catalyst by Dr. Patricia Schmidt

As a Neuroscientist, I Quit These 5 Morning Habits That Destroy You...

Most people do #1 within 10 minutes of waking (and it sabotages your entire day)

Jan 14 31K 559



Agent Native

Local LLMs That Can Replace Claude Code

Small team of engineers can easily burn >$2K/mo on Anthropic's Claude Code...

Jan 20 1.3K 39

In AI Software Engineer by Joe Njenga

Anthropic Just Released Claude Code Course (And I Earned My...

Anthropic just launched their Claude Code in Action course, and I've just passed—how...

Jan 21 2.8K 37

See more recommendations

PGR2026-00032
Krisp Technologies EX1034 Page 14

Help Status About Careers Press Blog Privacy Rules Terms Text to speech

PGR2026-00032
Krisp Technologies EX1034 Page 15