



**INSTITUTO LATINO AMERICANO DE CIÊNCIAS
DA VIDA E DA NATUREZA (ILACVN)
ENGENHARIA FÍSICA**

**Estudo Sobre Extração de Características e Reconhecimento de
Padrões Aplicados em Cifragem de Músicas**

Mauro Queiroz Nooblath Neto

**Foz do Iguaçu
2021**



**INSTITUTO LATINO AMERICANO DE CIÊNCIAS
DA VIDA E DA NATUREZA (ILACVN)
ENGENHARIA FÍSICA**

Estudo Sobre Extração de Características e Reconhecimento de Padrões Aplicados em Cifragem de Músicas

Mauro Queiroz Nooblath Neto

Trabalho de Conclusão de Curso apresentado
ao Instituto Latino Americano da Vida e Na-
tureza como requisito parcial à obtenção ao
título de Bacharel em Engenharia Física

Orientador: Prof. Dr. Rodrigo Bloot

Foz do Iguaçu
2021

MAURO QUEIROZ NOOBLATH NETO

**ESTUDO SOBRE EXTRAÇÃO DE CARACTERÍSTICAS E RECONHECIMENTO DE
PADRÕES APLICADOS EM CIFRAGEM DE MÚSICAS**

Trabalho de Conclusão de Curso apresentado
ao Instituto Latino Americano da Vida e Na-
tureza como requisito parcial à obtenção ao
título de Bacharel em Engenharia Física

Orientador: Prof. Dr. Rodrigo Bloot

Aprovado pela banca examinadora em: ___ / ___ / _____

Prof. Dr. Rodrigo Bloot - UNILA
Prof.º Orientador

Prof. Dr. Tiago Antonio Alves Coimbra - CEPETRO-UNICAMP
Membro titular

Prof. Dr. Emidio Santos portilho Junior - UNIOESTE
Membro titular

AGRADECIMENTOS

Durante todas as minhas etapas de vida, as primeiras pessoas que eu tenho uma enorme gratidão em virtude de sempre fornecerem conselhos válidos, ensinarem os melhores caminhos a serem seguidos por mim, pelo carinho, por me amarem intensamente, pela dedicação, pelo afeto, pela integridade, amizade, por eu sempre poder contar mesmo estando a cerca de 3363,4km de distância longe; são meus pais: Igor Silva Nooblath e Cris Guimarães Cirino da Silva.

Agradeço também a todos da minha família, especialmente aos meus avós: Mauro Nooblath, Cleotildes Nooblath, Ivanete Guimarães e Cleonaldo Cirino por sempre estarem me apoiando e fornecendo bons conselhos.

Agradeço aos meus irmãos; Vitória Guimarães, Iago Nooblath e Kauã Nooblath a quem sempre me ouviram em conversas pessoais.

Aos meus tios: Heber Guimarães (*in memoriam*), Glauber Guimarães e Reno Nooblath que me inspiraram a estudar música, física, matemática e computação.

O meu muito obrigado a uma pessoa que foi muito especial em minha vida, a quem hoje tenho um amor e um carinho enormes, eu sempre pude contar nas horas de desespero e mesmo que de longe pude ter o apoio dela; Jackeline Brazão

Agradeço imensamente às pessoas que eu tive o primeiro contato em Foz do Iguaçu ao qual eu tenho um carinho enorme, pois adquiri grandes aprendizados com elas.

Agradeço também ao meu orientador, professor, Dr. Rodrigo Bloot, pelos conselhos, apoio, credibilidade, confiança. Adquiri grandes conhecimentos e inspiração na área da matemática em virtude dele.

Ao professor Dr. Raphael Fortes, ex coordenador do curso de Engenharia Física da UNILA. Uma enorme gratidão por tudo que foi feito pelo curso durante seu período de gestão.

Ao professor Dr. Victor Arturo, professor a quem tive grande satisfação de ter tido as disciplinas de Cálculo II e III.

A todos os amigos da graduação a quem eu tenho uma enorme gratidão, especialmente: Abner Costa, Anton Simen, Ana Carolina, Bruno Oziel, Nicolas König obrigado pela resenha de sempre.

A toda comunidade da UNILA, grande satisfação de ter feito parte como discente.

Aos amigos de longa data: Fernando Lousan, Joseph Viana e Renato Lousan.

Agradeço de uma maneira geral a todos os professores que puderam transmitir o conhecimento durante a graduação e que servirão para a vida, a todas as pessoas que participaram dessa jornada e contribuíram em minha formação.

Um imenso obrigado a todos.

“Raciocínio matemático pode ser considerado um tanto esquematicamente como o exercício de uma combinação de duas instalações , o que podemos chamar de intuição e criatividade.”
Alan Turing

Neto, Mauro Queiroz Nooblath. **Estudo sobre extração de características e reconhecimento de padrões aplicados em cifragem de músicas** .73 páginas. 2021. Trabalho de Conclusão de Curso (Graduação em Engenharia Física) – Universidade Federal da Integração Latino-Americana, Foz do Iguaçu, 2021.

Resumo

Durante a etapa de treinamento de uma rede neural é muito importante que os dados utilizados sejam tratados de tal forma que a rede consiga identificar padrões para diferenciá-los e conseguir classificá-los eficientemente. Muitos trabalhos foram realizados por diversos autores na extração de características para uma ampla base de dados. No presente trabalho, foram aplicadas duas técnicas para tratar características em um banco de dados constituído de acordes musicais com o intuito de classificá-los em acordes maiores ou menores. Uma das técnicas é conhecida como Local Phase Quantization e a outra consiste em coletar as frequências associadas aos maiores valores de amplitude. Utilizando uma rede neural, própria do Matlab-Mathworks, realizou-se uma comparação entre as duas técnicas em seu treinamento. Afim de obter melhor performance, introduzimos do uso de clusters em uma estratégia híbrida junto com as duas técnicas de extração de características na entrada da rede neural. Os resultados demonstraram uma melhora significativa de desempenho.

Palavras chave: Técnicas de extração de Features, Inteligência Artificial, Classificação de Acordes.

Neto, Mauro Queiroz Nooblath. **Study on feature extraction and pattern recognition applied in music encryption**. 73 páginas. 2021. Trabalho de Conclusão de Curso (Graduação em Engenharia Física) – Universidade Federal da Integração Latino-Americana, Foz do Iguaçu, 2021.

Abstract

During the training stage of a neural network, it is very important treat the data in such a way that the network can identify patterns to differentiate them and perform classification efficiently. Several authors have studied how to extract the features from complex databases. In the present work, two techniques were applied to extract features in a database made up of musical chords in order to classify them into major or minor chords. One of the techniques is known as Local Phase Quantization. The other one, is to collect the frequencies associated with the largest amplitude values. Using a neural network, from Matlab-Mathworks, a comparison was made between the two techniques in their training. In order to get better performance, we introduce the use of clusters in a hybrid strategy together with the two feature extraction techniques at the input of the neural network. The results demonstrated a significant improvement in the neural network performance.

Keywords Feature extraction techniques, Artificial Intelligence, Chord Classification.

LISTA DE ILUSTRAÇÕES

Figura 1	Gráfico da função 3	15
Figura 2	Gráfico da função 2.6	16
Figura 3	Piano	19
Figura 4	Acorde de Lá Maior	25
Figura 5	Notas e Frequências	27
Figura 6	Fragmento de Áudio 45 segundos	30
Figura 7	Fragmento de Áudio 3 segundos	31
Figura 8	Fragmento de Áudio 200 milissegundos	31
Figura 9	Fragmento de Áudio 1 milissegundo	31
Figura 10	Sinal $x[t]$ e Sinal $x[n]$	33
Figura 11	Sinal $x[n]$	33
Figura 12	Janela de Hann	36
Figura 13	Janela de Hamming	36
Figura 14	Comparação entre as janelas	37
Figura 15	Janela de Blackman	37
Figura 16	Comparação entre as janelas	38
Figura 17	Janela Triangular	38
Figura 18	Comparação entre as janelas	39
Figura 19	Neurônio Biológico	41
Figura 20	Sinapse Biológica	42
Figura 21	Modelo <i>Perceptron</i>	42
Figura 22	Função Limiar	44
Figura 23	Função Sigmóide	45
Figura 24	Função Retificadora	46
Figura 25	Função tangente Hiperbólica	47
Figura 26	Perceptron	48
Figura 27	Problema linearmente e não linearmente separáveis	50
Figura 28	Rede Neural Feed Forward	51
Figura 29	Rede neural e seus pesos	52
Figura 30	Função Degrau - Linear e Função Logística Sigmoidal - Não Linear	53
Figura 31	Sinal do Acorde $F\#$	55
Figura 32	a) Janela de Blackman b) Janela de Hamming c) Janela de Hann d) Janela Triangular	55
Figura 33	Frequências do Acorde $F\#$	56
Figura 34	Rede Neural com 1 camada oculta	57

Figura 35	Matrizes de Confusão da Rede Neural dada pela Figura da 34. . . .	57
Figura 36	Curvas ROC com 1 camada oculta	58
Figura 37	Rede Neural com 2 camadas ocultas	59
Figura 38	Matrizes de Confusão da Rede Neural Figura 37	59
Figura 39	Curvas ROC com 2 camadas ocultas	60
Figura 40	Matrizes de Confusão da Rede Neural para o LPQ com 40 neurônios na camada oculta e duas saídas.	61
Figura 41	Curvas ROC com 1 camada oculta de 40 com entradas fornecidas pelo método LPQ	62
Figura 42	Matrizes de Confusão da Rede Neural com duas camadas ocultas usando LPQ	62
Figura 43	Curvas ROC com 2 camadas ocultas com 40 neurônios para o método LPQ	63
Figura 44	Matriz de confusão com 1 camada oculta com 40 neurônios para frequências usando cluster	64
Figura 45	Curvas ROC com 1 camada oculta com 40 neurônios para frequências usando cluster	65
Figura 46	Matriz de confusão com 2 camadas ocultas de 40 neurônios para frequências usando cluster	66
Figura 47	Curvas ROC com 2 camadas oculta oculta de 40 neurônios para frequência usando cluster	66
Figura 48	Matriz de confusão com 1 camada oculta de 40 neurônios para LPQ usando cluster	67
Figura 49	Curvas ROC com 1 camada oculta de 40 neurônios para LPQ usando cluster	68
Figura 50	Matriz de confusão com 2 camadas ocultas de 40 neurônios para LPQ usando cluster	68
Figura 51	Curvas ROC com 2 camadas ocultas de 40 neurônios para LPQ usando cluster	69

LISTA DE TABELAS

Tabela 1	Notas e suas cifras	18
Tabela 2	Notas e seus acidentes	18
Tabela 3	Intervalos Harmônicos	22
Tabela 4	Intervalos Harmônicos	22
Tabela 5	Acordes Tríades	23
Tabela 6	Acordes Tétrades	24
Tabela 7	Distribuição de Frequências na escala de C_3 a C_4	27

SUMÁRIO

1 INTRODUÇÃO	12
2 FÍSICA DO SOM E TEORIA MUSICAL	14
2.1 FÍSICA DO SOM	14
2.2 TEORIA MUSICAL	17
2.2.1 Notas musicais	17
2.2.2 Acordes	22
2.2.2.1 Tríades	22
2.2.2.2 Tétrades	23
2.2.2.3 Inversão de Acordes	25
2.3 EQUAÇÃO DA ESCALA TEMPERADA	26
3 TÓPICOS SOBRE PROCESSAMENTO DE SINAIS	30
3.1 PROCESSAMENTO DE SINAIS	30
3.2 FILTROS	34
3.3 CONVOLUÇÃO DE SINAIS	39
4 TÓPICOS SOBRE REDES NEURAIS	40
4.1 REDES NEURAIS ARTIFICIAIS	40
4.2 NEURÔNIO BIOLÓGICO	41
4.3 REDE DE UM NEURÔNIO	42
4.3.1 Função de Ativação	43
4.3.1.1 Função Limiar	44
4.3.1.2 Função Sigmóide	44
4.3.1.3 Função Retificadora(ReLU)	45
4.3.1.4 Função Tangente Hiperbólica	47
4.4 UM EXEMPLO DE UMA REDE PERCEPTRON	47
4.5 PERCEPTRON DE VÁRIAS CAMADAS	50
4.5.1 Perceptrons com várias camadas	50
4.5.2 Backpropagation	53
5 EXTRAÇÃO DE CARACTERÍSTICAS E REDES NEURAIS	54
5.1 TREINAMENTO COM VETOR DE FREQUÊNCIAS	54
5.2 TREINAMENTO UTILIZANDO O MÉTODO LOCAL PHASE QUANTIZATION (LPQ)	60
5.3 TREINAMENTO DA REDE NEURAL UTILIZANDO O K-MEANS CLUSTERING	63
5.3.1 Aplicação nas frequências	64

5.3.2 Aplicação no LPQ	67
6 DISCUSSÃO E CONCLUSÃO	70
REFERÊNCIAS	72

1 INTRODUÇÃO

Nos tempos atuais, a tecnologia possui uma contribuição muito importante em diversas profissões. Um ramo muito versátil da tecnologia é o aprendizado de máquinas que tem por finalidade criar programas que consigam ensinar um determinado sistema a executar tarefas de maneira autônoma. Uma das áreas da aprendizagem de máquina é o que conhecemos hoje como **Deep Learning** ou aprendizagem profunda que tem por finalidade treinar um sistema para executar tarefas humanas. Por exemplo, na medicina a aprendizagem profunda pode auxiliar o profissional à tomada de decisões na hora de diagnosticar um paciente, na indústria uma inteligência artificial pode atuar para aprender a produzir a quantidade ótima de insumos otimizando o processo de produção. Para entrarmos no escopo do presente trabalho, no ramo da música, a inteligência artificial pode auxiliar um músico a aprender tocar uma música de maneira prática e rápida em instrumentos melódicos como o violão, piano e guitarra fornecendo os acordes que a constituem. O processo de identificação de acordes em uma música chama-se cifragem. Recentemente, os autores (NETO, 2015) e (JESUS; SOUZA, 2020) realizaram estudos frutíferos neste campo. O presente trabalho teve por objetivo acrescentar alguma contribuição ao tema.

De acordo com (CAVALCANTI, 2000) a implementação de um sistema de reconhecimento de padrões requer a solução de alguns problemas que são básicos como as Extrações de Características ou Features. Muitos trabalhos estão sendo feitos para resolver problemas para reconhecer padrões utilizando Redes Neurais e poucos são os trabalhos que abordam problemas de extração e tratamento de features. Para isso, existem várias técnicas para fazer o tratamento desses dados, como por exemplo o método Local Phase Quantization (LPQ) introduzido por (OJANSIVU; HEIKKILA, 2008) e utilizado por (KAPP et al., 2014) para tratar features de imagens. No presente trabalho, duas técnicas de extração de características foram investigadas e aplicadas. Os objetivos do trabalho seguem abaixo:

- Utilizar duas técnicas de extração de features com o intuito de treinar uma rede neural, própria do matlab/mathworks, para classificar corretamente os acordes em maior ou menor. Pretende-se, também, comparar estas duas técnicas de

extração de features, uma relacionada a frequências associadas com as maiores amplitudes e outra usando o LPQ. Estamos interessados em avaliar qual possui melhor performance.

- Aplicar, de maneira adaptada, o método híbrido usado em (OLIVEIRA, 2019) para tratar features de análise para aprovação de cartão de crédito para melhorar a performance da rede neural. O objetivo é adaptar a técnica e aplicar nos vetores com as features obtidos pelas duas técnicas, LPQ e frequências.

O trabalho está organizado em seis capítulos sendo o capítulo 1 destinado a introdução. No capítulo 2, fazemos uma breve introdução a alguns fundamentos de teoria musical e, no capítulo 3, uma breve exposição sobre sinais e filtros. No capítulo 4 uma breve análise sobre redes neurais é apresentada. No capítulo 5 resultados são apresentados e, por fim, discussão e conclusões são apresentadas no capítulo 6.

2 FÍSICA DO SOM E TEORIA MUSICAL

Neste capítulo descrevemos alguns aspectos da física do som bem como alguns fundamentos de teoria musical.

2.1 FÍSICA DO SOM

O som é uma onda que se propaga através da vibração das moléculas de um determinado meio podendo este ser sólido, líquido ou gasoso. De acordo com (SIMONATO; DIAS, 2011) essa vibração molecular causa uma variação de pressão muito rápida que se propaga em um meio em certa direção e tempo. Em virtude da onda sonora precisar de um meio para se propagar, dizemos que o som é uma onda mecânica. Usando lei de Hooke e mecânica de Newton, podemos descrever a propagação da onda por meio de uma equação diferencial parcial unidimensional representada por

$$c^2 \frac{\partial^2 u}{\partial x^2} = \frac{\partial^2 u}{\partial t^2} \quad (1)$$

De acordo com a equação (1) podemos dizer que $u = u(x, t)$ é uma função que varia no espaço unidimensional na direção x (adotada arbitrariamente) e no tempo. A grandeza física em questão, como dito no parágrafo anterior, é a pressão causada pela vibração molecular do meio devido a perturbação causada pela onda que se propaga em determinado espaço em um tempo t . Logo a variável u na equação (1) é esta pressão. A constante c indica a velocidade de propagação dessa onda a qual depende do local em que ocorre. No ar por exemplo, essa velocidade de propagação é aproximadamente $340m/s$. Utilizando separação de variáveis, temos que a solução da equação da onda pode ser escrita como $u(x, t) = h(x)g(t)$. No contexto de teoria musical, um acorde musical é dado pela componente temporal $g(t)$ que pode ser descrita pela função abaixo

$$acorde(t) = \sum_{j=0}^{\infty} A_j \cos 2\pi f_j t, \quad (2)$$

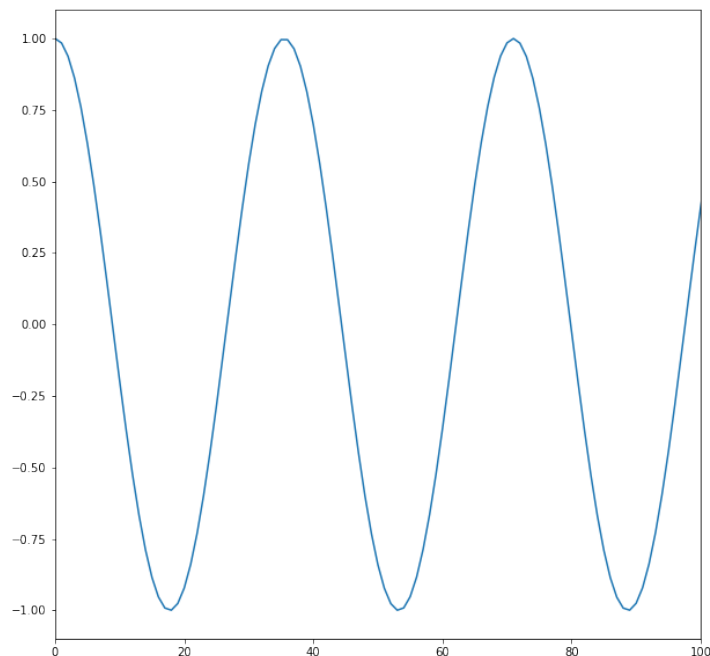
em que A_j são as amplitudes da onda e j são os índices correspondentes. Temos que

f_j são as frequências de vibração de uma onda. Observe que neste contexto, temos $g(t) = acorde(t)$. Vamos supor que para determinado índice j^* tenhamos a amplitude igual a 1 e uma frequência de $440Hz$ e as demais nulas. Portanto, teremos a seguinte função

$$acorde(t) = \cos 2\pi 440t, \quad (3)$$

cujo gráfico é mostrado na Figura 1.

