Professores:	César Teixeira Lorena Itatí Petrella	**Disciplina:**	Análise e Transformação de Dados
Assunto:	Mini-Projeto ATD	**Data:**	11 de março de 2024
Alunos:	Raul Sofia Ricardo Guegan	**Nº. de Estudante:**	2019225303 2020211358
Curso:	LEI	**Turmas**	PL1 e PL5

Relatório do Mini Projeto de ATD

Todo o código desenvolvido para resolver este enunciado está presentes no ficheiro `ProjetoATD.m` fornecido em anexo.

Introdução

Neste projeto vamos explorar reconhecimento de dígitos através de discurso. Vamos, a partir de um dataset onde um grupo de indivíduos pronunciam os números de zero a dez, ter de encontrar features capazes de identificar cada um dos números. Para tal faremos a análise dos sinais no domíno do tempo e da frequência e contruiremos uma arvore de decisão navegável para chegar à resposta.

Análise no domínio do tempo

A primeira etapa é ler os ficheiros de dados ".wav"e observá-los no domínio do tempo e tentar perceber características de cada dígito.

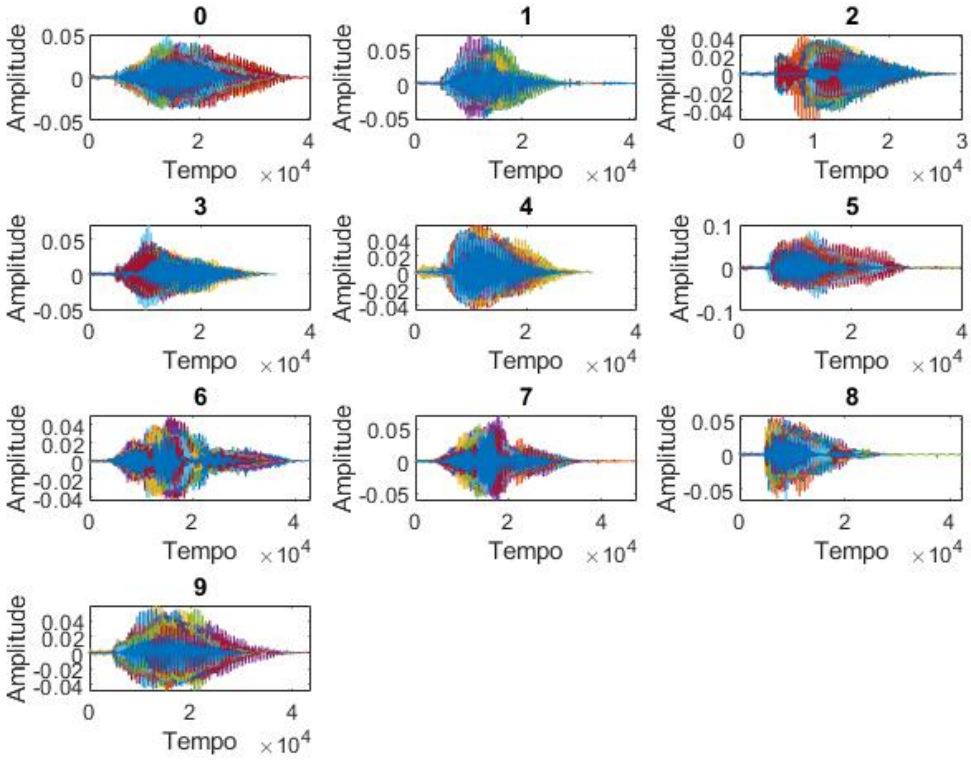

Figura 1: Dados no domínio do tempo

Como podemos ver na imagem, diferentes sinais do mesmo dígito têm amplitudes diferentes, pois a amplitude depende do volume da voz no momento da gravação. Têm ruido diferente, pois este depende do momento em que o indivíduo começa a gravar até dizer o número e depende do momento onde é cortada a gravação, bem como da velocidade a que é dito. Decidimos então que deviamos corrigir os sinais para que estes sejam mais concordantes nas amplitudes, no tempo e no alinhamento. Para a normalização do sinal na amplitude, dividimos o sinal pela sua amplitude máxima. Assim, todos os sinais ficam no intervalo que vai de -1 a 1. Para a normalização no eixo dos x, tentamos distinguir os tais ruídos iniciais e finais do sinal propriamente dito. Para tal, calculámos a energia do sinal em janelas de 100 amostras cada. Caso a energia dessa janela seja maior que um certo limiar, consideramos que já é sinal útil a partir