

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

CARLOS EDUARDO PINHEIRO CORRÊA
LUCAS BERTAN MENEGASSI

DESENVOLVIMENTO DE ALGORITMO DE RECONHECIMENTO DE
EMOÇÕES POR PROCESSAMENTO DA VOZ

São Paulo
2008

CARLOS EDUARDO PINHEIRO CORRÊA
LUCAS BERTAN MENEGASSI

DESENVOLVIMENTO DE ALGORÍTMO DE RECONHECIMENTO DE
EMOÇÕES POR PROCESSAMENTO DA VOZ

Monografia apresentada à Escola
Politécnica da Universidade de São
Paulo para a obtenção de graduação
em Engenharia

São Paulo
2008

CARLOS EDUARDO PINHEIRO CORRÊA
LUCAS BERTAN MENEGASSI

DESENVOLVIMENTO DE ALGORÍTMO DE RECONHECIMENTO DE EMOÇÕES POR PROCESSAMENTO DA VOZ

Monografia apresentada à Escola Politécnica
da Universidade de São Paulo para a
obtenção de graduação em Engenharia

Área de Concentração: Engenharia
Mecatrônica

Orientador: Prof. Doutor Marcos Ribeiro
Pereira Barretto

São Paulo
2008

DEDICATÒRIA

Dedicamos este trabalho aos nossos queridos familiares.

AGRADECIMENTOS

Ao professor Marcos Barretto pelos esclarecimentos e a liberdade de pesquisa e elaboração que contribuíram para determinar as diretrizes deste trabalho.

Aos familiares que souberam compreender e apoiar nosso esforço e comprometimento.

Aos amigos André Gouvinhas, Felipe Brito, Gustavo Soares, Jonathas Fantasia e tantos outros que contribuíram na execução deste trabalho.

Não se pode resolver um problema
utilizando-se o mesmo tipo de
raciocínio que o criou.
(Albert Einstein)

RESUMO

Procurou-se estruturar um algoritmo que seja capaz de avaliar uma dada amostra de áudio produzida por um determinado locutor para, a partir de então, determinar em que estado emocional este se encontra. Para tanto, foi necessário construir uma base de dados (a qual denominamos 'corpus') a ser usada para comparação com a amostra em questão. O passo seguinte é aplicar fundamentos de Redes Neurais. Aqui é possível identificar duas frentes: a de aprendizado, que é a etapa na qual o programa é aplicado sobre o corpus para estudar as diferenças existentes entre cada um dos estados emocionais a serem identificados; e a de comparação propriamente dita, em que a amostra estudada é submetida a uma análise de variáveis para que se determine o estado emocional do locutor, sempre em relação à base de dados construída.

Palavras-chave: locutor, estado emocional, corpus, Redes Neurais

ABSTRACT

The aim of this study is to build an algorithm able to recognize the emotional content of the speech of a specific audio sample. Therefore, it is needed to first compose a database (called 'corpus') for being used at the comparison with the studied sample. The next step is to apply the Neural Network fundamentals. At this moment it is possible to identify two different research issues: the learning step, which consists in applying the program over the entire database, so that the existing difference between each emotional standard can be detected and measured; and also the comparison between the studied sample and the corpus, so that the variables of the audio sample are analyzed and the program can find out which one is the most appropriated emotion that fits the sample.

Keywords: speaker, emotional speech, corpus, Neural Network

SUMÁRIO

1. Introdução	11
1.1. Apresentação do Tema	11
1.2 Objetivos	13
1.2.1. Objetivo Geral.....	13
1.2.2. Objetivos Específicos	13
2. Voz Humana	14
2.1. Aspectos Fisiológicos da Voz.....	14
2.2. As Propriedades da Voz.....	17
2.2.1. Extensão Fonatória.....	17
2.2.2. Frequência Fundamental.....	17
2.2.3. Intensidade	17
2.2.4. Ruído Espectral	17
2.2.5. Ressonância	18
3. Psicologia e Modelagem de Emoções	19
3.1. O Cérebro e as Emoções.....	19
3.2. Os Parâmetros do Som escolhidos e As Emoções modeladas	20
4. Algoritmos Desenvolvidos.....	21
4.1. Estrutura dos Algoritmos	21
4.2. Cálculo dos Parâmetros do Sinal de Voz	23
4.2.1. Aquisição do Sinal	23
4.2.2. Importação dos Dados para o MatLab.....	23
4.2.3. Segunda Discretização – Transformação dos Dados.....	24
4.2.5. Cálculo da Freq. Fundamental e Transf. De Fourier.....	26
4.2.6. Cálculo da Intensidade – RMS (Root Mean Square)	27
4.2.7. Identificação do Silêncio	28
4.2.8. Identificação de Segmentos Contínuos de Voz.....	29
4.2.9. Parâmetros Observados.....	29
4.2.10. Fluxograma Resumido da Função.....	33
4.3. Aprendizado	33
4.3.1. Conceitos de Redes Bayesianas.....	33
4.3.2. Modelagem do Problema.....	34
4.3.3. Corpus	35
4.3.4. Algoritmo analisa_população.....	36
4.4. Análise de um indivíduo	38
4.4.1. Princípio de Funcionamento	38
4.4.2. Exemplo.....	39
4.4.2. Fluxograma Resumido da Função.....	41