Figura 1: Gráfico da função 3

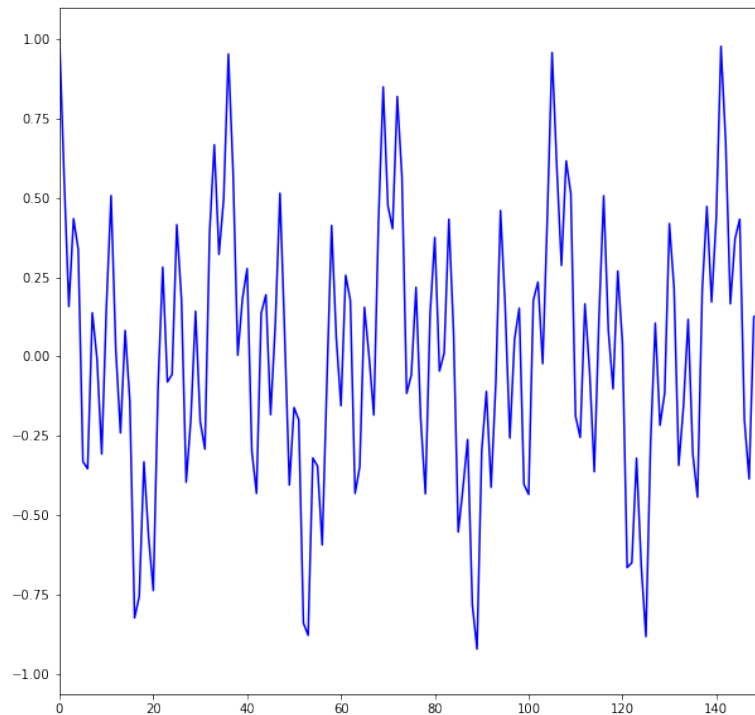


Fonte: O autor

Esta frequência é a correspondente da nota Lá, chamada de tom puro. Veremos adiante o significado de "tom puro". É importante sanar que uma nota é uma frequência simples e um acorde é uma frequência composta por várias frequências simples formada pela superposição de ondas propagadas em um tempo t . Uma vez que, de acordo com a função dada por (2) podemos ter uma solução com ondas superpostas. Suponha as seguintes frequências $440.00Hz$, $523.24Hz$ e $659.20Hz$. Neste caso, a expressão (4) representa uma superposição de três ondas com as respectivas frequências superpostas e propagadas ao mesmo tempo. O gráfico da expressão (4) é mostrado na Figura 2. Observe que, diferentemente da Figura 1, temos mais de uma onda em virtude do princípio da superposição.

$$acorde(t) = \cos(2\pi \cdot 440 \cdot 0.00 \cdot t) + \cos(2\pi \cdot 523 \cdot 24 \cdot t) + \cos(2\pi \cdot 659 \cdot 20 \cdot t). \quad (4)$$

Figura 2: Gráfico da função 2.6



Fonte: O autor

O som representado pela figura acima pode ter qualidades que podem ser distinguidas através dos nossos sentidos, como discutido em (NUSSENZVEIG, 2018). Essas qualidades são:

- Intensidade - Está relacionada com a amplitude da onda sonora. Medida do ponto de repouso da onda, até o seu ponto mais alto que o chamamos de crista.
- Altura - Correspondente a sons com frequências mais altas, sons agudos, como também sons com frequências mais baixas, sons graves.
- Timbre - Considere duas fontes sonoras independentes que emitem sons na mesma frequência. O timbre é a característica que permite-nos distinguir os sons. Veremos adiante que essa é a característica que permite distinguir um violão de uma guitarra por exemplo.

2.2 TEORIA MUSICAL

Na presente seção falaremos sobre teoria musical, com isso conseguiremos fazer a associação da matemática abordada e a música.

Quando falamos em música estamos nos referindo à uma reunião e combinação de sons, dispostos e ordenados em diversos padrões sendo constituído pelos elementos abaixo:

- **Melodia** - Onde o sentido musical é transmitido através de notas e pausas a quem se ouve durante o decurso do tempo. Chamamos de concepção horizontal da música.
- **Harmonia** - De acordo com (URTADO, 2009) harmonia é a combinação simultânea dos sons às quais preenchem a Música, encorpando e reforçando o sentido musical. Chamamos de concepção vertical da música.
- **Ritmo** - É tudo que está ligado a dimensão temporal da música. É como a harmonia e a melodia se dividem no decurso do tempo.
- **Contraponto** - É definido como o conjunto de melodias dispostas em ordem simultânea. A concepção musical é tanto horizontal como vertical.

2.2.1 Notas musicais

Durante a Idade Média a música foi reconhecida com bastante importância entre clérigos daquela época. Em virtude disso, de acordo com (SOUSA, 2012), houve a necessidade de organizar o sistema de notação musical que é reconhecido até os dias de hoje. Isso foi feito por um monge beneditino francês chamado Guido de Arezzo, nascido no final do século X. Através da percepção que criar uma escala musical simplificada poderia facilitar a compreensão dos alunos ao aprender música, a escala foi criada em função do hino cantado em louvor a São João Batista utilizando versos de estrofes em que as iniciais correspondiam a algumas das notas hoje conhecidas. Dentre elas temos o **ut, ré, mi, fá, sol, lá** e o **si** (como uma exceção obtido da junção das iniciais de um dos versos do hino "Sancte Iohannes"). Já o **dó** que substituiu o **ut**, foi somente adotado no século XVII, através de uma revisão do sistema concebido.

Ao passar dos anos, também foram adotados símbolos para identificar essas notas e que fisicamente delimitavam certas alturas do som, medida da frequência de vibração molecular de um meio em virtude da perturbação causada pela onda sonora. Na música existem as sete notas fundamentais como citadas no parágrafo anterior: **Dó, Ré, Mi, Fá, Sol, Lá, Si**. Cada uma dessas notas possuem uma respectiva cifra musical como representado na Tabela. 1.

Notas musicais e suas cifras						
Dó Maior	Ré Maior	Mi Maior	Fá Maior	Sol Maior	Lá Maior	Si Maior
C	D	E	F	G	A	B

Tabela 1: Notas e suas cifras

Vamos à algumas definições de acordo com (JESUS; SOUZA, 2020):

- **Distância** - Existem duas formas de medir a distância entre dois sons:
 - **Fisicamente** - Podemos medir a frequência de uma nota x e de uma nota y. Assim a distância física seria a diferença entre essas frequências.
 - **Musicalmente** - A distância entre dois sons é definida através da nossa sensação. Dados um som característico de uma nota x e de uma nota y, a distância musical seria a diferença entre a percepção desses sons. A grandeza que mede esta distância é chamada de Tom.
- **Semitom** - Um semitom é a menor distância entre dois sons.
- **Tom** - É a soma de dois semitons.
- **Acidentes** - São alterações nas quais modificam as alturas das notas. Podemos ter os seguintes acidentes:
 - **Sustenido** - Indicado pelo símbolo #. Tem a função de **elevantar** a nota em um semitom.
 - **Bemol** - Indicado pelo símbolo b. Tem a função de **abaixar** a nota em um semitom.

Na Tabela 2 temos todas as notas com seus acidentes. Distribuídas de tal forma que completam um intervalo de 12 semitons no qual chamamos de uma oitava.

Notas musicais e seus acidentes												
C	Db/C#	D	Eb/D#	E	F	Gb/F#	G	Ab/G#	A	Bb/A#	B	C

Tabela 2: Notas e seus acidentes

É importante notar na Tabela 2 que cada nota é um som e portanto a distância entre duas colunas consecutivas é considerada a menor distância na teoria da música ocidental entre dois sons. Portanto o intervalo entre essas colunas podem ser chamados de semitom. Logo o intervalo, por exemplo entre C e Db/C# é 0.5 tom ou 1 semitom. Veja que uma oitava possui 12 semitons. Observe também na Tabela 2 que entre as notas que possuem acidentes musicais, temos o acidente bemol ou o

acidente sustenido. A diferença dos dois está no sentido adotado na escala musical. Se estivermos no sentido esquerdo-direito (progredindo na escala) consideraremos os sustenidos, uma vez que estamos elevando às notas semitons e portanto podemos denotar estes semitons de acréscimos utilizando um sinal positivo. Por outro lado, se considerarmos o sentido direito-esquerdo (regredindo na escala), vamos adotar os bemóis, uma vez que estamos abaixando às notas semitons e portanto podemos denotar estes semitons de decréscimos utilizando um sinal negativo. De acordo com (JESUS; SOUZA, 2020) o efeito do mesmo som possuir duas representações é chamado de *enarmonia*.

Uma vez definidos os tons e semitons, como no parágrafo anterior. Podemos compreender o que chamamos de *intervalos musicais* nos quais são cinco: **M** - maior, **m** - menor, **J** - justa, **A** - aumentada, **D** - diminuta.

Na Figura 3 está representado um piano. As teclas brancas do piano indicam as notas e as teclas pretas os acidentes musicais (sustenido ou bemol) referentes àquelas notas. Observe também que temos 4 oitavas em relação a nota C e valores para cada nota definidos somente com o intuito de facilitar a identificação. Assim ficará mais fácil analisar os tons e semitons, ajudando assim, classificar os intervalos musicais que abordaremos no próximo parágrafo.

Figura 3: Piano



Fonte: Baseado em (JESUS; SOUZA, 2020)

De acordo com (Faria, Nelson, 2017), podemos classificar os intervalos musicais de dois modos, tanto quantitativamente como qualitativamente. A classificação quantitativa se dá pela contagem de semitons que constituem um determinado intervalo musical. Suponha que no piano da Figura 3 pressionamos a nota C, representada pelo número 7.0. Se pressionarmos em seguida a nota G representada pelo número 10.5, teremos as seguintes notas envolvidas representadas como elementos do intervalo 5

$$\text{Intervalo}_{CG} = [C, D, E, F, G] \quad (5)$$

A quantidade de elementos no conjunto 5 são cinco, portanto dizemos que a quinta de C é G. Observe agora os conjuntos 6 e 7

$$\text{Intervalo}CD = [C, C\#/Db, D] \quad (6)$$

$$\text{Intervalo}EF = [E, F] \quad (7)$$

Note que a diferença entre os dois conjuntos é que o conjunto 6 possui duas notas C e D com acidente musical $C\#/Db$. Por outro lado no conjunto 7 também temos duas notas E e F que não possuem acidente musical. Para ficar mais claro vamos substituir os conjuntos 6 e 7 respectivamente (utilizando a notação numérica da Figura 3), por 8 e 9. Assim vamos classificar o intervalo musical baseado na quantidade de semitons presentes nos respectivos conjuntos.

$$\text{Intervalo}CD = [7.0, 7.5, 8.0] \quad (8)$$

$$\text{Intervalo}EF = [9.0, 9.5] \quad (9)$$

Cada variação numérica nos conjuntos acima, representa um semitom. Logo no primeiro caso nós temos a presença de 2 semitons ($7.5 - 7.0 = 0.5$, $8.0 - 7.5 = 0.5$) e no segundo caso temos a presença de 1 semitom ($9.5 - 9.0 = 0.5$), mas repare que temos duas notas envolvidas nos dois casos que são respectivamente C e D, E e F. Baseado nisso, podemos dizer que em relação a nota C, a sua segunda que é D é denominada **segunda maior** pelo fato do intervalo ser constituído por 2 semitons. Em relação a nota E representada por 9, a sua segunda que é F é denominada **segunda menor**, em virtude do intervalo possuir apenas 1 semitom.

Suponha agora que pressionamos a nota G indicada na Figura 3 pelo número 10.5 e em seguida a nota D indicada na Figura 3 pelo número 14.0. O conjunto que representa esse intervalo é mostrado logo abaixo:

$$\text{Intervalo}GD = [G, A, B, C, D] \quad (10)$$

Então podemos dizer que a quinta de G é D, pois o conjunto contém cinco elementos. Vamos agora, além das notas, incluir no conjunto os acidentes musicais.

$$\text{Intervalo}GD = [G, G\#/Ab, A, A\#/Bb, B, C, C\#/Db, D] \quad (11)$$

Para facilitar mais, vamos representar os elementos do conjunto 11 pelos números representados através da Figura 3. Portanto 11 será o conjunto dado por 12.

$$\text{Intervalo}GD = [10.5, 11.0, 11.5, 12.0, 12.5, 13.0, 13.5, 14.0] \quad (12)$$

Vamos abaixar 1 semitom no extremo direito do conjunto 12. Observando o conjunto 11. A nova nota será *Db* e assim teremos o intervalo representado por 13.

$$\text{IntervaloGDb} = [10.5, 11.0, 11.5, 12.0, 12.5, 13.0, 13.5] \quad (13)$$

Veja que diminuimos o conjunto 12 e ficamos com o conjunto 13. Dizemos que *Db* é a quinta **D-diminuta** de *G*. Logo a quinta diminuta é um intervalo que possui 6 semitons contados a partir de uma determinada nota musical.

Considerando novamente o conjunto 12, vamos supor que aumentamos um semitom no extremo direito deste conjunto. Portanto será o conjunto 14.

$$\text{IntervaloGD} = [10.5, 11.0, 11.5, 12.0, 12.5, 13.0, 13.5, 14.0, 14.5] \quad (14)$$

Logo, veja que temos 8 semitons neste conjunto e que, portanto temos uma quinta **A - aumentada**. Logo a quinta aumentada é um intervalo que possui 8 semitons contados a partir de uma determinada nota musical.

Finalmente, observe que partindo da nota *G* representada pelo número 4.5 contamos até *D* representada pelo número 14.0 e assim temos a quinta **J-justa**, uma vez que temos exatamente cinco notas que constitui o intervalo. Uma característica sonora dessas notas é que pressionadas juntas elas tem um som quase indistinguível.

Na próxima seção estudaremos a formação de acordes. Para isso, é muito importante saber, de acordo com (FARIA, 1999) a classificação dos intervalos partindo de uma determinada nota (tônica) para classificar um acorde. Em virtude disso a Tabela 3 mostra quantos semitons existem nos intervalos mais comuns que constituem acordes servindo como consulta para suas respectivas classificações. Na Tabela 3 temos um exemplo onde a partir de uma determinada nota, identificamos seus intervalos.

Intervalos Harmônicos			
Intervalo	Símbolo	Semitons	Tons
Segunda menor	2 ^a m	1	1/2
Segunda maior	2 ^a M	2	1
Terceira menor	3 ^a m	3	1 + 1/2
Terceira maior	3 ^a M	4	2
Quarta justa	4 ^a J	5	2 + 1/2
Quarta aumentada	4 ^a A	6	3
Quinta diminuta	5 ^a D	6	3
Quinta justa	5 ^a J	7	3 + 1/2
Quinta aumentada	5 ^a A	8	4
Sexta menor	6 ^a m	8	4
Sexta maior	6 ^a M	9	4 + 1/2
Sétima menor	7 ^a m	10	5
Sétima maior	7 ^a M	11	5 + 1/2
Oitava	8 ^a	12	6

Tabela 3: Intervalos Harmônicos

Perceba que o intervalo Quinta aumentada possui a mesma quantidade de semitons que o intervalo da Sexta menor, como também o intervalo Quarta aumentada possui a mesma quantidade de semitons que o Quinta diminuta. São intervalos *enarmônicos* e só utilizaremos o Quinta diminuta e o Quinta aumentada no nosso exemplo.

Exemplo 1: Dadas as notas B, A e D identificar todos intervalos da Tabela 3.

Identificação de intervalos harmônicos partindo das notas dadas										
Tônica	2 ^a m	2 ^a M	3 ^a m	3 ^a M	4 ^a J	5 ^a D	5 ^a J	5 ^a A	6 ^a M	7 ^a M
B	C	C#	D	D#	E	F	F#	G	G#	A
A	A#	B	C	C#	D	D#	E	F	F#	G#
D	D#	E	F	G	G#	A	A#	B	C	C#

Tabela 4: Intervalos Harmônicos

2.2.2 Acordes

Um acorde é formada pela superposição de duas ou mais notas. Podemos ter dois tipos de acordes: **Triades** e **Tétrades**.

2.2.2.1 Triades

De acordo com (FARIA, 1999) uma tríade é formada pela superposição de duas terças no qual forma um acorde com três sons: Uma nota fundamental (tônica) e

as duas terças sobrepostas nesta. Eles são muito comuns em estilos como o Punk, Grunge, Pop rock.

Os acordes tríades podem ser: **Maior**, **Menor**, **Aumentado** ou **Diminuto**. Veremos abaixo a classificação de acordes tríades com mais detalhes na tabela 5. É importante sanar que acordes serão representados por letras maiúsculas em negrito, por exemplo o acorde de **C**, por outro lado, notas serão representadas por letras maiúsculas sem negrito, por exemplo nota de C e o nome do acorde é dado pela nota que representa a tônica (**T**).

Acordes Tríades	
Nome do Acorde	Estrutura
Acorde Maior	$T + 3^{\text{a}}M + 5^{\text{a}}J$
Acorde Menor	$T + 3^{\text{a}}m + 5^{\text{a}}J$
Acorde Diminuto	$T + 3^{\text{a}}m + 5^{\text{a}}D$
Acorde Aumentado	$T + 3^{\text{a}}M + 5^{\text{a}}A$
Acorde Suspenso (ou com quarta)	$T + 3^{\text{a}}M + 5^{\text{a}}A$

Tabela 5: Acordes Tríades

2.2.2.2 *Tétrades*

De acordo (JESUS; SOUZA, 2020) os acordes tétrades são mais encorpados e brilhantes. São bastante aplicados na Bossa Nova, Jazz, Blues, Rock progressivo. É formado pela superposição de três terças, formando um acorde de quatro sons: uma nota fundamental (tônica) e três terças sobrepostas sobre a mesma. (FARIA, 1999).

Os acordes tétrades podem ser: **Acorde Maior com Sétima Maior**, **Acorde Maior com Sétima Menor**, **Acorde Menor com Sétima Menor**, **Acorde Meio Diminuto**, **Acorde Suspenso com Sétima Menor**, **Acorde Maior com Sexta Maior**, **Acorde Menor com Sexta Maior**, **Acorde Aumentado Tétrade**. Veremos abaixo a classificação de acordes tétrades com mais detalhes. É importante sanar novamente que acordes serão representados por letras maiúsculas em negrito, por exemplo o acorde de **C**, por outro lado, notas serão representadas por letras maiúsculas sem negrito, por exemplo nota de C e o nome do acorde é dado pela nota que representa a tônica (**T**).

Acordes Tétradees	
Nome do Acorde	Estrutura
Acorde Maior com Sétima Maior	T + 3 ^a M + 5 ^a J + 7 ^a M
Acorde Maior com Sétima Menor	T + 3 ^a M + 5 ^a J + 7 ^a m
Acorde Menor com Sétima Menor	T + 3 ^a m + 5 ^a J + 7 ^a m
Acorde Meio Diminuto	T + 3 ^a m + 5 ^a D + 7 ^a m
Acorde Suspenso com Sétima Menor	T + 4 ^a J + 5 ^a J + 7 ^a m
Acorde Maior com Sexta Maior	T + 3 ^a M + 5 ^a J + 6 ^a M
Acorde Menor com Sexta Maior	T + 3 ^a m + 5 ^a J + 6 ^a M
Acorde Aumentado Tétrade	T + 3 ^a M + 5 ^a A + 7 ^a M

Tabela 6: Acordes Tétrades

Suponha agora que queiramos formar o acorde tríade de Lá Maior (**A**). Veja que os acordes maiores de acordo com a Tabela 5 são formados pela: Tônica (T), Terça Maior (3^aM) e Quinta Justa (5^aJ). Observe que como se trata do acorde Lá Maior (**A**) temos a Tônica sendo a nota Lá. A Terça Maior **3^aM**, de acordo com a Tabela 3, é formada por 4 semitons acima da nota lá, portanto teremos a nota dó elevada meio tom acima, ou seja, *C#* e a Quinta Justa que de acordo com a Tabela 3 são 7 semitons acima de lá, ou seja, E. Portanto para conseguirmos tocar o acorde de **A** em qualquer instrumento basta identificarmos as notas A, *C#*, E e tocá-las juntas. Seguindo a mesma linha de raciocínio, vamos identificar os acordes de **A** de acordo com a classificação das Tabelas 5 e 6.

- Lá Maior -> T + 3^aM + 5^aJ -> A + *C#* + E
- Lá Menor -> T + 3^am + 5^aJ -> A + C + E
- Lá Diminuto -> T + 3^am + 5^aD -> A + C + *D#*
- Lá Aumentado -> T + 3^aM + 5^aA -> A + *C#* + F
- Lá maior com Sétima Maior -> T + 3^aM + 5^aJ + 7^aM -> A + *C#* + E + *G#*
- Lá Maior com Sétima Menor -> T + 3^aM + 5^aJ + 7^am -> A + *C#* + E + G
- Lá Menor com Sétima Menor -> T + 3^am + 5^aJ + 7^am -> A + C + E + G
- Lá Meio Diminuto -> T + 3^am + 5^aD + 7^am -> A + C + *D#* + G
- Lá Suspenso com Sétima Menor -> T + 4^aJ + 5^aJ + 7^am -> A + D + E + G
- Lá Maior com Sexta Maior -> T + 3^aM + 5^aJ + 6^aM -> A + *C#* + E + *F#*
- Lá Menor com Sexta Maior -> T + 3^am + 5^aJ -> A + C + E + *F#*

- Lá Aumentado Tétrade -> T + 3ªM + 5ªA + 7ªM -> A + C# + F + G#

2.2.2.3 Inversão de Acordes

A primeira nota, como já sabemos, é a nota tônica, veremos mais adiante que sempre a tônica é a nota mais grave, fisicamente isso significa que a frequência desta nota é a mais baixa do acorde em questão. Sabemos que o acorde de **C** é formado pelas notas C, E, G. Quando falamos em inversão do acorde é fazer com que outra nota que constitui o acorde em questão (E ou G) seja a nota mais grave. As inversões costumam ser de três tipos de acordo com (CLUB, 2021):

- **1ª Inversão: terça no baixo:** Nesta inversão a terça do acorde que aparece como nota mais grave. Por exemplo: A/C#, A7/C#.
- **2ª Inversão: quinta no baixo:** Nesta inversão a quinta do acorde que aparece como nota mais grave. Por exemplo: G/D, G7M/D.
- **3ª Inversão: sétima no baixo:** Nesta inversão a sétima do acorde que aparece como nota mais grave. Por exemplo: D/C#, D/C.