daí. Utilizámos o mesmo algoritmo para o ruído final, mas agora considerando que as janelas se movem na direção contrária e a partir do fim. Esta filtragem revelou-se já bastante melhor que o original, mas ainda pudemos observar desalinhamento dos sinais em casos em que um ou outro artefacto elevavam muito momentaneamente a energia do sinal localmente ainda dentro da zona de ruído. Para corrigir estes casos, adicionámos ao algoritmo de limpeza um fator de tolerância: se for encontrada uma janela que se julga já ser sinal útil, verifica-se se as n (n=10, para o nosso caso) janelas seguintes também podem ser classificadas como tal. Se dentro dessas se encontrar ruído, descarta-se a janela inicial e prossegue-se com o algoritmo.

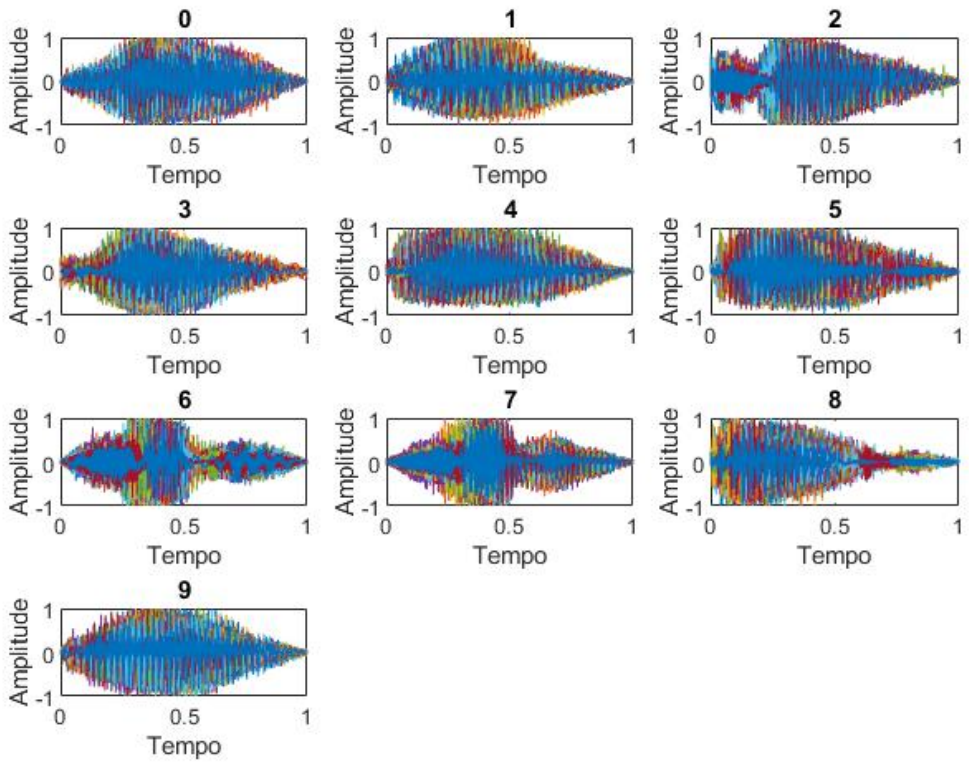

Figura 2: Dados no domínio do tempo apos normalização em ambos os eixos

Agora é mais visível um alinhamento nas amplitudes entre todos os sinas dentro de um mesmo dígito, sendo assim mais simples encontrar padrões comuns dentro de um mesmo dígito. Na figura 2, isto está muito evidente (compare-se a evolução do alinhamento dos sinais com a figura 1). Para fazer a identificação dos padrões utilizamos as métricas de energia total, evolução da energia do sinal no tempo e picos de amplitudes.
Métrica dos picos: contamos a quantidade de valores de amplitude que ultrapassam um determinado limiar e como é notável no gráfico, os digitos 2, 6 e 7 tem um teor em picos de grande

amplitude. Após realizarmos alguma testagem obtemos esta condição que consegue retirar esses digitos do conjunto de possibilidades que um sinal pode ser com uma accuracy de 96%.

```
treshHold = -0.80;
hit = zeros(10,1);
for sinal = 1:50
    for digito = 0:9
        count = sum(trim_waves{sinal,digito+1}<treshHold);
        if count<10
            hit(digito+1,1) = hit(digito+1,1) + 1;
            possibilidades(2+1) = -1;
            possibilidades(6+1)=-1;
            possibilidades(7+1)=-1;
        end
    end
end
```