4.5. Teste	41
4.5.1. Princípio de Funcionamento	41
4.5.2. Fluxograma resumido da função	42
5. Resultados e Discussão.....	43
5.1. Aprendizado e Teste	43
5.1.1. Caso I	43
5.1.2. Caso II	44
5.1.3. Caso III	45
5.2. Aplicação de Pesos aos Parâmetros.....	47
6. Conclusão	49
7. Considerações Finais.....	50
8 ANEXOS	
Anexo 1 – Função: segmentar.....	51
Anexo 2 – Função: descarta.....	52
Anexo 3 – Função: frequencia.....	53
Anexo 4 – Função: intensidade	54
Anexo 5 – Função: silencio.....	55
Anexo 6 – Função: transicao	56
Anexo 7 – Função: normalizar	57
Anexo 8 – Função: parametros.....	58
Anexo 9 – Função: calcula_parametros	61
Anexo10 – Tabela do corpus.....	62
Anexo 11 – Função: analisa_população.....	62
Anexo 12 – Função: analise_individuo	66
Anexo 13 – Função: teste_populacao	68
Anexo 14 – Matriz de Nível (Algoritmo Final).....	69
Anexo 15 – Matriz de Distribuição (Algoritmo Final)	70
Anexo 16 – Pesos (Algoritmo Final).....	72
9. Referências Bibliográficas.....	73

1. Introdução

1.1. Apresentação do Tema

A comunicação humana é sujeita de forma direta à influência de fatores diversos. Ela pode ser dividida basicamente quanto aos seguintes componentes: emissor, receptor, mensagem e ambiente. A combinação de características dos quatro componentes é o que determina o padrão da conversação.

Tanto o emissor quanto o receptor têm a capacidade de incrementar a conversação com elementos verbais, como volume e entonação da voz, e não verbais (não-léxicos), como gestos e expressões faciais. A forma com a qual o sujeito faz uso destes recursos tem relação muito íntima com a maneira que ele pensa e se sente no momento da fala. Sendo assim, estudar a expressão do locutor pode trazer como resultado o estado emocional e psicológico no qual ele se encontra.

Determinar o estado emocional com base na análise da voz é um tema muito estudado por diversas frentes científicas na atualidade. A literatura a respeito do assunto é muito vasta, e são diversos os métodos propostos, o que dificulta a definição de um padrão para a realização deste estudo. Ao mesmo tempo, a forma com a qual a emoção é expressa na voz também pode variar de uma língua para outra. Este aspecto torna-se mais latente ao comparar, por exemplo, a comunicação em línguas ocidentais e orientais.

No entanto, existe uma métrica padrão desses estudos; observa-se que aspectos perceptuais ou qualitativos da voz, como voz “tensa”, “frouxa”, “metálica” ou “suave”, podem ser traduzidos em variáveis numéricas mensuráveis que dizem respeito às propriedades acústicas do sinal sonoro.

Evidentemente, esses parâmetros variam de acordo com cada indivíduo; por exemplo, o timbre da voz masculina é diferente do que se observa na voz feminina. No entanto, independentemente do locutor, os parâmetros da voz estão compreendidos dentro de domínios limitados pelos aspectos fisiológicos do ser humano, tais como composição das cordas vocais, capacidade pulmonar ou velocidade de raciocínio.

Tais limites são vastamente conhecidos e estudados pela ciência moderna. Assim, notamos que, para uma determinada amostra de som, é possível destrinchar os parâmetros de interesse para análise e, realizando comparações com um banco de dados de voz (que neste trabalho será denominado “*corpus*”) previamente construído, dizer para aquela determinada amostra de voz estudada o estado emocional no qual o locutor se encontra.

Neste escopo, a engenharia está presente na aplicabilidade de estudos de redes neurais artificiais, que nada mais são do que sistemas computacionais com propriedades particulares como capacidade de aprendizagem ou de adaptação aos dados utilizados durante a sua aplicação, a capacidade de reconhecer dados ‘inéditos’ ao sistema, além de tolerância a falhas.

O “*corpus*” será o banco de dados que será usado como base de comparação com a amostra de análise. Para a sua construção, recolhemos amostras de vozes referentes aos estados emocionais propostos neste estudo.

Esta etapa foi necessária pela dificuldade de se captar amostras de voz legítimas e originais por parte do locutor nos estados emocionais que desejamos estudar, uma vez que emoções são espontâneas e imprevisíveis quanto ao instante em que elas irão se manifestar. Além disso, como o programa utilizado para análise é muito sensível a ruídos amostrais, é importante que se mantenha um padrão quanto à aparelhagem utilizada na aquisição, o que impossibilita, por exemplo, o uso de um gravador portátil de voz, visto que a baixa qualidade do sinal obtido em um aparelho tão simples poderia ter influência direta no resultado final do estudo.