Com base nas definições fornecidas na presente seção e também em algumas classificações que fizemos, é importante notar que para classificarmos um acorde precisamos somente de duas informações: A tônica e seu intervalo em relação as demais notas. Então observe a figura 4.

Figura 4: Acorde de Lá Maior



Fonte: O autor

Note que temos três notas, portanto se trata de um acorde tríade. Observe também que como a tônica é Lá, temos que só descobrir a relação desta com as demais notas C# e E. Veja que entre A e C# temos 4 semitons, entre A e E temos 7 semitons. Analisando a Tabela 3 de intervalos harmônicos vemos que C# é a terça maior de A,

como também vemos que E é a quinta justa de A . Portanto, pela Tabela 5 temos um acorde Maior, como a tônica é Lá. Classificamos o acorde sendo o Lá Maior.

Observe que dado notas conseguimos classificar acordes baseado nas tabelas 3 e 5. Veja que cada nota corresponde a uma frequência de vibração específica. Sendo a tônica a nota principal (nota mais grave), servirá de base para relacionarmos com as outras frequências detectadas e que por fim classificaremos o acorde. Mas, por outro lado, quando a música é transmitida para o computador, o sinal vem em termos de amplitude e tempo. Teríamos que ter um sinal em termos de amplitude e frequência. Para isso vamos entender como conseguiremos fazer isso utilizando **Transformada de Fourier**. Uma vez identificadas as amplitudes e frequências em um certo tempo. Identificaremos as notas que são correspondentes a estas frequências e as relações entre elas partindo da nota de frequência mais baixa (tônica). As relações entre estas frequências, pode ser obtidas através da **Equação da escala temperada**.

2.3 EQUAÇÃO DA ESCALA TEMPERADA

A equação da Escala Temperada foi uma reformulação de uma das descobertas de Pitágoras que de acordo com (PIRES, 2008) sua teoria de números o levou a fazer experimentos com notas musicais utilizando um monocórdio, instrumento musical composto por uma única corda. Em virtude disso, descobriu-se que cordas ao vibrarem emitiam sons em função dos seus comprimentos. Assim foram estabelecidas relações fracionárias para essas vibrações que permitiam gerar notas dando origem a Escala Pitagórica. Então por exemplo, uma corda com comprimento l , ao vibrar, produz uma nota N_i , sendo i um número natural qualquer, com frequência f_0 . Se pressionarmos a metade desta corda, ou seja em $\frac{l}{2}$ ao vibrá-la, teremos uma nota à oitava acima ou seja, N_{i+1} . Fisicamente N_{i+1} possui o dobro da frequência de N_i que seria $2f_0$. Portanto o estudo de Pitágoras, de acordo com (JESUS; SOUZA, 2020), contribuiu para a descoberta das 12 notas musicais e para cada uma delas existe uma fração da corda que a produz.

Ao decorrer do tempo o uso da escala pitagórica se tornava muito restrito, pois não favorecia certos procedimentos como a troca da tonalidade das músicas, uma vez que os intervalos entre as notas variavam de acordo com o tom escolhido, produzindo um efeito de desafinação. Em virtude disso a solução foi adotar um sistema com doze semitons igualmente distribuídos pela oitava. Chamamos essa escala de **Escala Temperada**. Partindo das 12 notas musicais que foram apresentadas no capítulo anterior, conseguimos estabelecer uma relação entre as notas e suas frequências, de modo que o espaço entre cada uma delas é igual a 1 semitom. Observe a figura abaixo:

Figura 5: Notas e Frequências

	0	1	2	3	4	5	6	7	8	9	10
C	16.351	32.703	65.405	130.81	261.62	523.24	1046.48	2092.96	4185.92	8371.84	16743.68
C#/Db	17.325	34.650	69.300	138.60	277.20	554.40	1108.80	2217.60	4435.20	8870.40	17740.80
D	18.350	36.700	73.400	146.80	293.60	587.20	1174.40	2348.80	4697.60	9395.20	18790.40
D#/Eb	19.450	38.900	77.800	155.60	311.20	622.40	1244.80	2489.60	4979.20	9958.40	19916.80
E	20.600	41.200	82.400	164.80	329.60	659.20	1318.40	2636.80	5273.60	10547.20	21094.40
F	21.825	43.650	87.300	174.60	349.20	698.40	1396.80	2793.60	5587.20	11174.40	22348.80
F#/Gb	23.125	46.250	92.500	185.00	370.00	740.00	1480.00	2960.00	5920.00	11840.00	23680.00
G	24.500	49.000	98.000	196.00	392.00	784.00	1568.00	3136.00	6272.00	12544.00	25088.00
G#/Ab	25.962	51.925	103.850	207.70	415.40	830.80	1661.60	3323.20	6646.40	13292.80	26585.60
A	27.500	55.000	110.000	220.00	440.00	880.00	1760.00	3520.00	7040.00	14080.00	28160.00
A#/Bb	29.137	58.275	116.550	233.10	466.20	932.40	1864.80	3729.60	7459.20	14918.40	29836.80
B	30.863	61.725	123.450	246.90	493.80	987.60	1975.20	3950.40	7900.80	15801.60	31603.20

Fonte: O autor

Note que na Figura temos os números das notas dispostos de 0 a 10. Logo, por exemplo, entre C_3 que possui uma frequência de $130.81Hz$ e C_4 que possui uma frequência de $261.62Hz$ a frequência praticamente dobra de C_3 para C_4 . Portanto diremos que subimos uma oitava em C . Podemos assim, estabelecer relações entre as frequências das notas e as notas. Note também que se contarmos a quantidade de semitons de C_3 a C_4 teremos exatamente 12 semitons (uma oitava) como visto na Tabela 3.

Vamos analisar as notas que constituem a escala de C_3 até C_4 e suas respectivas frequências de acordo com a tabela abaixo:

Distribuição de frequências												
C_3	$C\#_3$	D_3	$D\#_3$	E_3	F_3	$F\#_3$	G_3	$G\#_3$	A_3	$A\#_3$	B_3	C_4
f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}

Tabela 7: Distribuição de Frequências na escala de C_3 a C_4

O espaçamento entre as notas (semitons) podem ser modelados matematicamente utilizando progressões. Observe que sempre as frequências dobram quando completam 12 semitons, pela Tabela 7 $f_{12} = 2f_0$. Então tentaremos utilizar uma progressão geométrica pelo fato da frequência dobrar.

$$f_n = f_0 * q^n \quad (15)$$

Em que n , neste caso é a quantidade de semitons que separam duas notas, q é a

razão da progressão, f_n e f_0 é a frequência que determinadas notas possuem. O que precisamos descobrir é a razão q para conseguirmos usar a equação 15. Lembre que as frequências sempre dobram quando se atinge 1 oitava. Logo para $n = 12$ temos que $f_n = 2f_0$. Substituindo essas relações em 15, temos que:

$$q = \sqrt[12]{2} \quad (16)$$

Substituindo 16 em 15. Teremos a equação 17

$$f_n = f_0(\sqrt[12]{2})^n \quad (17)$$

Vamos supor que pressionamos a nota G_1 que de acordo com a Figura 5 possui uma frequência de $49.000Hz$. Em seguida queremos saber qual a frequência da nota que corresponde a 3 semitons acima de G_1 . Como são 3 semitons teremos $n = 3$ e a frequência da nota G_1 é $49.000Hz$ logo $f_0 = 49.000Hz$. Portanto, substituindo em 17 resulta que f_3 é dada por 18.

$$f_3 = 58.271Hz \quad (18)$$

Analisando novamente a Figura 5 a frequência f_3 é aproximadamente $58.275Hz$, ou seja temos a nota $A\#_1$, que de fato são 3 semitons acima da nota G_1 . Portanto é importante lembrar que de acordo com a Tabela 3, como temos 3 semitons o $A\#_1$ é a Terça menor de G_1 .

Com base no exemplo acima, vamos tentar utilizar a equação 17 para identificar qual é o acorde uma vez fornecidas as frequências. Sabemos, pelo capítulo anterior que um acorde é uma junção de notas e essas notas, fisicamente, são interpretadas como uma onda individual que vibra em determinada frequência. Portanto a nota $A\#_1$ é uma onda que vibra com uma frequência de aproximadamente $58.271Hz$. Logo é fácil concluir que um acorde são ondas superpostas em um determinado tempo. Através deste raciocínio faremos um exemplo logo abaixo relacionado a um dos trabalhos que o algoritmo proposto irá executar.

Dadas as seguintes frequências em um determinado tempo t de uma música qualquer, identificar o acorde.

$$130.81Hz - 164.80Hz - 196.00Hz$$

Repare que temos três frequências, portanto é um acorde formado por três notas. Logo se trata de um tríade. A frequência mais baixa que nós chamamos de tônica é $130.81Hz$. Precisamos descobrir qual a relação, número de semitons, dessa frequência com as outras. Utilizando a equação 17 precisamos agora isolar o n , uma vez que queremos descobrir a quantidade de semitons que existem entre duas frequências

fornecidas. Portanto temos a seguinte equação:

$$n = \frac{12}{\ln 2} (\ln f_n - \ln f_0) \quad (19)$$

Substituindo em 19 o valor de f_0 que é $130.81Hz$, frequência mais baixa (tônica), queremos saber quantos semitons existem até a nota com a frequência de $164.80Hz$ que seria no caso f_n . Substituindo esses valores temos que:

$$n \simeq 4$$

Portanto da nota com frequência de $130.81Hz$ até a nota com frequência de $164.80Hz$ temos 4 semitons. Por outro lado, podemos descobrir quantos semitons existem da nota com frequência de $130.81Hz$ até a nota com frequência de $196.00Hz$. Substituindo estes valores em 19:

$$n = \frac{12}{\ln 2} (\ln(196.00) - \ln(130.81))$$

$$n \simeq 7$$

temos 7 semitons entre a menor frequência e a maior frequência dada.

De acordo com a Tabela 3 temos um intervalo de Terça Maior, entre $130.81Hz$ e $164.80Hz$ pois temos 4 semitons e um intervalo de Quinta Justa, entre $130.81Hz$ e $196.00Hz$, pois temos 7 semitons. Portanto, de acordo com a Tabela 5 temos um acorde maior. Basta identificarmos agora, a nota correspondente a frequência mais baixa que é $130.81Hz$ na Figura 5 e classificarmos o acorde. Logo como $130.81Hz$ corresponde a nota C_3 teremos portanto o acorde de C - **dó maior**.

Veja que uma vez identificadas as frequências em um determinado tempo de uma música, conseguimos classificar os acordes utilizando a **Equação da Escala Temperada**. Todavia, quando transcrevemos uma música para dispositivos digitais a informação é carregada em termos de tempo e amplitude. Portanto, necessitamos de algo além disso. O que queremos é saber quais as frequências envolvidas em determinado tempo da música. Para isso, vamos utilizar uma ferramenta matemática chamada **Transformada discreta de Fourier**. Assim, quando temos um sinal em um determinado tempo e aplicamos a transformada de Fourier neste sinal, teremos as frequências que formam a música naquele determinado tempo. Para fazer isso, temos que entender algumas noções sobre processamento digital de sinais uma vez que a música ao ser importada para o computador ela é transformada em um sinal digital. Veremos mais sobre este tópico no próximo capítulo.

3 TÓPICOS SOBRE PROCESSAMENTO DE SINAIS

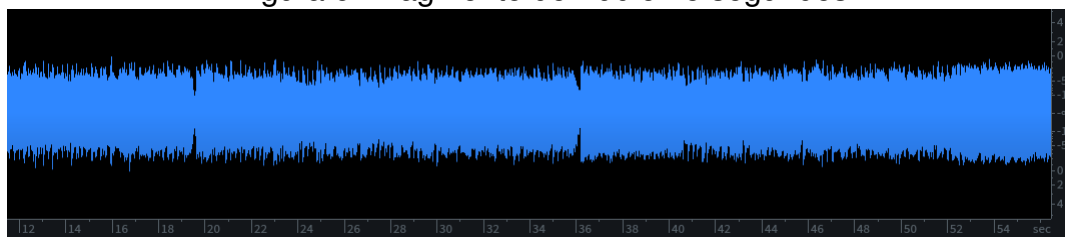
Neste capítulo, alguns conceitos relacionados ao processamento digital de sinais serão expostos e, com isso, conseguiremos entender os fundamentos do algoritmo proposto. Não pretendemos expor detalhadamente a teoria de sinais digitais, para maiores detalhes sobre a teoria consulte (LATHI, 2007) e (OPPENHEIM; WILLSKY, 2010).

3.1 PROCESSAMENTO DE SINAIS

O sinal de áudio, quando processado em um sistema digital é codificado como um vetor de amostras, segundo [http : //wiki.inf.ufpr.br/maziero/doku.php?id = prog2 : processamento_de_audio](http://wiki.inf.ufpr.br/maziero/doku.php?id=prog2:processamento_de_audio) , onde cada amostra corresponde à amplitude do sinal sonoro em um instante de tempo.

Suponha o fragmento do sinal de uma música mostrado na Figura 6 com um comprimento de 45 segundos.

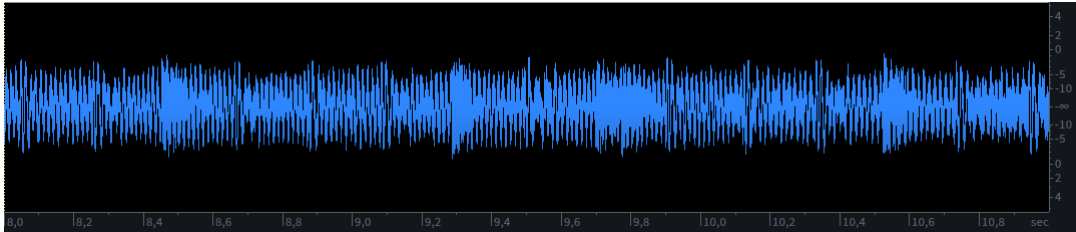
Figura 6: Fragmento de Áudio 45 segundos



Fonte: O autor

Para fixarmos a ideia de amostragem, vamos diminuir o tamanho do fragmento de música para 3 segundos e observá-lo Figura 7

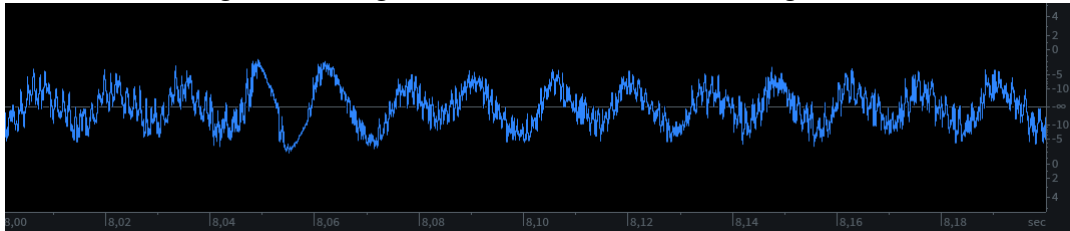
Figura 7: Fragmento de Áudio 3 segundos



Fonte: O autor

Note que a na figura acima já conseguimos perceber certa discretização do sinal. Então, diminuindo o mesmo fragmento para um tamanho de 200 milissegundos temos o sinal da Figura 8.

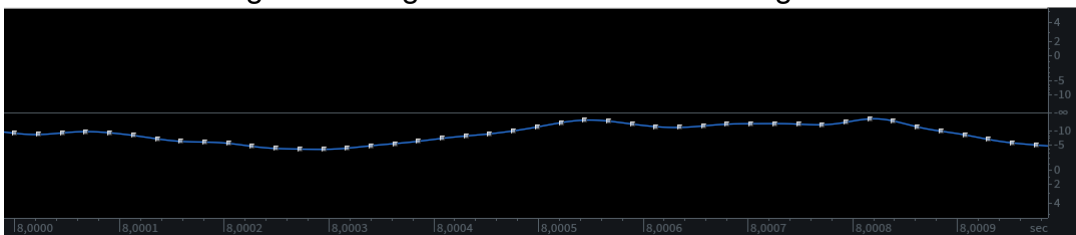
Figura 8: Fragmento de Áudio 200 milissegundos



Fonte: O autor

Finalmente, diminuindo o mesmo fragmento para um tamanho de 1 milissegundo, veremos claramente o sinal discretizado em amostras representadas pelos pontos claros.

Figura 9: Fragmento de Áudio 1 milissegundo



Fonte: O autor

Observe na Figura 6 que o sinal em questão parecia ser algo contínuo. Em contrapartida, a Figura 9 mostra que na verdade temos pontos distribuídos durante a "assinatura do sinal" de tal forma que para nossos sentidos este sinal pareça ser contínuo ao escutá-lo. Tudo isso é justificado pelo número de vezes em que se realiza a amostragem em uma unidade de tempo, ou seja, pela **taxa de amostragem** que geralmente é medida em Hertz. Um CD, por exemplo possui uma taxa de amostragem de áudio no valor de $44100Hz$ isso quer dizer que a cada segundo de som são tomadas

44100 medidas de variação de voltagem do sinal. Logo, quanto maior for essa taxa de amostragem, o sinal digital se aproxima de um sinal real (contínuo) e, portanto, conseguimos ouvir os sinais de áudio em sistemas digitais de forma contínua (sem pausas).

As frequências das combinações de notas musicais não são harmonicamente relacionadas e, portanto, utilizaremos a Transformada de Fourier para representar sinais tanto periódicos, com frequências harmonicamente relacionadas, quanto aperiódicos com frequências que não estão harmonicamente relacionadas. Como essas funções aperiódicas, possuem frequências infinitesimalmente próximas segundo (OPPENHEIM; WILLSKY, 2010), é possível representar um sinal aperiódico por um sinal periódico de período $T \rightarrow \infty$. Para ambos casos, usaremos a transformada de Fourier que é definida por

$$x(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt, \quad (20)$$

e sua inversa é dada por

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} x(\omega)e^{j\omega t} d\omega, \quad (21)$$

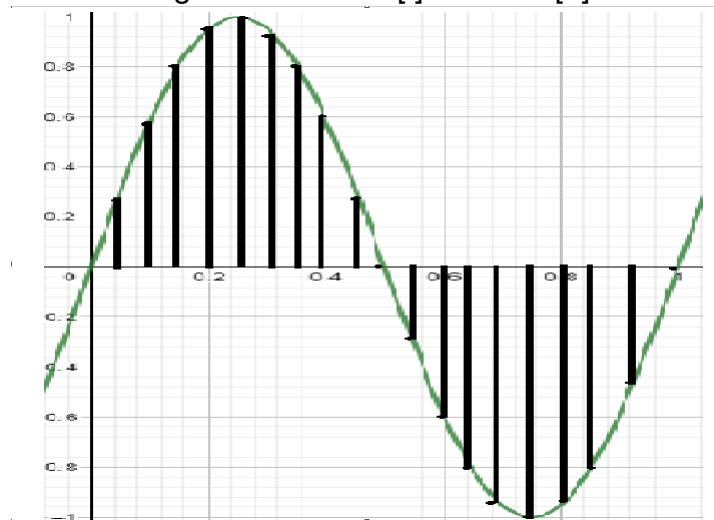
onde $x(\omega)$ é a transformada de Fourier em Tempo Contínuo e j representa a unidade imaginária.