Figura 3: Condiçao das grandes amplitudes

De modo a explorarmos melhor a progressão da energia pelo sinal, decidimos dividir o sinal em pequenas janelas e ver como é que a energia evoluia entre janelas consecutivas.

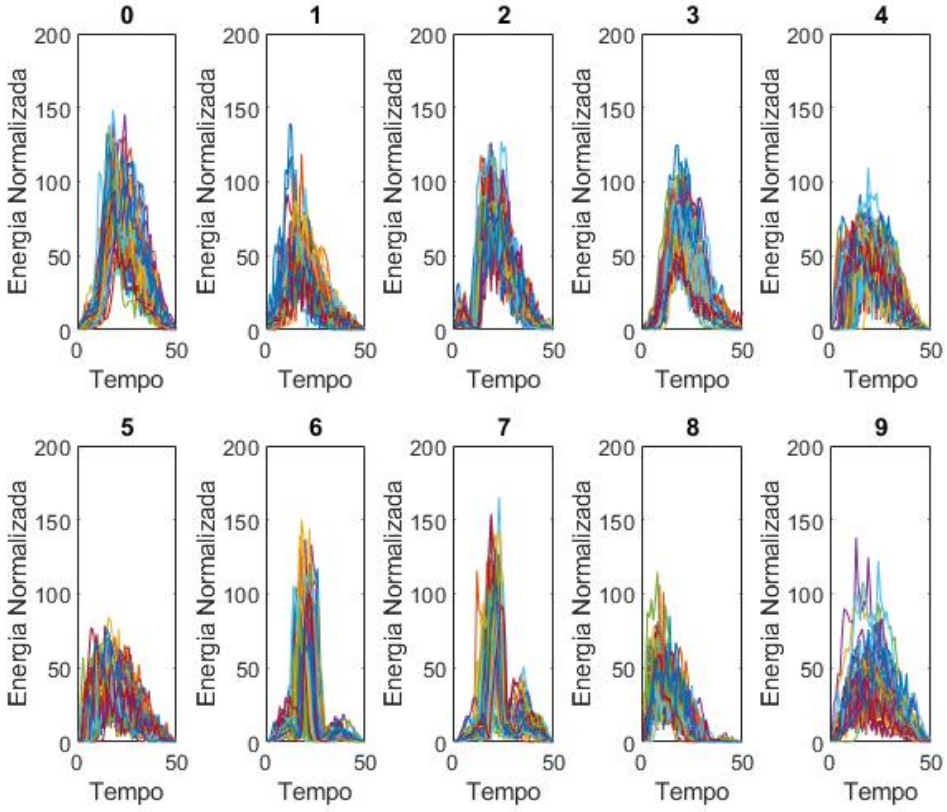

Figura 4: progressão da energia no sinal

Com base neste gráfico da evolução da energia, conseguimos perceber os seguintes padrões:

