

Spektrogram

```
spectrogram(x, window, noverlap, nfft)
```

Spektrogram är ett sätt att visa en signals innehåll i frekvensdomänen. Det som visas är magnituden/amplituden i olika frekvenser (mellan 0 och till halva samplingsfrekvensen) över tid. En färgskala anger amplituden från svart/mörkblått till ljusgult/vitt.

Läs mer här:

<https://se.mathworks.com/help/signal/ref/spectrogram.html>

<https://en.wikipedia.org/wiki/Spectrogram>

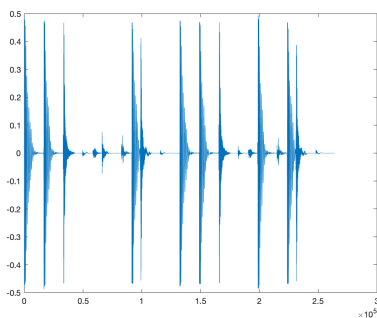
En utmaning i spektrogrammet är osäkerhetsprincipen i Fouriertransformen. Det leder till att det går antingen att ha hög upplösning i tid eller hög upplösning i frekvens, men inte båda samtidigt.

Läs mer här:

https://en.wikipedia.org/wiki/Fourier_transform

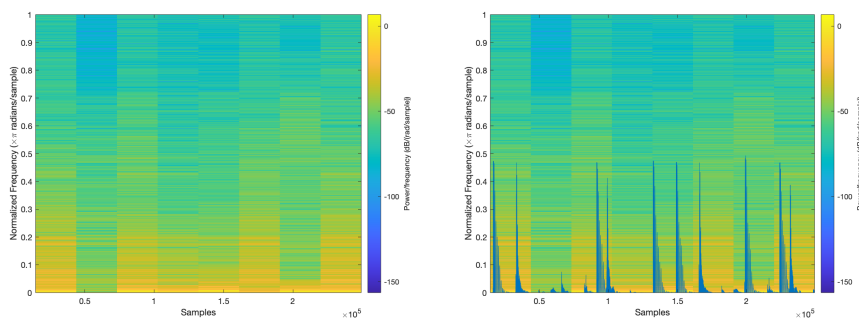
Läsa in och plotta en ljudfil

Med `[y, Fs] = audioread('filename.wav');` läses en text fil i Matlab, och med `plot(y)` visas vågformen.



Att visa ett spektrogram

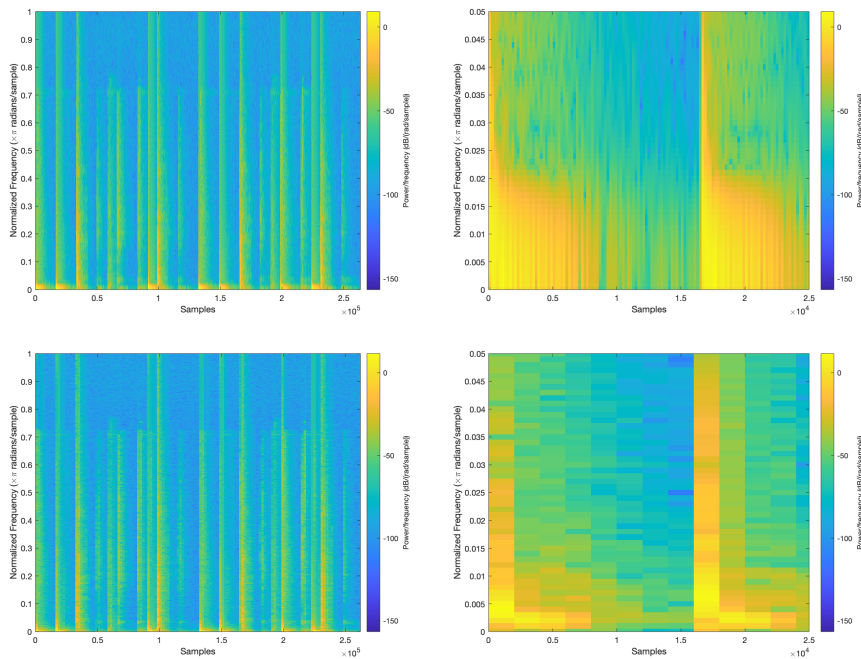
Med hjälp av `spectrogram(y, 'yaxis')` skapas ett spektrogram med defaultinställningar. Strängen 'yaxis' gör att frekvensen är på y-axeln och tid på x-axeln, vilket i alla fall jag tycker är enklare att läsa.



Detta spektrogram är inte så tydligt (till vänster), men det går att se/ana en periodicitet i ljudet, och att det är mest låga frekvenser närvarande. Upplösningen i spektrogrammet är dock ganska dålig. Med `hold on` går det att plotta ljudfilen på spektrogrammet för att kunna jämföra aspekter i tid (till höger). Signalen som plottas då visas bara till hälften då ljudet svänger runt 0 mellan -1 och 1, medan spektrogrammet plottar mellan 0 och 1 på y-axeln.