É em virtude da discretização do sinal que utilizaremos, com o intuito de descobrir as frequências, a Transformada de Fourier em Tempo Discreto. Um sinal armazenado em um sistema digital é amostrado em termos de vários pontos como mostrado na Figura 9. Matematicamente, podemos representar essa discretização de um sinal x em termos de uma sequência finita de valores, $x[0], x[1], \dots, x[L-1]$, desde que existam L amostras de sinais no tempo. É em virtude disso, portanto, que nós não precisamos de um número infinito de frequências para representar o sinal adequadamente. Um número necessário de frequências seria da ordem de $N \approx 2L$. (FESSLER, 2011)

Como mostrado na Figura 9, os sinais que trataremos, foram discretizados por um sistema digital utilizando uma certa taxa de amostragem que é dada por $t_s = \frac{1}{F_s}$, onde F_s é a taxa de amostragem ou frequência de amostragem. Para ser amostrado, o sinal $x(t)$ é multiplicado por impulsos (deltas de Dirac) de período igual a t_s , ou seja:

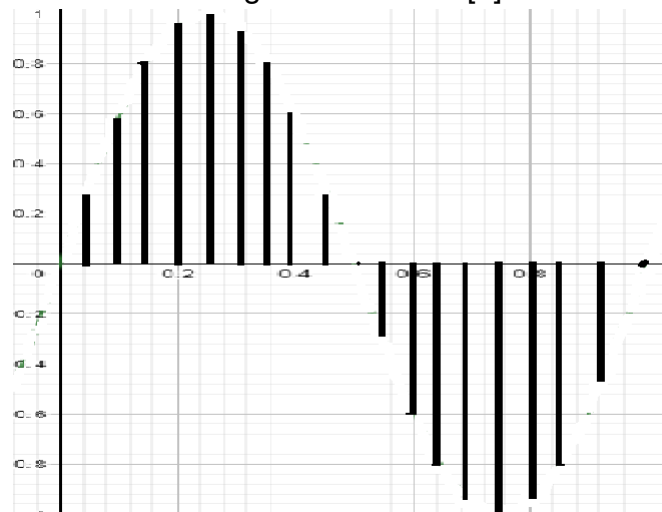
$$x[n] = x(t) \cdot \sum_{n=-\infty}^{\infty} \delta(t - nt_s) \quad (22)$$

Para exemplificar, seja um sinal descrito por $x(t) = \sin(2\pi t)$. Observe que como $T = \frac{2\pi}{\omega_0}$, temos que $\omega_0 = 2\pi$ se $T = 1s$. Suponha que queremos discretizar esta função em uma frequência de amostragem de $20Hz$. Neste caso, no intervalo de 1 segundo temos 20 amostras ($x_0, x_1, x_2 \dots x_{19}$).

Figura 10: Sinal $x[t]$ e Sinal $x[n]$ 

Fonte: O autor

Perceba que na Figura 10 as linhas verticais pretas representam a discretização do sinal. Por fim, eliminando a linha verde, teremos a Figura 11 que mostra o sinal discretizado.

Figura 11: Sinal $x[n]$ 

Fonte: O autor

Pelo Teorema de Nyquist, que pode ser encontrado em (OPPENHEIM; WILLSKY, 2010) e (LATHI, 2007), a frequência de Amostragem deve obedecer uma relação com a frequência máxima do sinal a ser amostrado que denotaremos por F_M . Logo:

$$F_s = 2F_M. \quad (23)$$

Um arquivo de áudio com qualidade de um MP3, possui uma frequência máxima de 20050Hz . Pelo Teorema de Nyquist a frequência de amostragem deve ser no mínimo

igual a $40100Hz$. Como essa frequência é alta, isso poderia exigir uma grande capacidade de armazenamento. Para contornar essa situação, vamos analisar o áudio criando intervalos de 1 em 1 segundo de música. E assim, amostraremos o áudio em uma frequência de 44100 amostras por segundo. Uma vez tendo o sinal discretizado, aplicaremos a transformada rápida de Fourier, disponível em uma das bibliotecas da linguagem de programação *python*. Assim, o resultado vai permitir obtermos os impulsos associados as frequências que constituem o sinal estudado.

3.2 FILTROS

Segundo (LATHI, 2007) uma das necessidades da aplicação de filtros é na separação de um sinal contendo informações de frequências que são indesejáveis, ruídos e produtos de distorção. Neste trabalho abordaremos os filtros digitais do tipo resposta ao impulso com duração finita, mais precisamente, vamos abordar sobre os filtros passa baixas ideais e por conseguinte; métodos de janelamento. Também falaremos sobre os filtros de medianas que foram utilizados no presente trabalho para separar os instrumentos percussivos dos harmônicos em um dado sinal. Os filtros lineares aplicam uma operação matemática linear a um determinado sinal de entrada. São muito utilizados para retirar ou selecionar frequências não desejadas ou desejadas, respectivamente, desses sinais. Os filtros lineares podem se dividir nas seguintes classes:

- **Resposta de impulso infinita (IIR)** Quando a resposta ao impulso possui uma duração infinita. Devido ao número infinito de termos envolvidos, o uso desses tipos de filtro, em determinadas aplicações, pode ocasionar alguma dificuldade computacional;
- **Resposta de impulso finita (FIR)** A resposta ao impulso finita $h(n)$ tem valores não nulos sobre um intervalo de tempo finito e zero fora deste intervalo. Existem vários tipos de filtros lineares do tipo **FIR**. Abordaremos alguns deles a seguir:

Filtro passa-baixas: Dada uma frequência de corte f_c , qualquer frequência **acima** desta será rejeitada. Estes filtros são representados pela Função 24

$$H(f) = \begin{cases} Ke^{-j\omega t_d}, & |f| \leq f_c \\ 0, & |f| > f_c \end{cases} \quad (24)$$

Filtro passa-altas: Dada uma frequência de corte f_c , qualquer frequência

abaixo dessa será rejeitada. Estes filtros são representados pela Função 25:

$$H(f) = \begin{cases} K e^{-j\omega t_d}, & f_c \leq |f| \leq \infty \\ 0, & |f| < f_c \end{cases} \quad (25)$$

Filtro passa-bandas: Pode ser entendido como uma junção dos filtros passa-baixas e passa-altas. Portanto dado um intervalo tal que seus extremos são duas frequências f_b e f_a . Serão consideradas as frequências dentro deste intervalo. Estes filtros são representados pela Função 26

$$H(f) = \begin{cases} K e^{-j\omega t_d}, & f_b \leq |f| \leq f_a \\ 0, & |f| < f_b \text{ ou } |f| > f_a \end{cases} \quad (26)$$

Filtro rejeita-faixa: Esses filtros seriam o oposto dos filtros passa-banda. Portanto teríamos frequências rejeitadas dentro do intervalo entre f_b e f_a . A Função 27 representa estes tipos de filtros.

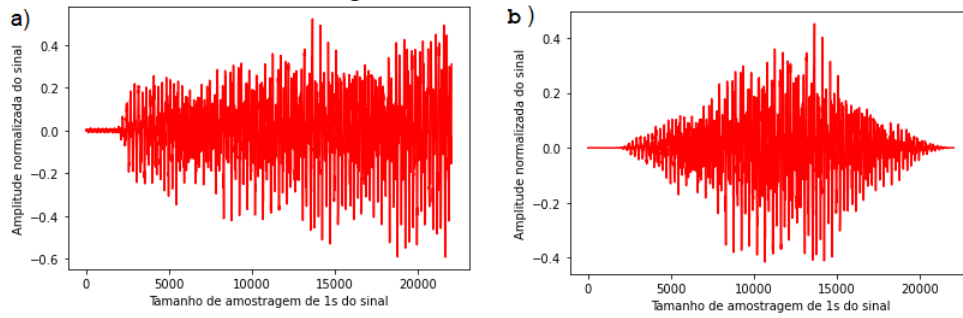
$$H(f) = \begin{cases} K e^{-j\omega t_d}, & |f| < f_b \text{ ou } |f| > f_a \\ 0, & f_b \leq |f| \leq f_a \end{cases} \quad (27)$$

De grande utilidade é o janelamento, a função de Hann é usada em processamento digital de sinais como uma janela utilizada para selecionar um subconjunto de uma série de amostras visando a aplicação da Transformada de Fourier ou outros métodos. Matematicamente, a função de Hann é classificada como de janelamento discreto representada pela função 28.

$$\omega(n) = 0.5 - 0.5 \cos\left(\frac{2\pi n}{M-1}\right), \quad 0 \leq n \leq M-1 \quad (28)$$

Essa janela foi nomeada, de acordo com (HARRIS F., 1976), em homenagem a um meteorologista austríaco chamado *Julius von Hann* e tem como objetivo suavizar determinados tipos de sinais. Na Figura 12 podemos observar como se comporta a janela de Hann em um sinal que contém a informação do acorde $F\#$.

Figura 12: Janela de Hann



Fonte: O autor

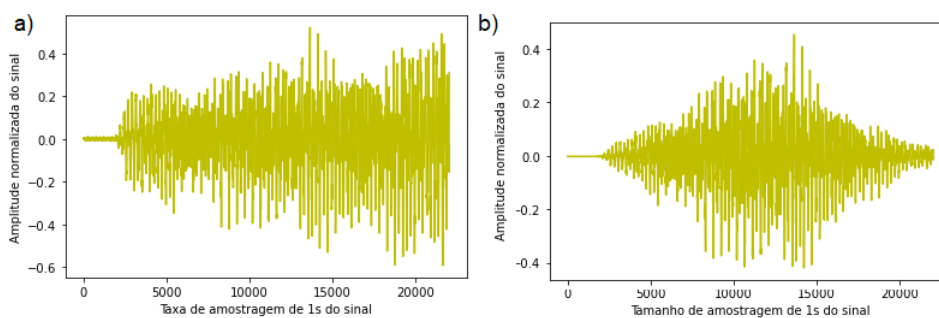
Em a) temos um fragmento de um sinal equivalente a 1 segundo e em b) a janela de Hann foi aplicada neste sinal. Em b), observe que o sinal foi empacotado.

A Janela de Hamming é semelhante a janela de Hann. Abaixo podemos definir a função que a representa.

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{M-1}\right), \quad 0 \leq n \leq M-1 \quad . \quad (29)$$

Tal janela é muito utilizada em processamento de sinal para valores de suavização e é muito reconhecida por conseguir suavizar descontinuidades no início e no final de um sinal amostrado por isso ela pode ser chamada também de função de redução gradual. Observe a Figura 13 onde mostra a janela de Hamming sendo aplicada no mesmo sinal do caso anterior.

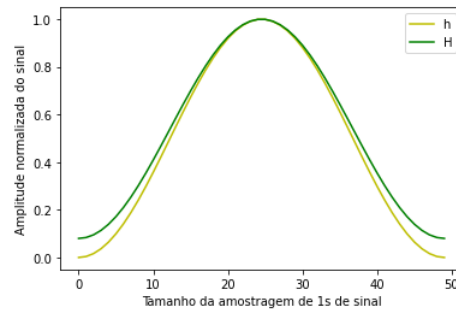
Figura 13: Janela de Hamming



Fonte: O autor

Essa janela, diferentemente da janela anterior é um pouco menor. Conseguimos observar esse fato na Figura 14.

Figura 14: Comparação entre as janelas



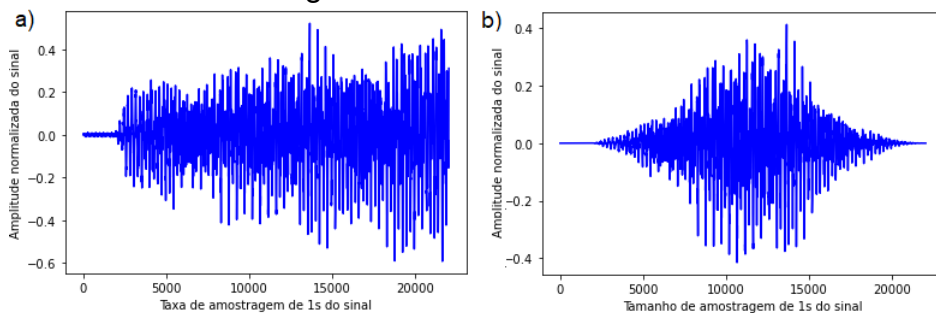
Fonte: O autor

Observe que nas extremidades inferiores do gráfico claramente mostra que a janela de Hamming é menor que a janela de Hann. A Janela de Blackman é definida de acordo com a Função 30 dada por

$$w(n) = 0.42 - 0.5 \cos\left(\frac{2\pi n}{M}\right) + 0.08 \cos\left(\frac{4\pi n}{M}\right). \quad (30)$$

Esta janela é também utilizada para valores de suavização. Conhecida como apodização ou função de redução gradual. E diferencia das outras por ter um afilamento quase ótimo. Na Figura 15 podemos notar como a função de Blackman atua no sinal dos casos anteriores. Observa-se que a janela de Blackman, representada pela cor vermelha no gráfico da figura 16 é, de fato, mais afilada que as outras duas janelas. A Janela Triangular, como o nome indica possui um formato triangular. Também utilizada para suavizar valores de sinais. Abaixo podemos observar a função que a representa.

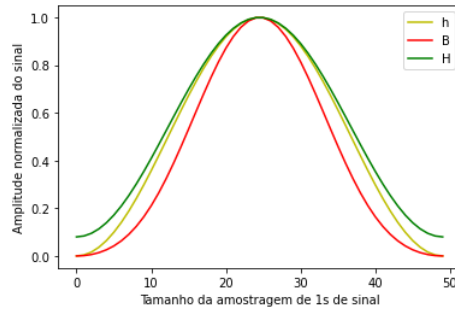
Figura 15: Janela de Blackman



Fonte: O autor

Note, na Figura 16, a comparação entre a janela de Blackman e as outras duas janelas discutida anteriormente.

Figura 16: Comparação entre as janelas



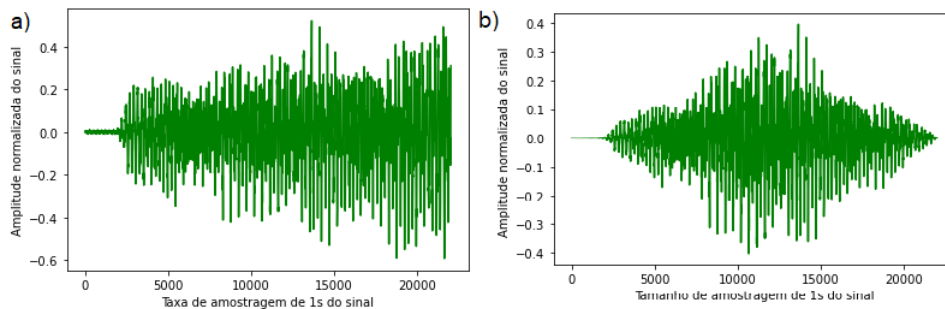
Fonte: O autor

A Janela Triangular, como o nome indica possui um formato triangular. Também utilizada para suavizar valores de sinais. Abaixo podemos observar a função que a representa

$$w(n) = 1 - \left| \frac{n - \frac{N}{2}}{\frac{L}{2}} \right|, \quad (31)$$

onde L representa o tamanho da janela e N é o tamanho do sinal. Geralmente $L = N$ para a janela se ajustar ao sinal. Observe a janela sendo aplicada no mesmo sinal dos casos anteriores na Figura 17. Observe que de fato a janela possui um formato triangular e ao ser aplicada a um sinal o torna em um formato aproximado igual ao da janela.

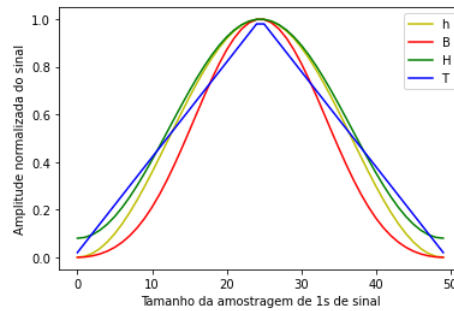
Figura 17: Janela Triangular



Fonte: O autor

Na Figura 18, para efeitos de comparação mostra-se as 4 janelas com diferentes cores. A janela triangular, nessa figura por exemplo, representada pela cor azul, possui o formato triangular em relação às outras Janelas. Observe que a janela de Blackman, representada pela cor vermelha, é capaz de suavizar o sinal em suas bordas e deixá-lo mais afilado.

Figura 18: Comparação entre as janelas



Fonte: O autor

3.3 CONVOLUÇÃO DE SINAIS

Um outro tópico importante que será utilizado no presente trabalho é a convolução de sinais. A operação de convolução, de acordo com (SANTO, 2008), (OPPENHEIM; WILLSKY, 2010) e (LATHI, 2007), é definida em sinais discretos pela equação 32.

$$y[n] = \sum_{k=-\infty}^{+\infty} x[k]\delta[n - k] \quad (32)$$

A Convolução é definida em sistemas Lineares e Invariantes no tempo, podemos estabelecer que:

- A resposta de um sistema em virtude de uma entrada a impulso possuirá também uma resposta na forma de impulso;
- A resposta de um impulso que esteja deslocado no tempo levará a uma resposta também deslocada no tempo, devia a característica da invariância no tempo;
- Se o impulso for modificado por algum fator de escala, a resposta também será afetada por este mesmo fator;
- A soma de um número de entradas de impulsos modificados por um fator de escala é igual à soma das respostas aos impulsos modificados pelo mesmo fator de escala, uma vez que a operação de multiplicação é linear.

A operação de convolução pode ser representada pelo símbolo *, como denotado na equação 33.

$$y[n] = x[n] * y[n]. \quad (33)$$

As ferramentas expostas, de maneira simplificada, no presente capítulo serão aplicadas no capítulo 5 na data base a ser utilizada. No próximo capítulo são discutidos aspectos importantes das redes neurais.

4 TÓPICOS SOBRE REDES NEURAIS

No presente capítulo descrevemos os fundamentos de redes neurais artificiais, as quais serão utilizadas no presente trabalho para a classificação dos acordes que constituem uma determinada música. Para isso, faremos uma descrição simples sobre a matemática para ficar claro o processo de aprendizagem de uma rede.

4.1 REDES NEURAIS ARTIFICIAIS

Segundo (GRUBLER, 2018), rede neural artificial pode ser entendida como uma estrutura complexa interligada por elementos de processos simples que são os neurônios, estes são capazes de realizar cálculos para processamento de dados e representação do conhecimento. O marco do surgimento das redes neurais foram três publicações, de acordo com (BAXTER et al., 2008), essas foram em ordem cronológica desenvolvidas por: McCulloch e Pitts (1943), Hebb (1949) e Roseblatt (1958) onde a primeira introduziu o primeiro modelo de redes neurais simulando "máquinas", o segundo fala sobre o modelo básico de rede de auto-organização e por fim o modelo Perceptron de aprendizado supervisionado. Atualmente há diversas topologias das redes neurais artificiais (RNA's) onde tentam resolver problemas tais como:

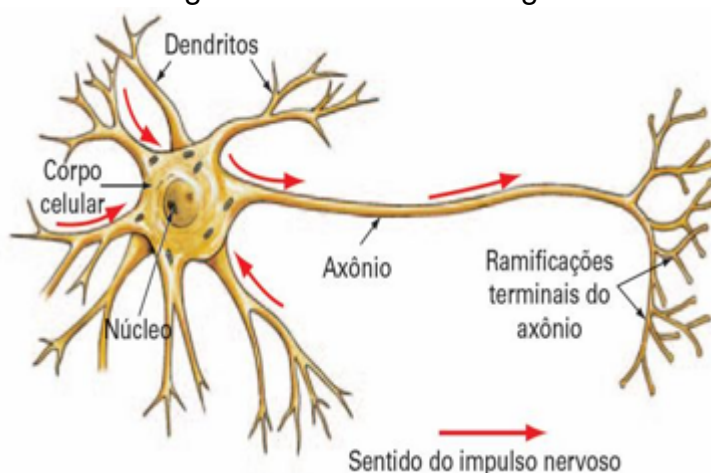
- Processamento de linguagem natural
- Reconhecimento de falas, músicas e imagens
- Previsões de valores
- Previsão de um determinado negócio ser bem sucedido

Para conseguirmos entender as redes neurais artificiais, vamos entender primeiramente no que esta técnica está baseada. Sabendo que uma rede neural artificial pode ser entendida como vários neurônios artificiais interligados onde há troca de informações entre os mesmos, precisamos estudar o neurônio artificial uma vez que é a menor unidade de uma RNA. Mas para isso vamos fazer uma breve explicação de como o neurônio biológico funciona.

4.2 NEURÔNIO BIOLÓGICO

O Neurônio é a unidade básica do cérebro humano que tem por função transmitir informação para outros neurônios e é constituído por três partes: Corpo Celular, Dendritos e Axônios as quais podem ser vistas na Figura 19.