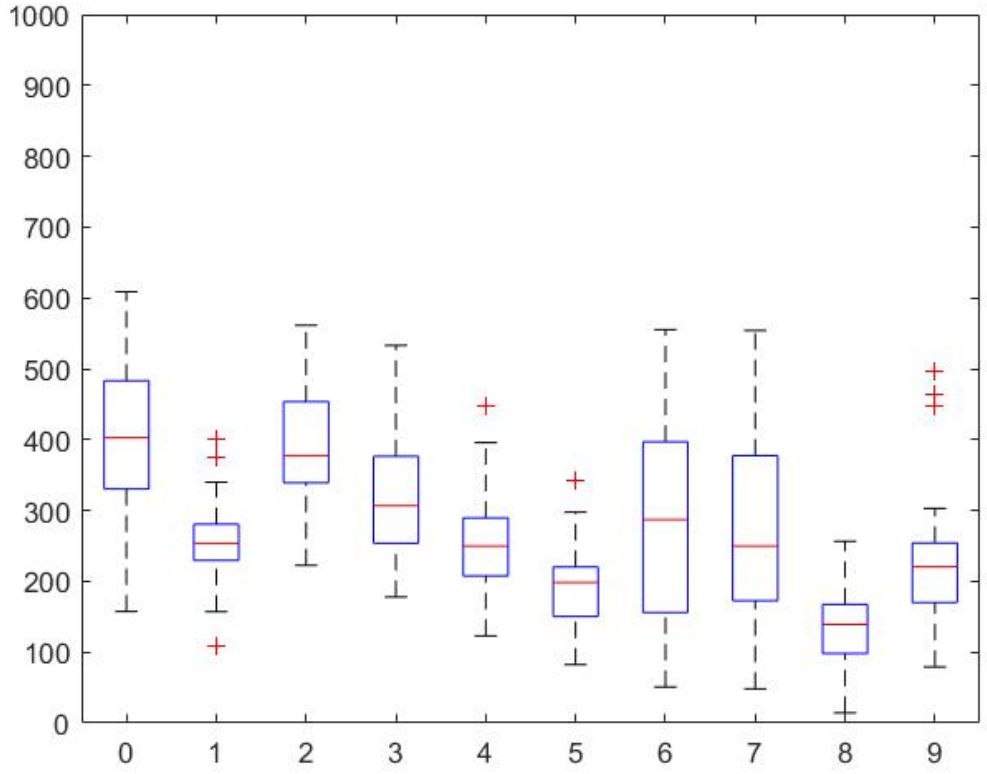

Figura 5: Energia no range de 10 a 20 % do sinal

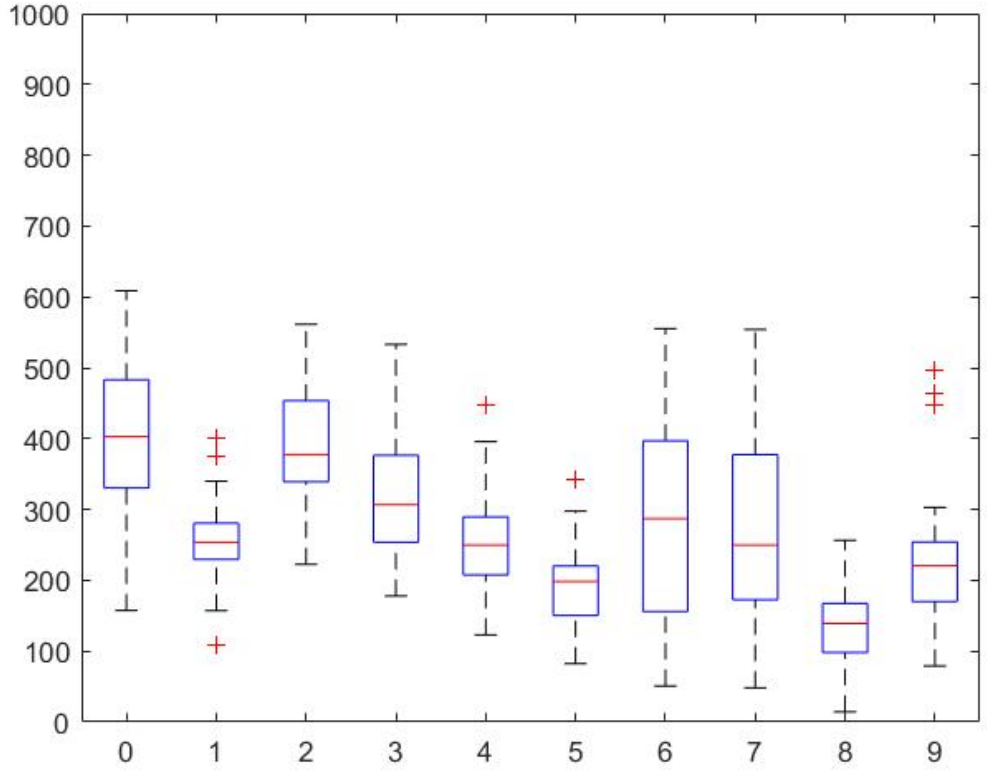

Figura 6: Energia no range de 70 a 80 % do sinal

Com estas duas tabelas de energia combinadas conseguimos distinguir os dígitos **{0;2;3;6;7}**