Fönster i spektrogram

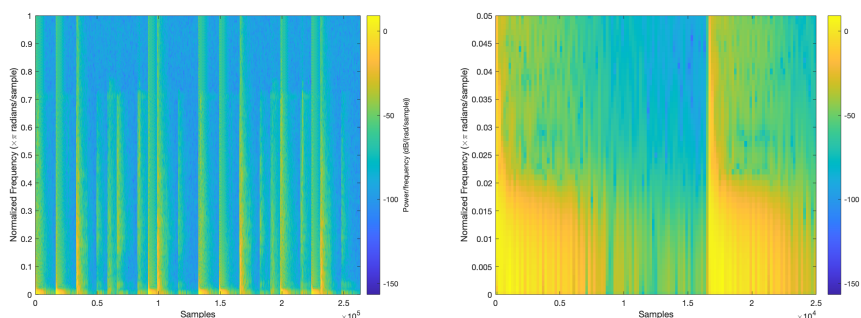
Med `window` bestäms fönsterstorleken. Det innebär att signalen delas in i segment som sedan används i analysen av frekvensen. Tänk det som att antalet fönster ger upplösningen i frekvens och tid som ska visas i spektrogrammet. 512 kan vara ett bra värde för `window`, eller kanske 1% av samplingsfrekvensen (dvs 441 i detta exempel). För att hitta rätt måste man experimentera lite. Storleken på fönstren är en trade-off mellan upplösning i tid (tidsdomän) och upplösning i frekvens (frekvensdomän). Det går att få perfekt upp lösning i en domän mot att den andra domänen har noll upplösning. Om frekvensinnehållet är det viktiga bör fönsterstorleken ökas, och tvärt om.



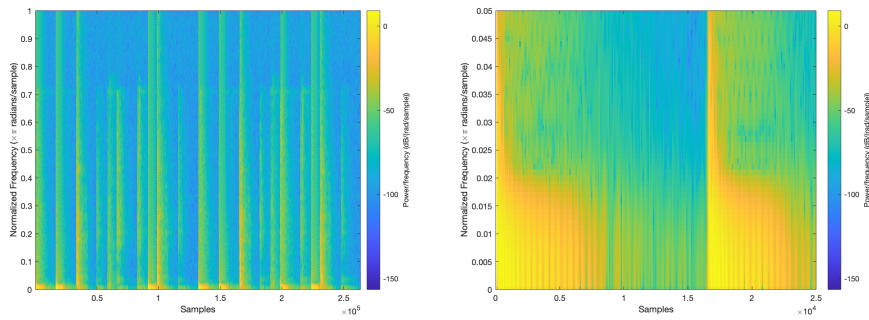
Överst används ett värde för `window` på 200 och underst på 2000, och bilderna till höger är inzoomade visualiseringar med samma värden.

Överlapp i spektrogram

Med `noverlap` anges antalet samples som ska överlappa mellan de olika segmenten (fönstren) i analysen. Även detta ökar noggrannheten av analysen vilket syns i visualiseringen. Oftast används ett överlapp som är halva fönsterbredden.



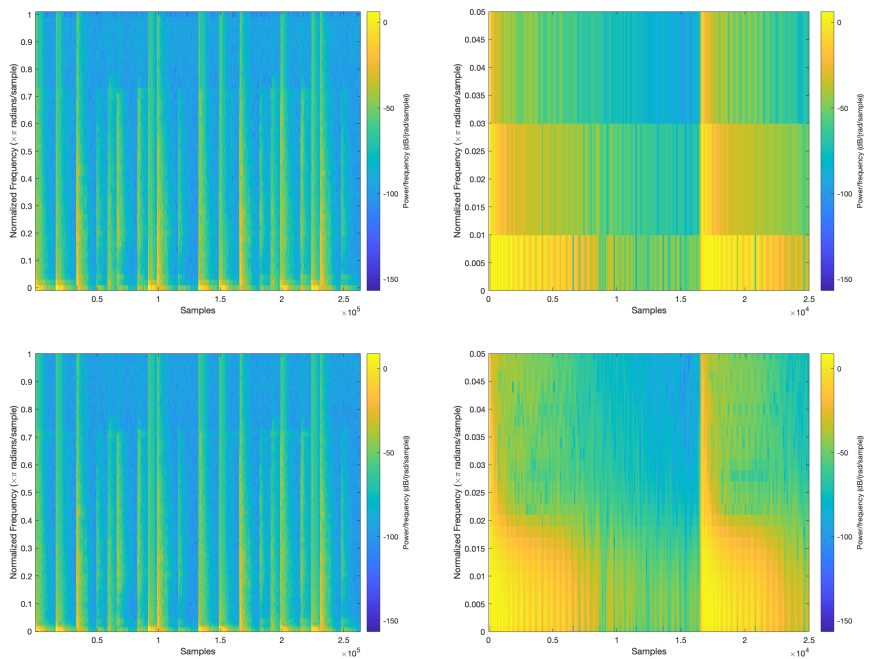
TNM103 - Ljudteknik



I båda dessa används en fönsterstorlek på 200, och överst används ett överlapp på 0 samples, och underst ett överlapp på 100 samples.

Antalet samplingspunkter i FFTn

Nästa parameter som kan anges är antalet samplingspunkter som ska användas för att beräkna den diskreta Fouriertransformen, `nfft`. Överst används 100 samplingspunkter och underst 1000.



Anpassa och normalisera mot samplingsfrekvensen

Avslutningsvis kan F_s stoppas in efter `window`, `noverlap`, `nfft` och innan `'yaxis'` så visas y-axeln i frekvens (Hz) och färgskalan normaliseras för tydligare visning.