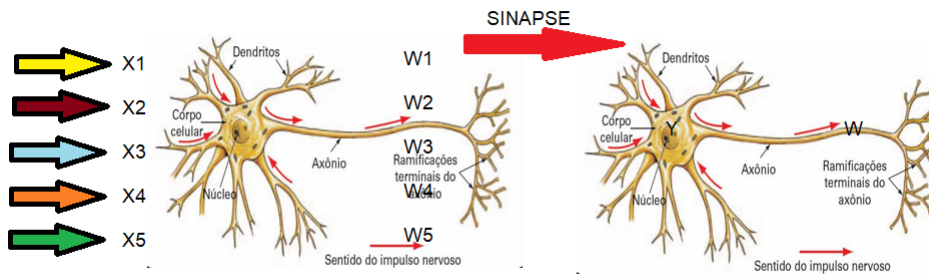
Figura 19: Neurônio Biológico



Fonte: (ACADEMY, 2021)

O núcleo do Neurônio se localiza no corpo celular que também é conhecido como Soma. Esse nome será bastante interessante mais à frente para o entendimento do neurônio artificial. Os dendritos são as ramificações que estão presentes no corpo celular e o sentido do impulso nervoso ocorre dos dendritos aos axônios. O corpo celular tem por função carregar informações vitais sobre suas características, além de ser constituído por um plasma que contém as substâncias moleculares para o seu funcionamento. A comunicação se dá através do impulso nervoso. Como estamos tratando de uma rede de neurônios o impulso é captado inicialmente pelos dendritos que repassam para o corpo celular através do axônio. Este recebe sinais a partir do corpo da célula e os transporta para os dendritos e assim repassam para os dendritos de outros neurônios vizinhos através de um processo conhecido sinapse. A característica comum que há tanto no neurônio biológico quanto no artificial é que ambos possuem dendritos e axônios nos quais a comunicação também se dá por um processo de sinapse. Observe a Figura 20.

Figura 20: Sinapse Biológica

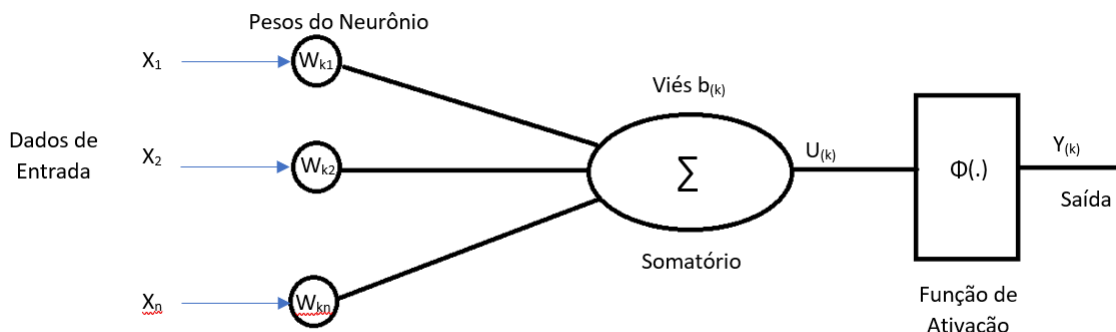


Fonte: Adaptado de (ACADEMY, 2021)

Note que na Figura acima temos na extremidade esquerda do neurônio dados de entrada de outros neurônios que constituem o sistema nervoso identificados por X_1, X_2, X_3, X_4, X_5 e W_1, W_2, W_3, W_4, W_5 representam as forças sinápticas. Tanto a rede neural biológica quanto artificial conseguem ajustar as amplitudes das forças sinápticas em uma série de camadas. Na próxima seção falaremos sobre o perceptron que é uma rede neural constituída de um único neurônio.

4.3 REDE DE UM NEURÔNIO

O modelo *Perceptron* foi criado por Frank Rosenblatt, inspirado nos trabalhos de Walter Pitts e Warren Sturgis McCulloch. De acordo com (GRUBLER, 2018) é um dos modelos mais antigos e lida somente com um neurônio artificial, classificando resultados de forma linear. A Figura 21 mostra um neurônio artificial que é o próprio modelo *Perceptron*.

Figura 21: Modelo *Perceptron*

Fonte: O autor

Observe que na Figura temos os dados de entrada representados por X_n que são ponderados por pesos sinápticos, determinados durante o treinamento da rede e são

denotados por W_{kn} onde k é um índice que denota uma camada anterior da rede e n denota a posição do peso para um respectivo dado de entrada. Após este processo, os resultados ponderados passam por um somatório formando-se um conjunto de entrada. Matematicamente, este processo pode ser modelado pela equação 34.

$$U_k = \sum w_{kn} * x_n \quad (34)$$

Após essa soma ser calculada este valor é submetido a uma função de ativação que será responsável por ativar ou não ativar o Neurônio. Se o valor dessa função de ativação aplicada no resultado do somatório for maior que um viés de ativação b_k a informação chega na saída do neurônio, caso contrário a informação é retida no mesmo. O valor do viés de ativação é definido durante a etapa de treinamento da rede. A equação 35 modela o que foi falado anteriormente.

$$\phi(U_k) = \begin{cases} 1, & U_k \geq b_k \\ 0, & U_k < b_k \end{cases} \quad (35)$$

Podemos reescrever a equação 35 da seguinte maneira:

$$\phi(U_k - b_k) = \begin{cases} 1, & U_k - b_k \geq 0 \\ 0, & U_k - b_k < 0 \end{cases} \quad (36)$$

Onde $\phi(U_k - b_k)$ é a função de ativação aplicada no valor do somatório $U_k - b_k$. Existem vários tipos de funções de ativação e falaremos sobre elas na seção que se segue.

4.3.1 Função de Ativação

Observe na Figura 21 que a última etapa do processo é o valor do somatório passar por uma função de ativação. É a função de ativação que vai ser responsável por difundir ou não difundir a informação para os outros neurônios que constituem uma rede neural ou reproduzir a saída no caso do modelo *perceptron*. Existem vários tipos de funções de ativação, no presente trabalho falaremos de 4 tipos que são as seguintes:

- Função Limiar;
- Função Sigmóide;
- Função Retificadora;
- Função Tangente Hiperbólica;

Faremos uma breve descrição de cada uma das funções de ativação abaixo.

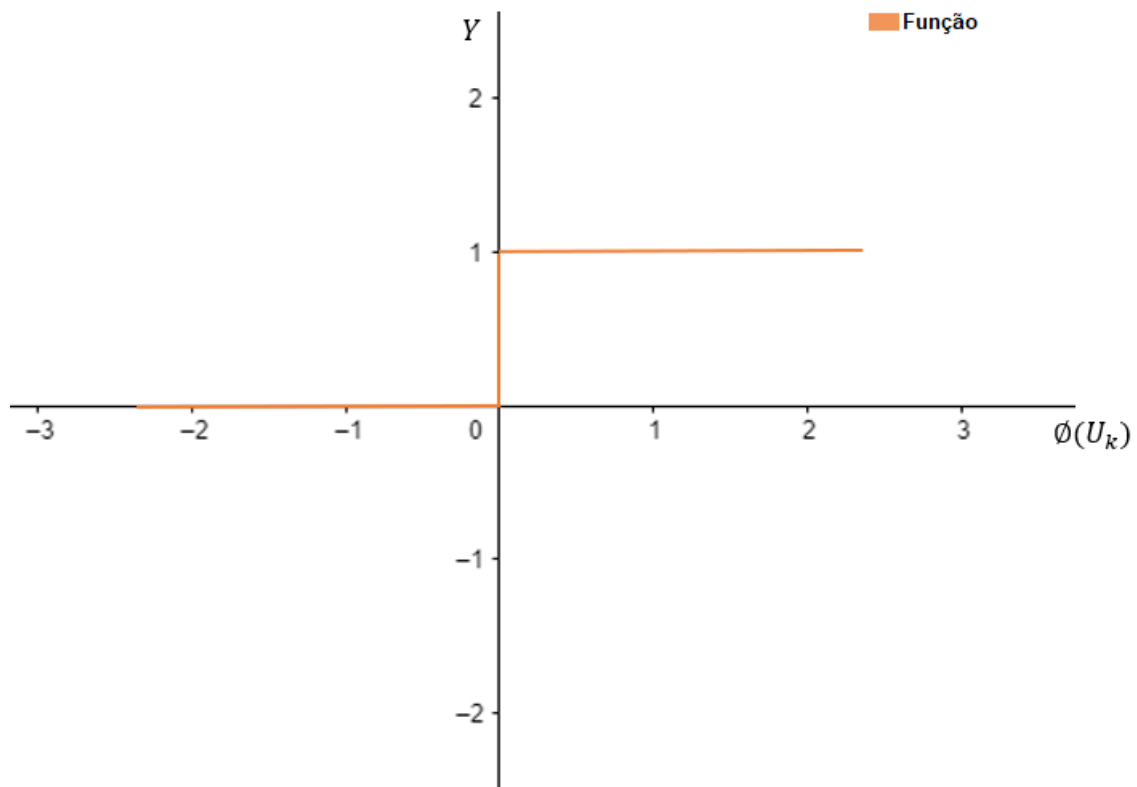
4.3.1.1 Função Limiar

É uma função constante que utilizamos para dizer que se o valor é maior que um determinado viés o neurônio dispara, ou seja a função assume o valor 1. Caso contrário assume valor 0. Portanto matematicamente podemos representar a função limiar de acordo com a função 37.

$$\phi(U_k) = \begin{cases} 1, & U_k \geq 0 \\ 0, & U_k < 0 \end{cases} \quad (37)$$

É importante sanar que na Função 37 o termo do viés b_k está implícito em U_k . Na Figura 22 observa-se o gráfico da respectiva função.

Figura 22: Função Limiar



Fonte:O autor

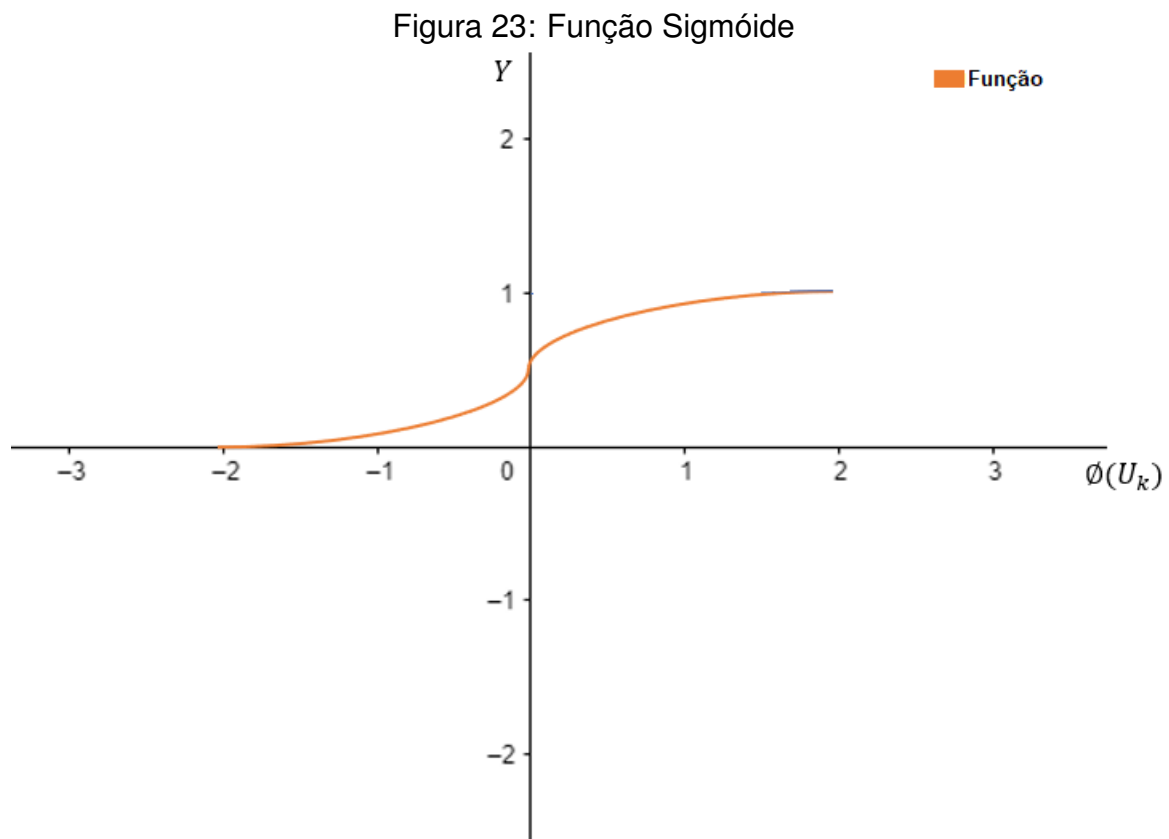
Ela é muito utilizada no modelo perceptron e em modelos que precisa-se tomar uma decisão binária.

4.3.1.2 Função Sigmóide

A função Sigmóide recebe esse nome em virtude do seu formato 'S'. É escrita, matematicamente da seguinte forma:

$$\phi(U_k) = \frac{1}{1 + e^{U_k}} \quad (38)$$

A função 38 era a mais utilizada em Redes Neurais Artificiais, em virtude de serem biologicamente mais plausíveis (FACURE, 2017), pois os neurônios biológicos funcionam de forma binária, como visto anteriormente a função sigmoide era uma boa forma de modelar esse comportamento uma vez que assume valores apenas entre 0 e 1. Dizemos que a derivada da função, ou seja, a taxa de variação da reta tangente ao gráfico da Figura 23 é 0 para $U_k < -2$ e $U_k > 2$.



Fonte: O autor

A função Sigmóide ou função logística como algumas referências a chamam, ainda é utilizada na última camada da rede Neural, com o objetivo de modelarmos variáveis binárias e assim obtermos uma saída para a rede de 1 ou 0.

4.3.1.3 Função Retificadora(ReLU)

Redes Neurais que utilizam a função retificadora, segundo (FACURE, 2017) são fáceis de otimizar uma vez que essa é parecida com a função identidade. O que as diferenciam é o fato da ReLU produzir zero na metade do seu domínio. Como consequência disso, as taxas de variações se mantêm grandes enquanto a unidade estiver ativa contribuindo com o aprendizado relativamente mais rápido da rede. A

função 39 é a função ReLU.

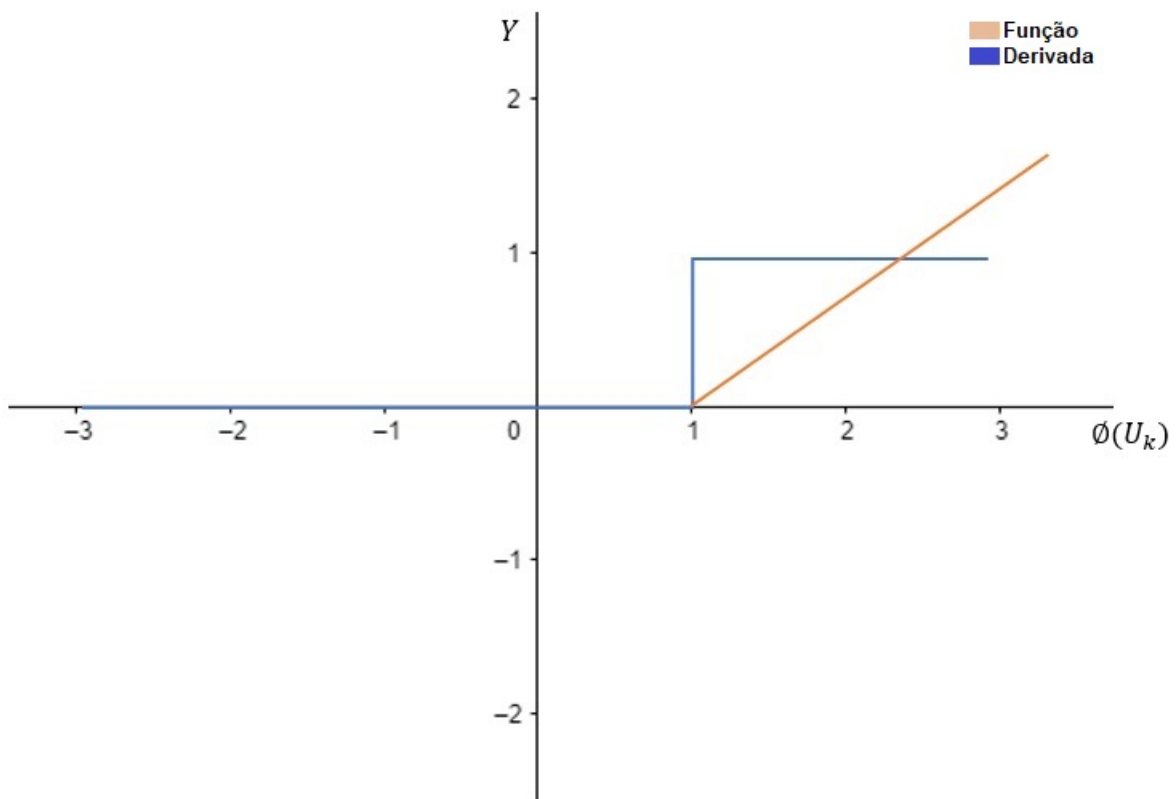
$$\phi(U_k) = \max(0, U_k) \quad (39)$$

A derivada da função 39 é dada por 40.

$$\phi'(U_k) = \begin{cases} 1, & U_k \geq 0 \\ 0, & U_k < 0 \end{cases} \quad (40)$$

Note que a derivada da função ReLU são estáveis, pois assumem sempre 1 para $U_k \geq 0$ e 0 para $U_k < 0$. A ativação ReLU, de acordo com (FACURE, 2017) é muito mais eficiente do que as funções sigmóides vistas anteriormente e é uma das descobertas que contribuiu significativamente para a popularidade da *Deep Learning*.

Figura 24: Função Retificadora



Fonte: O autor

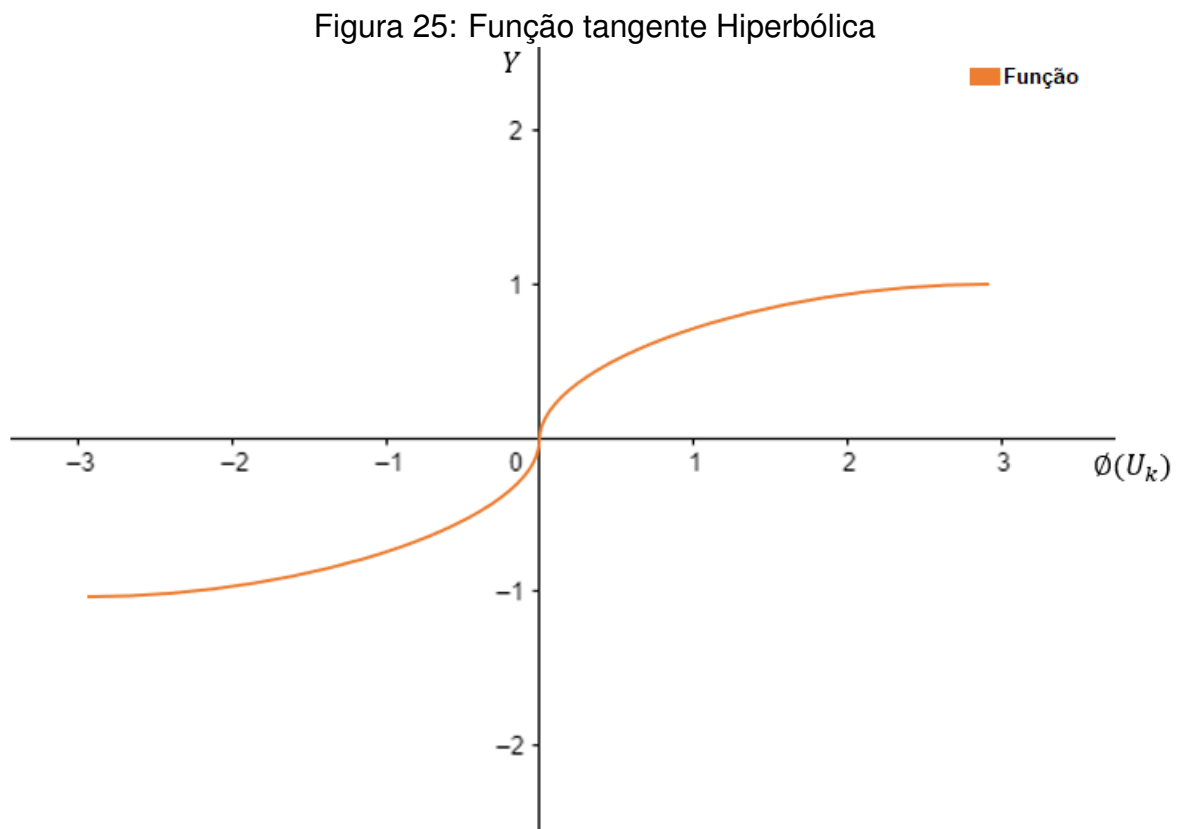
As funções ReLU's possuem uma desvantagem em virtude da tendência de propagação de pesos com valores baixos pela rede neural durante o treinamento e isso implica em uma taxa de aprendizagem muito baixa. Um caso que acontece quando a soma ponderada antes da aplicação da ReLU se torna negativa e o neurônio produz 0 ao aplicar a ReLU. Nessa região, a derivada também é zero implicando na não atualização dos parâmetros w durante o treinamento da rede. Na Figura 24 notamos o comportamento da função ReLU.