Análise no domínio da frequência

Vamos agora passar o sinal para o domínio da frequência para tal vamos usar a tranformada de fourier. Na aplicação da tranformada de fourier utilizamos um filtro highpass, função que permite ignorar as baixas frequências que estão presentes em todos os sinais e que não contém informação útil ao reconhecimento de discurso e utilizamos uma janela de blackman pela mesma razão, pois esta é a que melhor consegue isolar a informação útil do sinal.

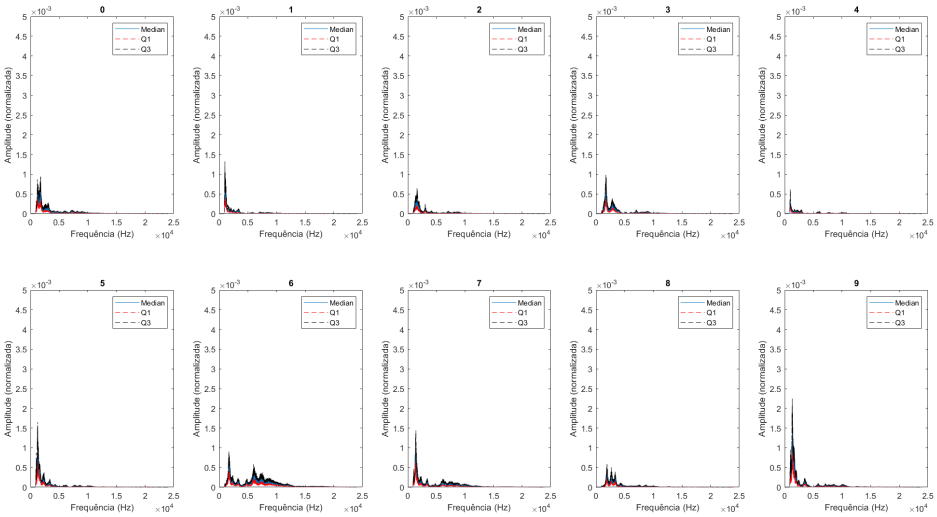

Figura 7: Resultado da aplicação da janela de Blackman

A partir da informação nesta janela as métricas que utilizamos para distinguir dígitos no domínio da frequência foi o spectrall roll off e a localização dos picos no sinal.

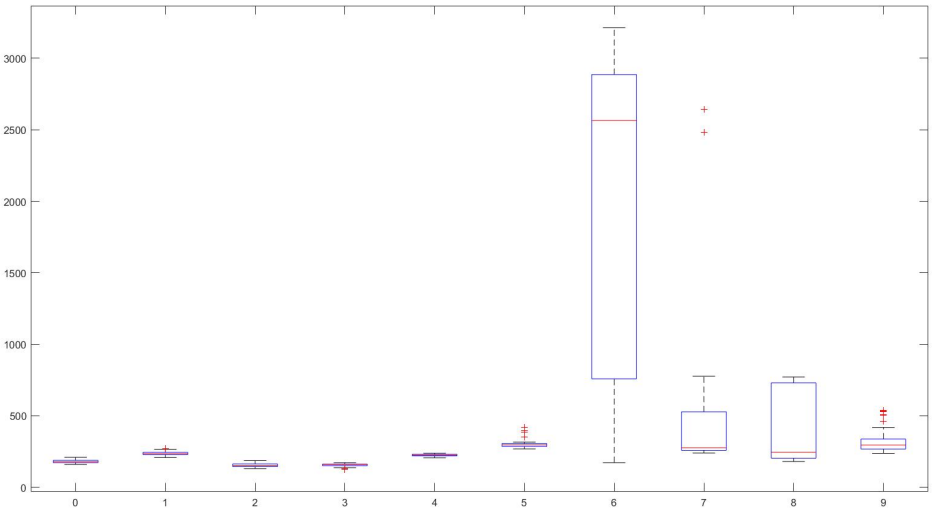

Figura 8: Boxplot do spectral roll off

Como podemos ver na imagem, os dígitos de 0 a 5 têm todos um spectral roll off muito específico o que permite fazer uma distinção entre eles muito clara, para os restantes dígitos utilizamos o

número de picos e a localização dos picos, as regras para os dígitos são:
Dígito 0 tem um spectral roll off muito específico e apresenta 2 picos nas frequências de 1000 a 2000 Hz
Dígito 1 tem um spectral roll off muito específico e apresenta 1 pico na frequência de 1000 Hz de grande amplitude
Dígito 2 tem um spectral roll off muito específico e apresenta 1 pico na frequência de 2000 Hz com amplitude media
Digito 3 tem um spectral roll off muito específico e apresenta 2 picos, um perto de 1000 Hz e outro de 3500 Hz
Digito 4 tem um spectral roll off muito específico e apresenta 1 pico na frequência de 1000 Hz com amplitude media
Digito 5 tem um spectral roll off muito específico e apresenta 1 pico na frequência de 1500 Hz de grande amplitude
Digito 6 tem 3 picos e tem um pico nas frequências dos 6000 Hz
Digito 8 tem 3 picos mas não nas frequências dos 3000Hz aos 4000 Hz todos com amplitude média
Digito 7 e 9 Ambos têm 1 pico e tem um pico nas frequências perto dos 1500 Hz, não conseguimos encontrar uma métrica para os distinguir com boa precisão

Short-Term Fourier Transform

Finalmente, depois de analisarmos os sinais no domínio temporal e no domínio de frequência, podemos agora usar a STFT para ter uma noção híbrida dos dois domínios: observar a evolução do espectro ao longo do tempo. Para tal, geramos espectrogramas das medianas dos 50 sinais de cada dígito. Usámos as transformações aplicadas anteriormente aos sinais, como o alinhamento e corte de ruído, bem como o filtro highpass para filtrar frequências não respeitantes ao reconhecimento de discurso. Foram experimentados vários parâmetros como o tamanho da janela e o overlapping. Concluiu-se que o melhor valor para estes dois, para evidenciar as diferenças, era uma janela de 1024 com um overlap de 75 % (o que torna o espectrograma mais suave). Abaixo encontram-se apresentados os ditos espectrogramas:

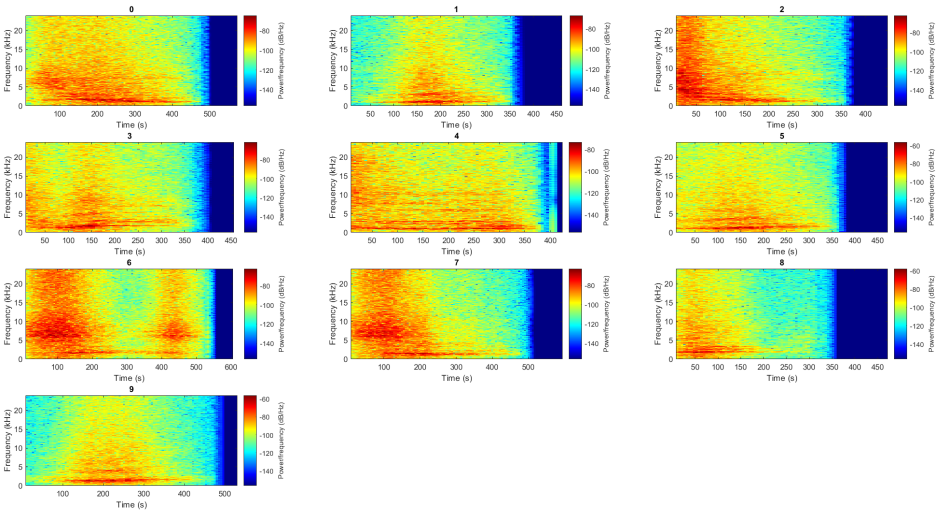

Figura 9: Resultado da aplicação da STFT

Assim sendo, conseguiram detetar-se zonas do espctrograma correspondentes a diferentes letras: consoantes como "s"e "x"têm frequências na casa dos 6000 Hz, enquanto que vogais como