4.3.1.4 Função Tangente Hiperbólica

A função Tangente Hiperbólica é uma função similar a função sigmoide na qual possui também um formato de 'S', todavia varia entre -1 e 1. Essa função se aproxima mais da identidade e por isso possui uma melhor aplicação para as camadas ocultas de uma rede neural. Matematicamente a função 41 mostra sua descrição.

$$\phi(U_k) = \frac{1 - e^{2U_k}}{1 + e^{-2U_k}} \quad (41)$$

A Figura 25 mostra o gráfico da Função 41.

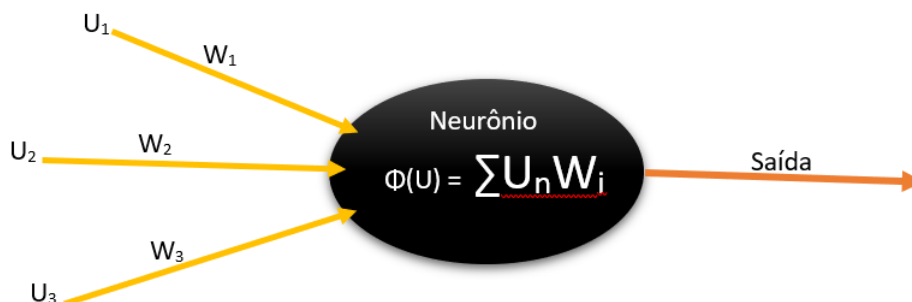


Veja que podemos ter taxas de variações maiores ao contrário da função sigmoide, uma vez que a reta tangente ao gráfico da Figura 25 possui maiores variações em sua inclinação.

4.4 UM EXEMPLO DE UMA REDE PERCEPTRON

Na presente seção faremos um exemplo de como um modelo perceptron funciona. Observe a Figura abaixo:

Figura 26: Perceptron



Fonte: O autor

Vamos supor que temos um problema de ir ou não a um show. Perceba que temos duas possibilidades de saída que é ir; a representaremos por 1 e a possibilidade de não ir ao show que vamos representá-la por 0. Logo seria interessante para este problema duas funções de ativação a Limiar ou a Sigmoide. Não precisamos de funções que tenham um gradiente grande pois as redes do tipo *Perceptron* possuem um neurônio apenas. Vamos utilizar, por questões de simplificação a função Limiar. Vamos escrever algumas premissas para uma pessoa que vai ao show decidir se vai ou não.

- U_1 : Vai chover ou não?
- U_2 : A banda que vai tocar, a pessoa gosta?
- U_3 : Vai sozinha ou acompanhada?

Baseado nisso vamos distribuir pesos a cada uma dessas premissas. Mas temos que fazer algumas considerações. Vamos supor que essa pessoa gosta muito de sair. Portanto a gente escolhe um viés baixo. Então faremos $b = 5$, lembrando que o viés é o valor que o somatório tem que chegar para o neurônio ser ativado. Como a pessoa gosta de assistir shows de bandas que ela costuma ouvir o fato de chover ou não possui pouca relevância nessa questão, em virtude disso possui um peso baixo. Vamos imaginar que a pessoa fique com muita vontade de ir ao show em virtude da banda que vai tocar. Portanto essa situação terá um maior peso para a pessoa decidir se vai ou não ao show. Como o fato de qual banda vai tocar ser um fator decisivo, se ela for sozinha ou acompanhada terá pouca relevância relativa ao fato de qual banda vai tocar. Todavia relativo ao fato de chover ou não terá mais relevância. Logo podemos supor que se essa pessoa gosta muito de ir com amigos ao show, o peso da última situação é um peso relativamente relevante para a decisão da pessoa de ir ou não para o show. Então atribuiremos um valor para $W_1 = 2$, para $W_2 = 5$ e para

$W_3 = 4$. É importante sanar que os valores atribuídos são valores aleatórios seguindo o raciocínio exposto.

Vamos imaginar as seguintes situações representadas abaixo:

1. 1,0,1
2. 0,0,1
3. 1,1,1
4. 0,1,0

No primeiro caso temos que:

$$1 * W_1 + 0 * W_2 + 1 * W_3 \quad (42)$$

No caso temos que não vai chover, a pessoa não gosta da banda mas a pessoa iria para esse show acompanhada. Logo observe que o somatório total seria $1 * 2 + 0 * 5 + 1 * 4 = 6$. Observe também que o viés é 5 logo o somatório é maior que o viés e portanto a resposta é positiva e a pessoa vai para o show bastando que não chova e que os amigos vão. Agora observe que para esse primeiro caso o fato de chover ou não é importante. Pois caso chovesse a resposta seria negativa em virtude do resultado do somatório ser menor que o viés.

No segundo caso temos que:

$$0 * W_1 + 0 * W_2 + 1 * W_3 \quad (43)$$

No caso temos que vai chover, a pessoa não gosta da banda mas a pessoa iria para esse show acompanhada. Logo observe que o somatório total seria $0 * 2 + 0 * 5 + 1 * 4 = 4$. O viés é 5 logo o somatório é menor que o viés. Portanto a resposta é negativa implicando que a pessoa não vai ao show bastando que chova, que os amigos vão e que ela não goste da banda.

No Terceiro caso temos que:

$$1 * W_1 + 1 * W_2 + 1 * W_3 \quad (44)$$

No caso temos que não vai chover, a pessoa gosta da banda e a pessoa vai para esse show acompanhada. O somatório total seria $1 * 2 + 1 * 5 + 1 * 4 = 9$. O viés é 5 então o somatório é maior que o viés, portanto a resposta é positiva e a pessoa vai para o show.

No quarto caso temos que:

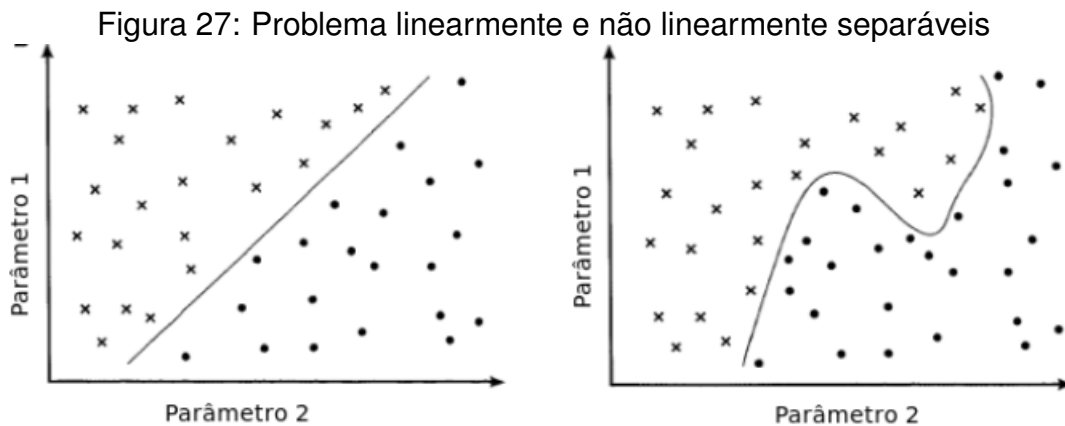
$$0 * W_1 + 1 * W_2 + 0 * W_3 \quad (45)$$

No caso temos que vai chover, a pessoa gosta da banda e a pessoa vai para esse show não acompanhada. Logo observe que o somatório total seria $0*2+1*5+0*4 = 5$. Observe também que o viés é 5 logo o somatório é igual que o viés e portanto a resposta é positiva e a pessoa vai para o show, bastando que a banda que ela goste vá tocar.

Note que quanto menor o viés mais provável é a rede *perceptron* gerar uma resposta positiva e quanto maior o viés menos provável é a rede *perceptron* gerar uma resposta negativa.

4.5 PERCEPTRON DE VÁRIAS CAMADAS

Um perceptron de várias camadas é o que chamamos de rede neural. Foram instauradas com o intuito de resolver problemas que não são linearmente separáveis. A Figura 35 mostra dois gráficos que ajuda a nos mostrar a ideia intuitiva de problemas linearmente separáveis e não separáveis.

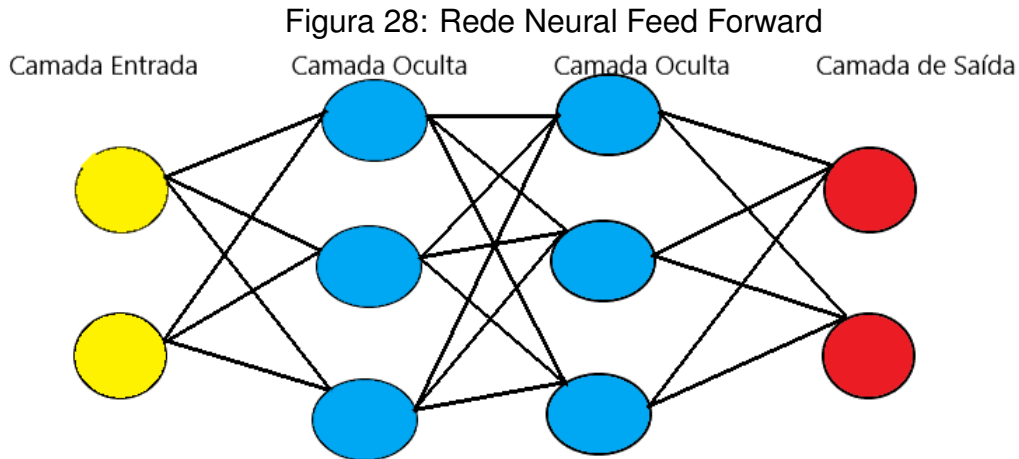


Fonte: (GONÇALVES, 2002)

Note que na Figura 27 que o gráfico da esquerda mostra um problema linearmente separável que pode ser resolvido simplesmente utilizando perceptron de uma única camada, como no caso do exemplo discutido na seção anterior. Por outro lado, no gráfico da direita temos os problemas que não são linearmente separáveis e é neste sentido que necessitamos de perceptrons de várias camadas.

4.5.1 Perceptrons com várias camadas

Existem diversos tipos de redes neurais, um deles são as redes do tipo *feedforward*, ou seja, uma rede progressiva tal que a saída de um neurônio se conecta com outro neurônio da próxima camada, sempre no sentido esquerdo/direito, formada por um conjunto de neurônios denominado "nós". Observe na Figura 28 uma rede neural do estilo descrito.



Fonte: O autor

Numa rede como da Figura 28 temos dois inputs de entrada no qual podemos identificá-los por um único vetor de entrada I .

$$I = \begin{pmatrix} i_1 & i_2 \end{pmatrix} \quad (46)$$

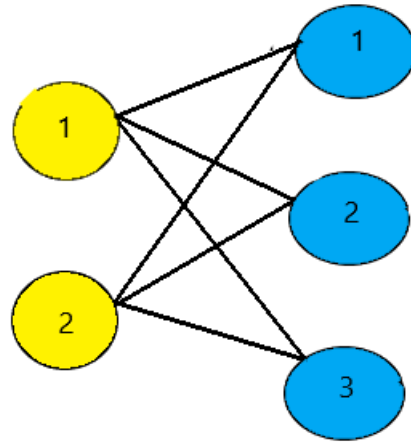
Neste caso temos também um único vetor de saída com dois resultados

$$O = \begin{pmatrix} o_1 & o_2 \end{pmatrix} \quad (47)$$

A complexidade deste tipo de rede se dá pela quantidade de camadas ocultas como também pelo número de neurônios que existem em cada uma. Repare que a saída de cada neurônio se conecta com outro neurônio da próxima camada sempre no sentido esquerdo/direito. Cada neurônio recebe os valores de entrada como identificados no perceptron, nas seções anteriores do presente capítulo e são multiplicados pelos pesos sinápticos e somados entre si junto com o bias. É a mesma expressão representada na equação 34.

Observe a Figura 29 no qual está representada uma rede neural com uma camada de entrada e somente uma camada oculta para ficar claro como ocorre o funcionamento de uma rede neural.

Figura 29: Rede neural e seus pesos
Camada Entrada Camada Oculta



Fonte: O autor

Observe na rede neural que temos dois neurônios na camada de entrada identificada pela cor amarela e três neurônios na camada oculta identificada pela cor azul. Podemos atribuir pesos sinápticos para cada neurônio. Logo denotaremos o peso da informação do neurônio 1 da camada amarela no neurônio 1 da camada azul usando a notação w_{11} , o peso da informação do neurônio 1 da camada amarela no neurônio 2 da camada azul será denotado por w_{12} e assim por diante, além disso esses pesos serão somados junto com o bias que na equação 48 está representado pela letra b .

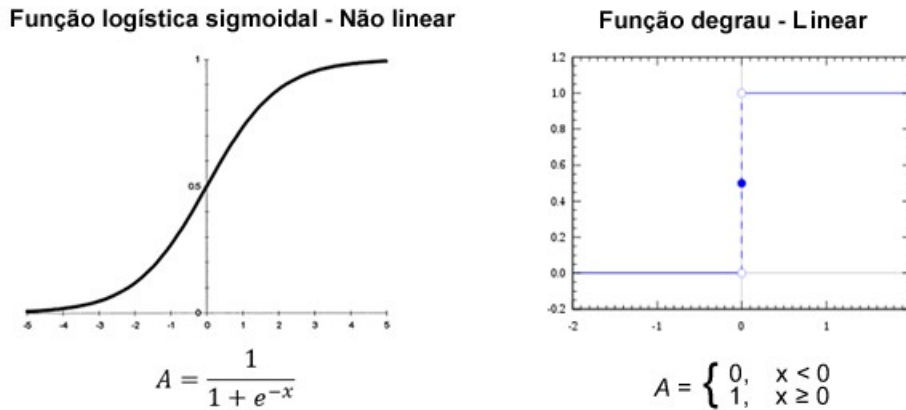
$$U = \sum_{i=1}^2 \sum_{j=1}^3 y_i w_{ij} + b \quad (48)$$

É importante notar na equação acima que temos os dados de entrada sendo ponderados pelos pesos denotados por w_{ij} e o bias que tem o intuito de centralizar a curva da função de ativação, como já vimos em seções anteriores, em um valor conveniente. Se o valor do bias for positivo, o gráfico se movimentará para a esquerda, diminuindo o valor do eixo x . Caso contrário, se o valor for negativo, o gráfico se movimentará para a direita, aumentando o valor do eixo x . A soma ponderada mostrada na equação 48, de acordo com (GRUBLER, 2018), gera o potencial de ativação que será utilizado para determinar seu valor e propagar para outros neurônios da próxima camada.

Uma vez realizado o somatório ponderado dos neurônios, o total obtido passa pela função de ativação ou também chamada de transferência. Como já visto, existem várias funções de ativação como a função degrau, identidade e funções não lineares como a gaussiana, tangente hipérbolica, sigmóide, entre outras. Em problemas

lineares a função limiar é uma das mais populares para se utilizar em perceptrons como mostrada no exemplo da seção 4.4. Por outro lado, em problemas não lineares a técnica mais utilizada é a sigmoideal. Observe a imagem abaixo extraída do site (GRUBLER, 2018).

Figura 30: Função Degrau - Linear e Função Logística Sigmoideal - Não Linear



Fonte: (GRUBLER, 2018)

Observe que na equação mostrada na Figura 30 X representa o valor obtido na soma ponderada do neurônio.

4.5.2 Backpropagation

A partir do momento em que uma rede neural artificial é inicializada, os pesos sinápticos recebem valores aleatórios. Após esses valores serem estabelecidos, esses são multiplicados pelos valores recebidos. Quando este processo é realizado, algumas vezes os valores desejados não são atingidos durante o treinamento em virtude desses pesos sinápticos. A técnica mais utilizado é o **Backpropagation** (GRUBLER, 2018), que tem por finalidade corrigir os valores dos pesos pela diferença entre o valor obtido e o valor esperado pelo algoritmo, propagando-os para todos os neurônios.

O processo de correção citado no parágrafo anterior, ocorre em duas etapas. O primeiro processo é chamado *Feed Forward*. Nesse processo temos um valor estabelecido na camada de entrada e outro na camada de saída. Então ocorre o processo de cálculo entre as camadas ocultas da rede até que a resposta seja reproduzida na camada de saída. (GRUBLER, 2018). Após ocorrer este processo o valor obtido na saída é comparado com o valor desejado da forma como mostra a equação 49

$$\Delta v = V_o - V_e \quad (49)$$

Onde V_o é o valor obtido, V_e é o valor esperado e Δv é o erro que será propagado para as demais camadas internas e é através deste processo que ocorrerá o aprendizado.

5 EXTRAÇÃO DE CARACTERÍSTICAS E REDES NEURAIIS

No presente capítulo estudamos duas formas distintas de entradas para uma rede neural, utilizando o pacote matlab nnstart-mathworks, e comparamos os desempenhos. As características (ou features do inglês) do modelo foram extraídas do dataset. Tais Features foram extraídas de um banco de arquivos de áudio com 501 acordes no formato *WAV* com auxílio da linguagem de programação *Python*. Esse banco de dados foi retirado do site *MIREX* tal qual possui 249 acordes maiores e 252 acordes menores tocados em uma guitarra. Observe a Figura 31 onde podemos ver a assinatura de um sinal do acorde de $F\#$ que faz parte do dataset usado..

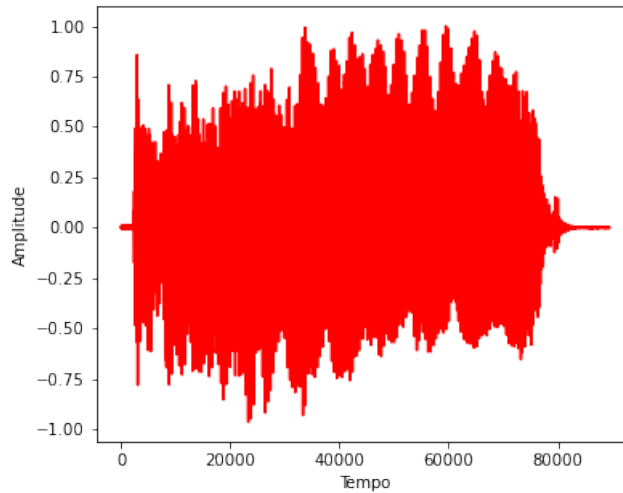
A partir do data set obtêm-se o vetor de entrada da rede neural. Este vetor é obtido por meio de rotina implementada no Python. Na primeira técnica, o vetor com as features é caracterizado com as primeiras 800 frequências relacionadas aos picos maiores para cada acorde. Para a segunda, o vetor com as features foi feito utilizando uma técnica chamada Local Phase Quantization (LPQ) descrita em (OJANSIVU; HEIKKILA, 2008) e (KAPP et al., 2014), a qual é muito aplicada para sinais do tipo imagem e , portanto, adaptamos esta técnica para sinais de som com o propósito de avaliar se o desempenho é satisfatório. No caso da LPQ o vetor de features, que corresponde a quantização do data set, possui apenas 256 valores. Em ambas abordagens a normalização do vetor de features é feita por meio de standartscale a qual possui função em matlab/python para uso.