"i"apresentam diferentes picos consoante a pronúncia: se for pronúnciada como "i"em português (como em "six") apresentam um pico por volta dos 1000 Hz (esta é a vogal com o som menos complexo); se em vez disso for pronúnciada de forma semelhante a um "ai"em português, como em "five", apresenta o pico anterior acrescido de um outro à volta dos 3000 Hz. O som "ou"apresenta ainda os picos anteriores com ainda mais frequências fundamentais adicionadas, nos 8000, 10000 e 15000 Hz. Podemos também dizer que, de uma forma geral, e com exceção do som "i"(como em português), se esquecermos os picos e olharmos para todo o espectrograma, podemos ver com clareza que as vogais apresentam maior distribuícção de intensidades, talvez devido à ressonância utilizada no aparelho vocal para produzir este tipo de sons (ocorre soma das frequências fundamentais de vibração de cada zona do aparelho vocal). Como as consoantes são regra geral produzidas por modulação apenas na boca não apresentam tanta dispersão.

Assim sendo, podemos distinguir os dígitos com base nas seguintes regras:
Digitos 0, 6 e 7 são sons em cujos primeiros 150 ms temos a presença de uma consoante da família "zxs", pelo que se observa o pico nos 6000 Hz. Para distinguir o 0 do 6 e o 7, usa-se os 300 a 400 ms: se tiver picos entre os 5000 e os 1000 Hz é um 0, correspondendo ao som "ou"do fim. Caso apresente apenas um pico forte na zona dos 1000 Hz trata-se de um "e"ou de um "i", e portanto de um 6 ou 7. Para distinguir estes dois, usam-se os 400 a 500 ms, de novo com a presença do pico nos 6000 Hz apenas no caso do 6, correspondendo ao último "x".

Digitos 1 5 e 9 são bastante semelhantes a nível de espectrograma. Para os distinguir dos restantes, temos os picos já referidos correspondentes ao som "ai"(1000 e 3000 Hz) no centro do sinal. A única forma de os distinguir é a dispersão no tempo desse pico: no 5 é durante quase toda a duração do sinal, enquanto que no nove é mais centrado.
Digito 2 apresenta um espectro quase uniforme nos primeiros 60 ms e depois um pico nos 1000 Hz correspondente ao "u"
Digito 3 tem distribuícção quase uniforme até aos 50 ms seguida de um pico nos 1000 Hz (o som "i").
Digito 4 tem um início (50ms) quase uniforme seguido das caraterísticas do som "ou"durante quase toda a duração, com picos nos 1000, 3000, 8000, 10000 e 15000 Hz.

Digito 8 tem um pico curto (150 ms) correspondente ao "ei"no início, seguido de baixa intensidade em todo o espectro (o "gth"é bastante silencioso, apesar de se ver algum aumento nos 6000 Hz e 1000 Hz)

Conclusão

Com este projeto pudemos aplicar técnicas de análise de sinal para detetar dígitos, porém consideramos que aquilo que fizemos sofre de problemas de overfitting, ou seja, a semelhança dos sinais entre si leva a que as regras obtidas sejam úteis apenas para sons deste dataset, prevendo-se que a precisão dos resultados piore com a inclusão de vozes de diferentes tons. Nos nossos exemplos apenas existia uma voz masculina, o que apresenta pouca variabilidade para poder separar features comuns a todas as vozes e descartar as específicas de cada timbre. Também consideramos que o modelo árvore de decisão é demasiado inflexível, o que leva mais facilmente a overfitting. Para compensar este problema propomos sistemas de classificação como matrizes de pontos ou algo mais avançado como modelos de regressão como redes neurais recorrentes. Dos diferentes domínios aquele em que conseguimos obter melhores resultados foi o de frequência na íntegra do sinal, pois é aquele em que visualmente é mais fácil identificar métricas, porém acreditamos que o metodo com mais potencial para reconhecimento de fala é o que faz uso da STFT, pois, como explicado acima, conseguimos detetar as letras e as sílabas o que permite montar depois as palavras. Isto porque o som é um sinal não estacionário que evolui no tempo logo tem de ser analisado por um metodo também capaz de demonstrar a evolução no tempo como é o caso da STFT.