5.1 TREINAMENTO COM VETOR DE FREQUÊNCIAS

Utilizando conceitos relacionados ao processamento digital de sinais, aplicamos a transformada de Fourier para obtermos a assinatura espectral do acorde em termos das frequências que os constituem. Veja que a Figura 31 mostra o sinal em função do tempo de propagação. Dividimos esse tempo que equivale a 4 segundos, em partes de 1 segundo de sinal, uma vez que em 1 segundo já é o bastante para conseguirmos extrair frequências características dos acordes tocados neste banco de dados utilizado. Com respeito aos janelamentos discutidas no capítulo 3, a Figura 32 mostra

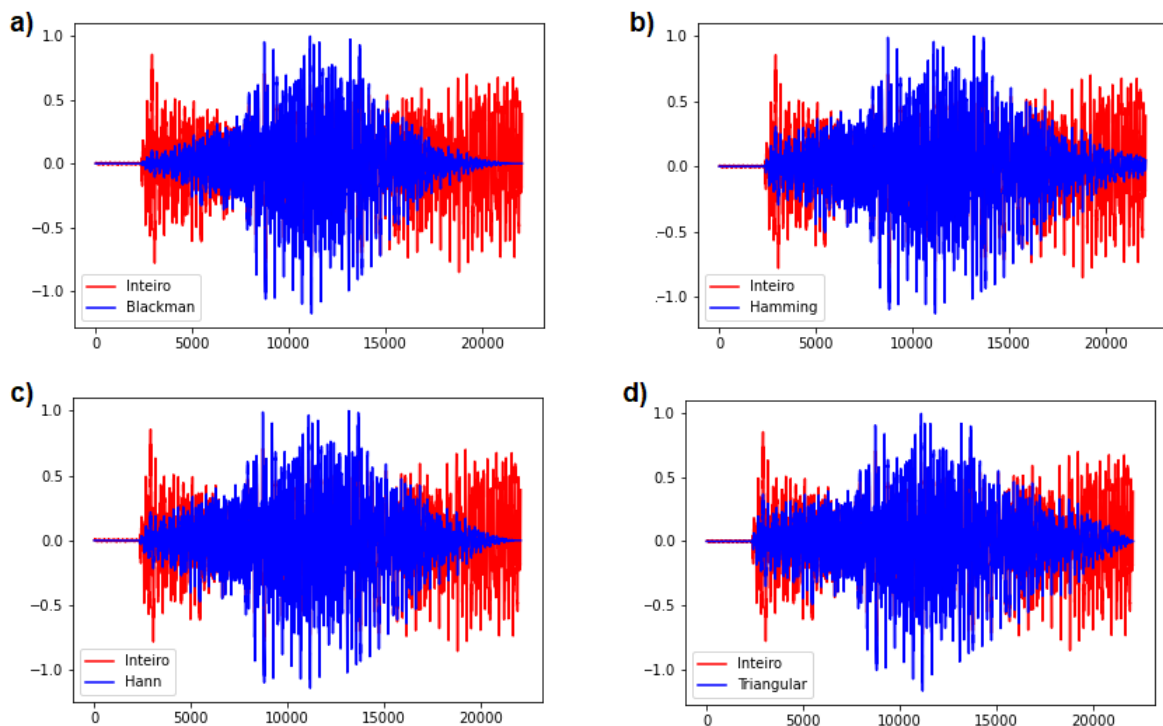
os resultados obtidos ao aplicar tais janelas no sinal.

Figura 31: Sinal do Acorde $F\#$



Fonte: O autor

Figura 32: a) Janela de Blackman b) Janela de Hamming c) Janela de Hann d) Janela Triangular

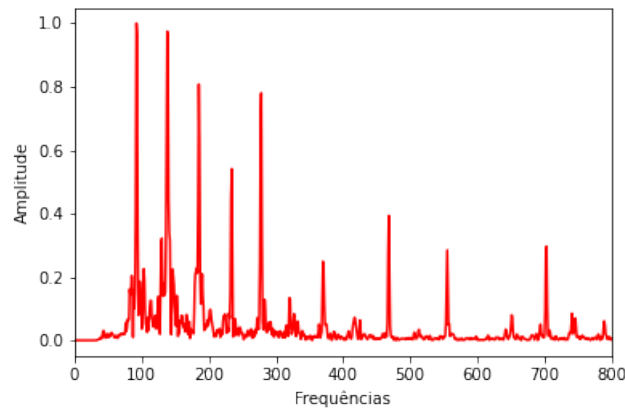


Fonte: O autor

Note entre as janelas mostradas, a que suaviza mais o sinal na entrada e na saída é a de Blackman. Estamos interessados em pegar a parte do sinal sem muito ruídos

no começo e no final de sua propagação e assim obter frequências mais próximas do acorde tocado. Por este motivo, o janelamento escolhido será o de Blackman. Após isso, aplica-se a transformada de Fourier na janela de Blackman. A Figura 33 mostra o espectro de frequências envolvidas no sinal.

Figura 33: Frequências do Acorde $F\#$

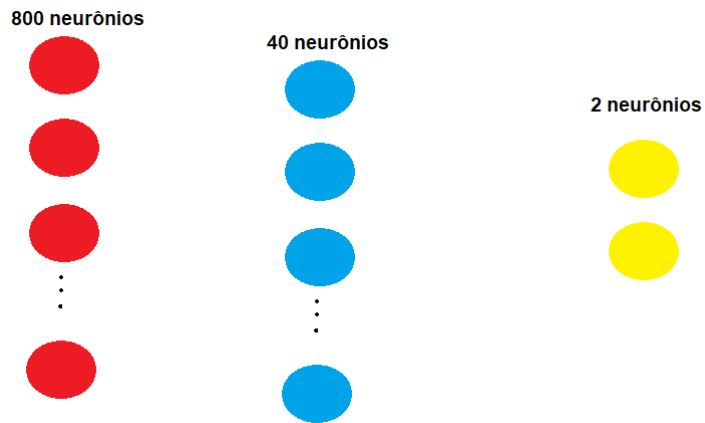


Fonte: O autor

Após o procedimento, foi gerado um vetor de frequências com um tamanho de 800 valores listados com frequências associadas a maiores amplitudes em ordem decrescente. Cada vetor de frequência possui um valor de saída identificado no preparo do arquivo no formato `.x/sx`. Denotamos a classificação em acordes maiores como $[1, 0]$ e acordes menores como $[0, 1]$. Esses vetores de frequências para entrarem na rede neural tiveram de ser normalizados utilizando uma função da biblioteca *sklearn* da linguagem python chamada *standard scaler*, pois os dados precisam estar em uma escala padrão para não haver erros na classificação, respostas absurdas e a rede conseguir diferenciá-los de maneira correta. Ao prepararmos o arquivo com esses features utilizamos uma rede própria de um dos pacotes do *MatLab* da Mathworks. A rede neural foi ajustada de tal forma que chegássemos a obter um bom reconhecimento utilizando 70% dos dados para treino, 15% para validação e 15% para teste. A Figura 34 mostra a arquitetura da rede utilizada para esses features.

Utilizando essa configuração, a Figura 35 mostra os resultados de reconhecimento de treino, teste, validação e o reconhecimento total se for utilizada uma rede neural, configurada na forma da Figura 34, com os features descritos no parágrafo anterior.

Figura 34: Rede Neural com 1 camada oculta



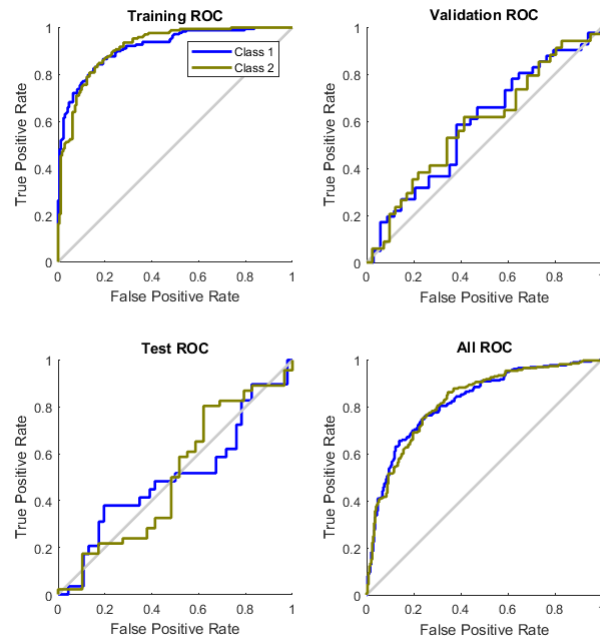
Fonte: O autor

Figura 35: Matrizes de Confusão da Rede Neural dada pela Figura da 34.



Fonte: O autor

Figura 36: Curvas ROC com 1 camada oculta



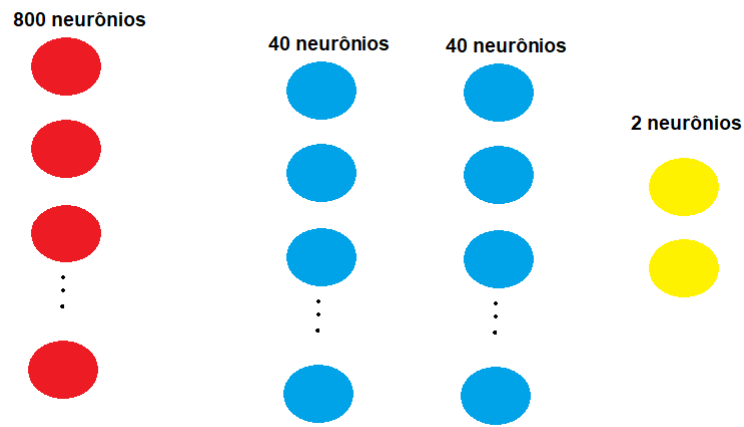
Fonte: O autor

Observe que na Figura 35 temos 4 matrizes de confusão. A matriz de treino, identificada pela letra **a)** mostra que os 70% dos dados que foram usados para treino tiveram uma acuracidade de 83.2%, a matriz identificada pela letra **b)** mostra que os 15% dos dados foram validados com uma acuracidade de 60.0%, a matriz identificada pela letra **c)** mostra que os outros 15% dos dados foram testados com uma acuracidade de 53.3% e a matriz identificada pela letra **d)** mostra o desempenho total da rede que foi de 75.2%.

Agora observe a Figura 36 onde temos 4 curvas ROC que são curvas para medir e comparar o desempenho de modelos classificatórios binários. Temos curvas representando duas classes, tanto acordes maiores como menores. Quanto mais a curva se aproxima do canto superior esquerdo do gráfico melhor é o valor da acuracidade. Essa acuracidade corresponde às obtidas nas matrizes mostradas na Figura 35. Estes tipos de gráficos mostram o desempenho das duas classes que foram utilizadas, tanto de acordes maiores, *classe 1* quanto de acordes menores, *classe 2*. O desempenho da base de teste não foi satisfatório apresentando falsos positivos com performance menor que a situação de escolha randômica.

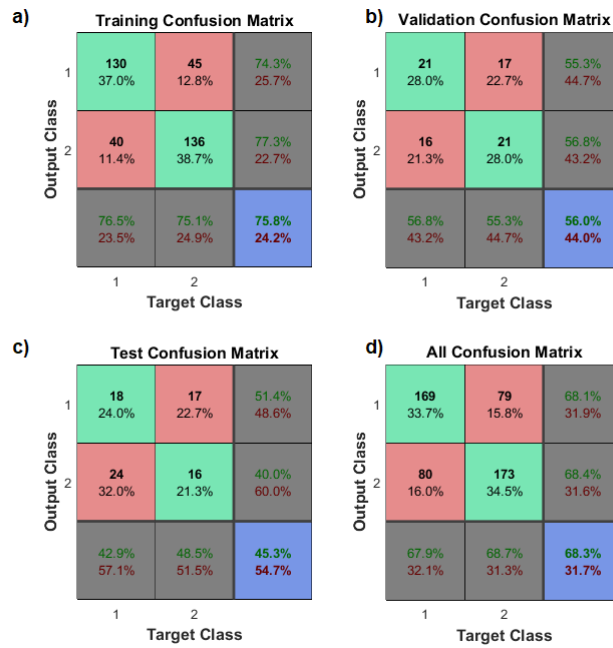
Com base neste desempenho, foi feito um outro treinamento com a rede neural modificada onde foi adicionada mais uma camada oculta com 40 neurônios semelhante a outra. A Figura 37 mostra a arquitetura desta rede. Utilizando a configuração da Figura 37, a Figura 38 mostra os resultados obtidos.

Figura 37: Rede Neural com 2 camadas ocultas



Fonte: O autor

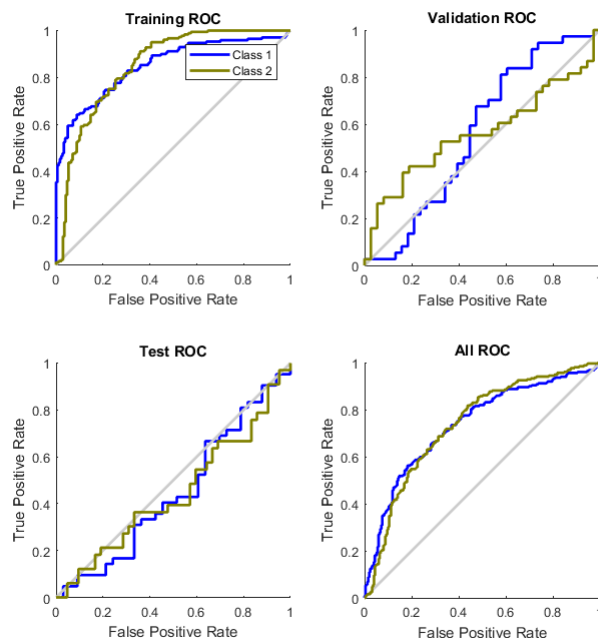
Figura 38: Matrizes de Confusão da Rede Neural Figura 37



Fonte: O autor

Observe que na Figura 38 temos as 4 matrizes. A matriz de treino, identificada pela letra **a)** mostra que os 70% dos dados que foram usados para treino tiveram uma acuracidade de 75.8%, a matriz identificada pela letra **b)** mostra que os 15% dos dados foram validados com uma acuracidade de 56.0%, a matriz identificada pela letra **c)** mostra que os outros 15% dos dados foram testados com uma acuracidade de 45.3% e a matriz identificada pela letra **d)** mostra o desempenho total da rede que foi de 68.3%. Adicionar uma camada extra teve pouco impacto na melhora dos dados de treino como pode ser observado na Figura 39 onde temos as 4 curvas ROC referente a rede neural da Figura 37.

Figura 39: Curvas ROC com 2 camadas ocultas



Fonte: O autor

Observe que a introdução de uma nova camada oculta pouco melhorou o desempenho dos testes bem como da validação.

5.2 TREINAMENTO UTILIZANDO O MÉTODO LOCAL PHASE QUANTIZATION (LPQ)

Os resultados utilizando os features com as frequências relacionadas aos picos foram insatisfatórios, uma vez que os reconhecimentos nos testes ficaram muito abaixo de 70.0%. Para comparação, com respeito a mesma rede, utilizamos o método conhecido como Local Phase Quantization (LPQ) que, de acordo com (OJANSIVU; HEIKKILA, 2008) e (KAPP et al., 2014), é um método que é aplicado para imagens e que

melhora muito o reconhecimento quando utilizado. Adaptamos o Local Phase Quantization para vetores de áudio que são unidimensionais.

Para isso utilizamos o sinal de áudio inteiro, pois o LPQ consiste na convolução do sinal com a função $e^{-2\pi j\xi t}$. A convolução não é calculada em todo o domínio e sim em um intervalo I_t em torno de cada tempo t . Nesta técnica, são usados dois parâmetros $\xi_1 = \alpha$ e $\xi_2 = -\alpha$. O escalar positivo α é uma frequência dada em termos do comprimento do intervalo I_t . Por fim, uma vez que as convoluções são calculadas em ξ_1 e ξ_2 , os valores são quantizados e, após a quantização, convertidos em um vetor de 256 componentes. Com esta técnica reduz-se bastante o conjunto de dados e melhora o reconhecimento. No caso, um vetor com 44709 elementos de áudio foi reduzido para 256 elementos.

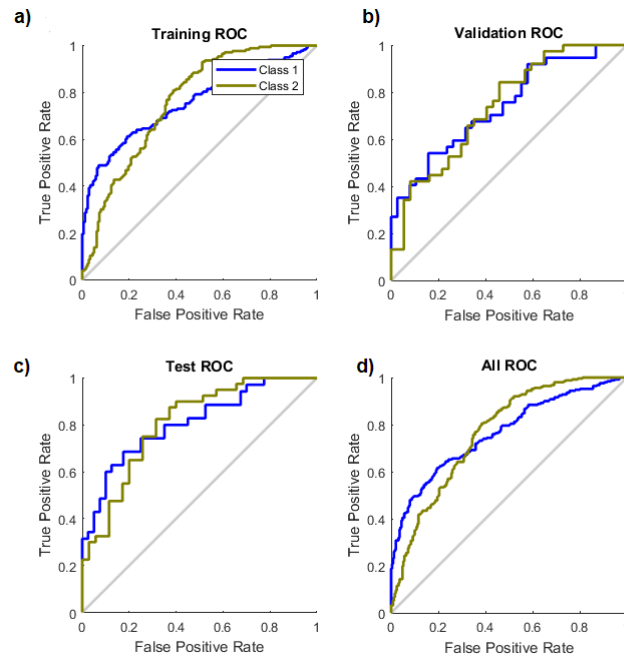
Observe na Figura 40 as matrizes de confusão para estes features. A rede neural possui a mesma arquitetura apresentada na Figura 34, mas com apenas 256 entradas. Note que na Figura 40, a matriz de treino, identificada pela letra **a)** mostra que os 70% dos dados que foram usados para treino tiveram uma acuracidade de 70.1%, a matriz identificada pela letra **b)** mostra que os 15% dos dados foram validados com uma acuracidade de 64.0%, a matriz identificada pela letra **c)** mostra que os outros 15% dos dados foram testados com uma acuracidade de 74.7% e a matriz identificada pela letra **d)** mostra o desempenho total da rede que foi de 69.9%. Agora observe a Figura 41 onde temos as 4 curvas ROC, note que as curvas da Figura 41 estão mais próximas do canto superior esquerdo.

Figura 40: Matrizes de Confusão da Rede Neural para o LPQ com 40 neurônios na camada oculta e duas saídas.



Fonte: O autor

Figura 41: Curvas ROC com 1 camada oculta de 40 com entradas fornecidas pelo método LPQ



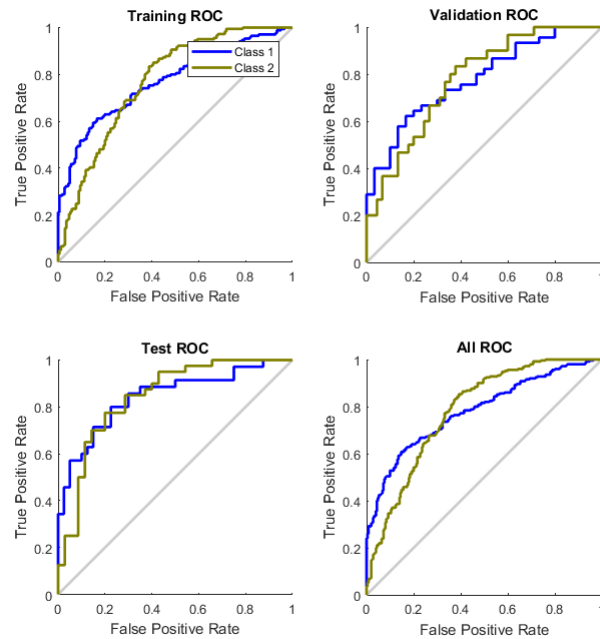
Fonte: O autor

Figura 42: Matrizes de Confusão da Rede Neural com duas camadas ocultas usando LPQ



Fonte: O autor

Figura 43: Curvas ROC com 2 camadas ocultas com 40 neurônios para o método LPQ



Fonte: O autor

Após isso, foi utilizado os mesmos features com uma rede neural semelhante, a menos do número de entradas, a da Figura 37 com duas camadas. E os resultados são mostrados nas matrizes da Figura 42. Observe a Figura 42, a matriz de treino, identificada pela letra **a)** mostra que os 70% dos dados que foram usados para treino tiveram uma acuracidade de 72.4%, a matriz identificada pela letra **b)** mostra que os 15% dos dados foram validados com uma acuracidade de 69.3%, a matriz identificada pela letra **c)** mostra que os outros 15% dos dados foram testados com uma acuracidade de 78.7% e a matriz identificada pela letra **d)** mostra o desempenho total da rede que foi de 72.9%. Agora observe a Figura 43 onde temos as 4 curvas ROC referente a rede neural da Figura 37, com 256 entradas, que mostra o bom desempenho da rede para a data-base escolhida. Veja que os gráficos estão mais próximos do canto superior esquerdo, uma vez que esses tipos de features criados utilizando a técnica do LPQ obtiveram um melhor reconhecimento na rede neural utilizada.

5.3 TREINAMENTO DA REDE NEURAL UTILIZANDO O K-MEANS CLUSTERING

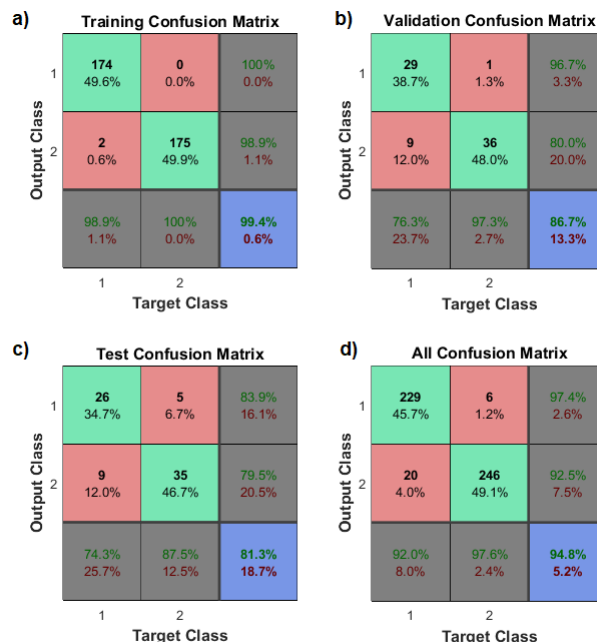
De acordo com (OLIVEIRA, 2019), utilizando o método não supervisionado K-means para tratar features com o intuito de treinar uma rede neural faz com que o desempenho de taxas de acerto seja maior. Seguindo a abordagem do relatório, aplicamos os k-means nos features das frequências e LPQ e, em nossa abordagem,

usamos 501 clusters para nossos 501 vetores com as features. Os vetores que possuem as features são elementos de um espaço euclidiano com dimensão d , onde temos $d = 800$ para frequências e $d = 256$ para LPQ. Considerando a base canônica $\{e_1, e_2, \dots, e_d\}$, vamos fazer a organização em clusters nas projeções dos vetores em cada um dos elementos desta base canônica. As novas entradas da rede neural tem como elementos os centroides obtidos após a aplicação do método. Para a realização, usamos a rotina K-means do Matlab/mathworks.

5.3.1 Aplicação nas frequências

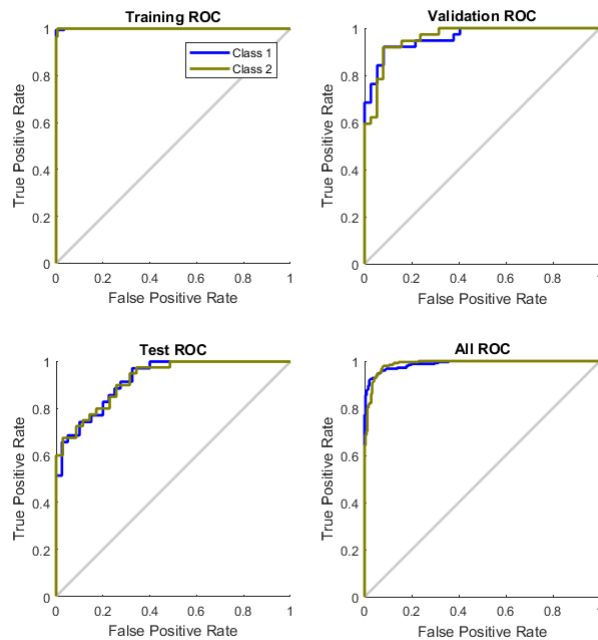
Em relação ao modelo com frequências, os resultados obtidos para uma rede com a arquitetura da Figura 34 estão representados na Figura 44. Note que na Figura 44 temos as 4 matrizes. A matriz de treino, identificada pela letra **a)** mostra que os 70% dos dados que foram usados para treino tiveram uma acuracidade de 99.4%, a matriz identificada pela letra **b)** mostra que os 15% dos dados foram validados com uma acuracidade de 86.7%, a matriz identificada pela letra **c)** mostra que os outros 15% dos dados foram testados com uma acuracidade de 81.3% e a matriz identificada pela letra **d)** mostra o desempenho total da rede que foi de 94.8%. Agora observe a Figura 45 onde temos as 4 curvas ROC referente a rede neural da Figura 37.

Figura 44: Matriz de confusão com 1 camada oculta com 40 neurônios para frequências usando cluster



Fonte: O autor

Figura 45: Curvas ROC com 1 camada oculta com 40 neurônios para frequências usando cluster



Fonte: O autor

Observe que as curvas ROC estão muito próximas de 1.0, ou seja temos uma boa classificação ao utilizar o k-means para tratar features do tipo utilizados no trabalho. Vamos agora aplicar os mesmos features, com uma rede neural com uma arquitetura da Figura 38 para as frequências.

Note que na Figura 46 temos as 4 matrizes. A matriz de treino, identificada pela letra **a)** mostra que os 70% dos dados que foram usados para treino tiveram uma acuracidade de 100%, a matriz identificada pela letra **b)** mostra que os 15% dos dados foram validados com uma acuracidade de 92.0%, a matriz identificada pela letra **c)** mostra que os outros 15% dos dados foram testados com uma acuracidade de 90.7% e a matriz identificada pela letra **d)** mostra o desempenho total da rede que foi de 97.4%. Agora observe a Figura 47 onde temos as 4 curvas ROC referente a rede neural da Figura 37.

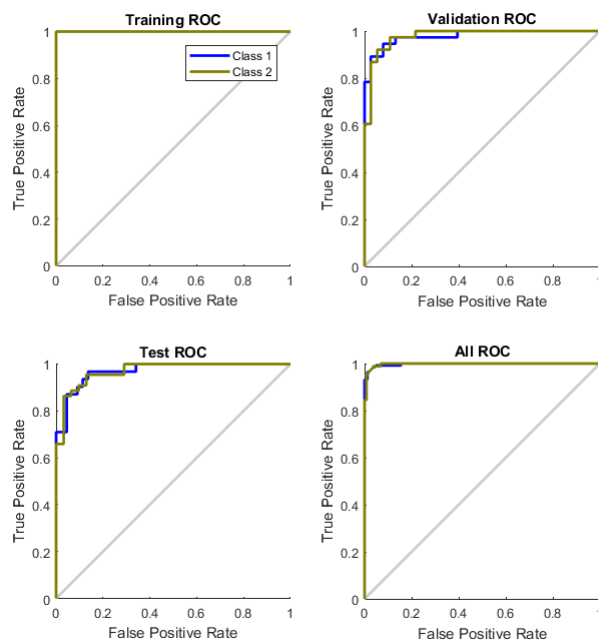
Observe que as curvas ROC estão muito próximas novamente de 1.0, ou seja temos uma classificação ainda melhor se utilizarmos o k-means e a rede neural com 2 camadas, com uma arquitetura semelhante a da Figura 37.

Figura 46: Matriz de confusão com 2 camadas ocultas de 40 neurônios para frequências usando cluster



Fonte: O autor

Figura 47: Curvas ROC com 2 camadas oculta oculta de 40 neurônios para frequência usando cluster



Fonte: O autor

5.3.2 Aplicação no LPQ

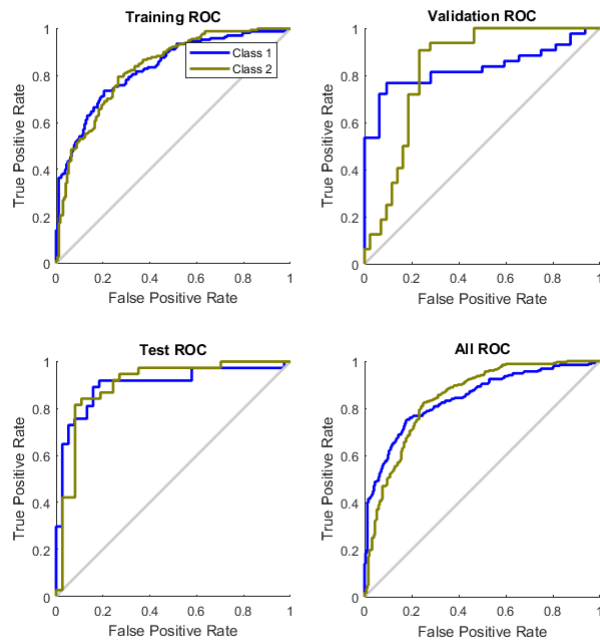
Por outro lado, fizemos mais testes aplicando o k-means e o LPQ. Os resultados obtidos aparecem nas matrizes de confusão na Figura 48 em uma rede neural com arquitetura semelhante a rede da Figura 34. Note que na Figura 48 temos as 4 matrizes. A matriz de treino, identificada pela letra **a)** mostra que os 70% dos dados que foram usados para treino tiveram uma acuracidade de 76.1%, a matriz identificada pela letra **b)** mostra que os 15% dos dados foram validados com uma acuracidade de 82.7%, a matriz identificada pela letra **c)** mostra que os outros 15% dos dados foram testados com uma acuracidade de 86.7% e a matriz identificada pela letra **d)** mostra o desempenho total da rede que foi de 78.6%. Agora observe a Figura 49 onde temos as 4 curvas ROC referente a rede neural da Figura 37. Observe que as curvas ROC estão menos próximas de 1.0, ou seja não temos uma classificação melhor se utilizarmos o k-means junto do LPQ. Uma rede Neural com arquitetura semelhante a da Figura 37 tem os seguintes resultados.

Figura 48: Matriz de confusão com 1 camada oculta de 40 neurônios para LPQ usando cluster



Fonte: O autor

Figura 49: Curvas ROC com 1 camada oculta de 40 neurônios para LPQ usando cluster



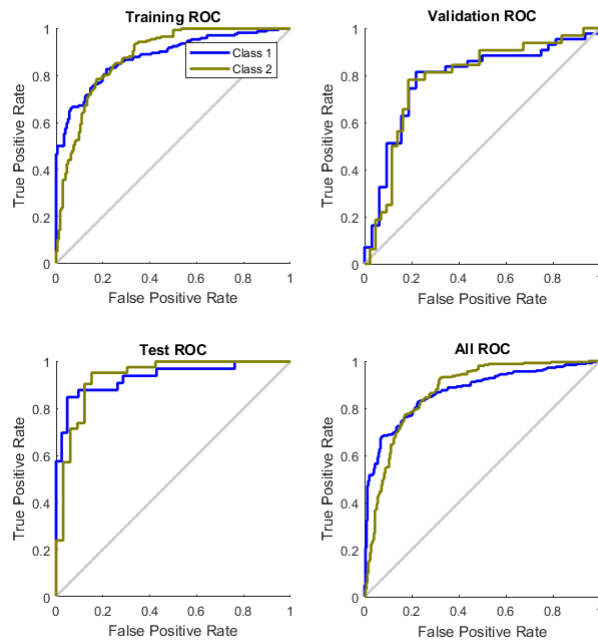
Fonte: O autor

Figura 50: Matriz de confusão com 2 camadas ocultas de 40 neurônios para LPQ usando cluster



Fonte: O autor

Figura 51: Curvas ROC com 2 camadas ocultas de 40 neurônios para LPQ usando cluster



Fonte: O autor

Note que na Figura 50 temos as 4 matrizes. A matriz de treino, identificada pela letra **a)** mostra que os 70% dos dados que foram usados para treino tiveram uma acuracidade de 79.8%, a matriz identificada pela letra **b)** mostra que os 15% dos dados foram validados com uma acuracidade de 72.3%, a matriz identificada pela letra **c)** mostra que os outros 15% dos dados foram testados com uma acuracidade de 84.0% e a matriz identificada pela letra **d)** mostra o desempenho total da rede que foi de 79.8%. Agora observe a Figura 51 onde temos as 4 curvas ROC referente a rede neural da Figura 37. Observe que as curvas ROC estão menos próximas de 1.0, ou seja não temos uma classificação melhor se utilizarmos o k-means junto do LPQ em uma rede neural com 2 camadas ocultas para os features preparados

6 DISCUSSÃO E CONCLUSÃO

No presente trabalho duas formas de treinar uma rede neural foram estudadas. Em uma delas, foram utilizadas frequências relacionadas com a escala temperada. Na outra, foi feita uma adaptação do método LPQ com uma redução dimensional. O LPQ apresentou uma melhor performance em relação ao método baseado nas frequências relacionadas ao maiores picos. Adaptando a abordagem proposta em (OLIVEIRA, 2019), foi possível melhorar significativamente o desempenho do método que usa frequências obtendo a melhor performance. Os resultados estão organizados de acordo com as categorias abaixo.

1. Frequências sem K-means - 1 Camadas Oculta : Neste caso, foi alcançada uma boa acuracidade durante o treino da rede neural com um valor de 83.30%, uma validação de 60.0%, um teste de 53.3% e um desempenho geral de 75.2%. Além disso, os testes não obtiveram um bom reconhecimento.
2. Frequências sem K-means - 2 Camadas Ocultas: Neste caso, foi obtida uma boa acuracidade durante o treino da rede neural com um valor de 75.8%, uma validação de 56.0%, um teste de 45.3% e um desempenho geral de 68.3%. Neste caso os testes não atingiram um reconhecimento melhor que o caso de 1 camada oculta. Durante o reconhecimento, se os dados não tiverem bem distribuídos, os neurônios podem adquirir pesos muito pequenos e, desta forma, na hora da multiplicação esses valores chegam próximos a zero e os neurônios acabam não participando da classificação. Então, neste caso, a adição de mais uma camada piora o reconhecimento da rede neural.
3. LPQ sem K-means - 1 Camada Oculta: Neste caso, obtivemos uma melhora nos resultados, com uma acuracidade no treino de 70.1%, uma validação de 64.0%, um teste de 74.7% e um desempenho geral de 69.1%, Então podemos concluir que o LPQ contribuiu para a melhoria dos resultados como mostra (KAPP et al., 2014) para tratamento de vetores de imagens. Todavia não houve uma melhora consideravelmente boa uma vez que a quantidade de dados foram poucas para o problema em questão.

4. LPQ sem K-means - 2 Camadas Ocultas: Para esta situação, obtivemos uma melhora nos resultados, com uma acuracidade no treino de 72.4%, uma validação de 69.3%, um teste de 78.7% e um desempenho geral de 72.9%. O acréscimo de mais uma camada oculta contribuiu desta vez para um melhor reconhecimento. No entanto, como explicado no parágrafo anterior não foi uma melhora considerável pelo mesmo motivo.
5. Frequências com K-means - 1 Camada Oculta: Neste caso, a melhora no resultado foi obtida, com uma acuracidade no treino de 99.4%, uma validação de 86.7%, um teste de 81.3% e um desempenho geral de 94.8%. Agora, observe que o k-means resolveu o problema da distribuição, como abordado por (OLIVEIRA, 2019), dos dados, facilitando assim a classificação.
6. Frequências com K-means - 2 Camadas Ocultas: Neste caso, a melhora no resultado foi obtida, com uma acuracidade no treino de 100%, uma validação de 92.0%, um teste de 90.7% e um desempenho geral de 97.4%. Observe que o k-means resolveu o problema da distribuição, como abordado por (OLIVEIRA, 2019), dos dados e a adição de mais uma camada foi bem sucedida para melhorar ainda mais o resultado, facilitando assim a classificação.
7. LPQ com K-means - 1 Camada Oculta: Neste caso, a melhora no resultado foi obtida, com uma acuracidade no treino de 76.1%, uma validação de 82.7%, um teste de 86.7% e um desempenho geral de 78.6%. Observe que os resultados melhoraram no reconhecimento. No entanto, a performance com o uso das frequências foi melhor.
8. LPQ com K-means - 2 Camadas Ocultas: A melhora no resultado foi obtida, com uma acuracidade no treino de 79.8%, uma validação de 73.3%, um teste de 84.0% e um desempenho geral de 79.4%. A adição de mais uma camada melhorou o reconhecimento com o LPQ e o k-means em relação ao caso anterior.

O método LPQ consegue uma boa performance quando aplicado sem o método híbrido. A análise indica que uma maior quantidade de dados para treino pode implicar em melhoria de performance. Com relação ao método híbrido apresentado em (OLIVEIRA, 2019), pesquisas adicionais são necessárias para determinar se tal técnica mantém sua performance em um volume grande de dados e em modelos de maior complexidade. Para um trabalho posterior, pretende-se criar uma interface inteligente. O objetivo é que tal interface tenha condições de, ao ser inserido uma música em algum formato de áudio, mostrar essa música com suas cifras. Desta forma, os músicos podem obter informações sobre a harmonia e tom das músicas.

REFERÊNCIAS

ACADEMY, Data Science. **Capítulo 4 - O Neurônio, Biológico e Matemático - Deep Learning Book**. Disponível em: <<https://www.deeplearningbook.com.br/o-neuronio-biologico-e-matematico/>>.

BAXTER, R.; HASTINGS, N.; LAW, A.; GLASS, E. J.. **Redes neurais artificiais**. 561–563p. v.39, n.5. Disponível em: <<https://sites.icmc.usp.br/andre/research/neural/>>. Acesso em: 2021-08-23.

CAVALCANTI, H. M. V. da Cunha. **Extração de características via redes neurais**. Campinas: Dissertação de Mestrado, 2000.

CLUB, Cifra. **Princípios de harmonia para guitarra violão e outros instrumentos**. Online: e-book, 2021.

FACURE, Matheus. **Funções de ativação**. Disponível em: <<https://matheusfacure.github.io/2017/07/12/activ-func/>>. Acesso em: 2021-08-24.

FARIA, Nelson. Acordes, arpejos e escalas para violão e guitarra. 2 a Edição. **Lumiar**, Rio de Janeiro, 1999.

Faria, Nelson. **Como classificar os intervalos musicais**. Disponível em: <<https://www.youtube.com/watch?v=Rs49A902S5o>>. Acesso em: 2021-07-01.

FESSLER, J. A. The Discrete Fourier Transform Contents. **Analysis**, Notas de Aula, p.1–31, 2011.

GONÇALVES, A. R. **Máquina de Vetores Suporte**. Unicamp: Notas de aula, 2002. Disponível em: <<https://www.dca.fee.unicamp.br/andreric/>>.

GRUBLER, MURILLO. **Entendendo o funcionamento de uma Rede Neural Artificial | by Murillo Grüber | aibrasil | Medium**. Disponível em: <<https://medium.com/brasil-ai/entendendo-o-funcionamento-de-uma-rede-neural-artificial-4463fcf44dd0>>. Acesso em: 2021-08-23.

HARRIS F., J. Windows, harmonic analysis, and the discrete fourier transform. **PN editors**, New York, p.1–69, 1976.

JESUS, W.; SOUZA, G. R. de. Identificação e Classificação de Acordes Musicais aplicando a Transformada de Fourier. **Revista de Matemática da UFOP**, Pará, n.1, p.151–237, 2020.

KAPP, Marcelo N.; BLOOT, Rodrigo; CAVALIN, Paulo R.; OLIVEIRA, Luiz E.S. Automatic forest species recognition based on multiple feature sets. **Proceedings of the International Joint Conference on Neural Networks**, Beijin, n.August 2017, p.1296–1303, 2014.

LATHI, B. P. **Sinais e Sistemas Lineares**. Porto Alegre: bookman, 2007.

NETO, P. J.S. **Solução Computacional para Reconhecimento de Harmonias Musicais**. Brasília: Dissertação de Mestrado, 2015.

NUSSENZVEIG, Herch Moysés. **Curso de Física Básica**: fluidos, oscilações e ondas, calor. São Paulo: Editora Blucher, 2018. v.2.

OJANSIVU, V.; HEIKKILA, J. Blunr insensitive texture classification using local phase quantization. **Proceedings of the 3rd International Conference on Image and Signal Processing (ICISP '08)**, Cherbourg-Octeville, p.236–243, 2008.

OLIVEIRA, V. P. L. Redes Neurais Artificiais e K-médias em um Modelo Híbrido Alternativo para a Classificação de Clientes em Aprovação de crédito. **Relatório de pesquisa-UFG**, Goiânia, 2019.

OPPENHEIM, Alan V; WILLSKY, Alan S. **Sinais e Sistemas, 2. ed.** São Paulo: Pearson, 2010. 592p.

PIRES, A. S.T. **Evolução das Ideias da Física**. 1.ed. São Paulo: Copyright, 2008.

SANTO, Antônio Espírito. **Convolução de Sinais**. Disponível em: <<http://webx.ubi.pt/aes/AS/lab7.pdf>>.

SIMONATO, Adriano Luís; DIAS, Maria Palmira Minholi. A Relação Matemática e Música. **Revista Fafibe On-line**, Bebedouro, v.1, n.1, p.1–6, 2011.

SOUSA, Rainer. **A origem das notas musicais**. [Online; accessed 21-06-2021]. Disponível em: <<http://brasilecola.uol.com.br/curiosidades/a-origem-das-notas-musicais.html>>.

URTADO, Miguel. **Teoria Musical Básica**. n.Parte I. Disponível em: <[http://cifradventista.com/download/Teoria Musical/Apostila de Teoria Musical \[2.0\].pdf](http://cifradventista.com/download/Teoria%20Musical/Apostila%20de%20Teoria%20Musical%20[2.0].pdf)>.