Logo, tendo a amostra da voz a ser estudada (armazenada em um arquivo no formato wave comum), os parâmetros do som e os estados emocionais de análise definidos, e o “*corpus*” construído, chegamos à etapa de implementação das Redes Neurais. O treinamento das redes era feito baseado no sinal no domínio do tempo, onde o sinal era analisado na sua forma original pela rede. Entretanto, para a grande parte das aplicações em processamento digital de sinais a representação no domínio do tempo não é a mais adequada, uma vez que as informações de interesse do sinal estão contidas no espectro de frequência, que mostra as frequências existentes no sinal que estará sendo processado. Logo, o tratamento do sinal e sua utilização no domínio

da frequência farão possível armazenar as características fundamentais específicas de cada voz para utilizá-las no treinamento das Redes Bayesianas.

1.2 Objetivos

1.2.1. Objetivo Geral

Dada a apresentação inicial, a principal proposta deste trabalho de conclusão de curso é elaborar um algoritmo no software MatLab [4] que seja capaz de digitalizar e fazer o processamento completo de um determinado sinal de voz para, por meio da aplicação de conceitos de Redes Bayesianas, determinar o estado emocional no qual os padrões da amostra se encaixam melhor, obtendo o estado emocional de maior probabilidade.

1.2.2. Objetivos Específicos

Visamos ao longo do desenvolvimento atingir as seguintes etapas:

- levantar parâmetros acústicos que sejam relevantes na inferência do estado emocional em que o locutor se encontra.
- construir um banco de dados (*corpus*) que seja confiável e eficiente na comparação com a amostra adquirida
- estudar conceitos e aplicações de Redes Bayesianas na identificação de padrões de um sinal de voz
- elaboração e implementação do código no software MatLab

2. Voz Humana

2.1. Aspectos Fisiológicos da Voz

A voz, como já dissemos, é o principal instrumento de comunicação do homem, sendo o recurso mais imediato utilizado pelo ser humano para expressar seus pensamentos e emoções. Mais do que isso, a voz faz parte da identidade de cada pessoa, sendo uma das extensões mais fortes de nossa personalidade.

Alguns aspectos da voz, como intensidade, frequência, ressonância, entre outros, são na verdade manifestações físicas de aspectos fisiológicos do locutor, conforme veremos mais à frente. Stetson(1928) citou que “fala é o movimento que se faz audível”. O movimento referido é o das pregas (cordas) vocais (*Figura 1 e Figura 2*), que são fibras de um tecido tensionado e que vibram com a passagem do fluxo controlado de ar pela glote, emitindo som fundamental e seus harmônicos. O som glótico é modulado na fala pelos efeitos da ressonância e da articulação.

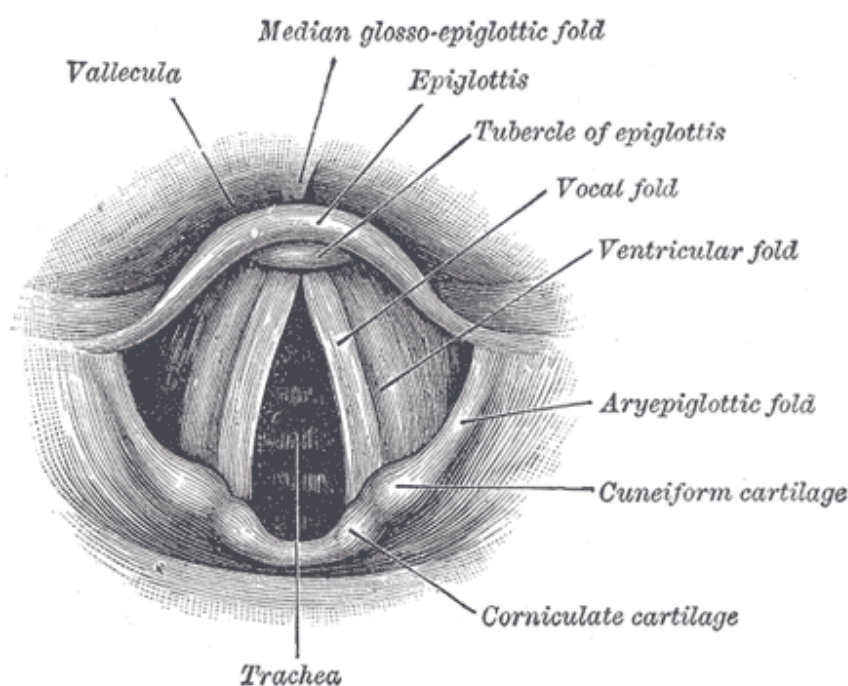


Figura 1 - Desenho esquemático das pregas vocais.

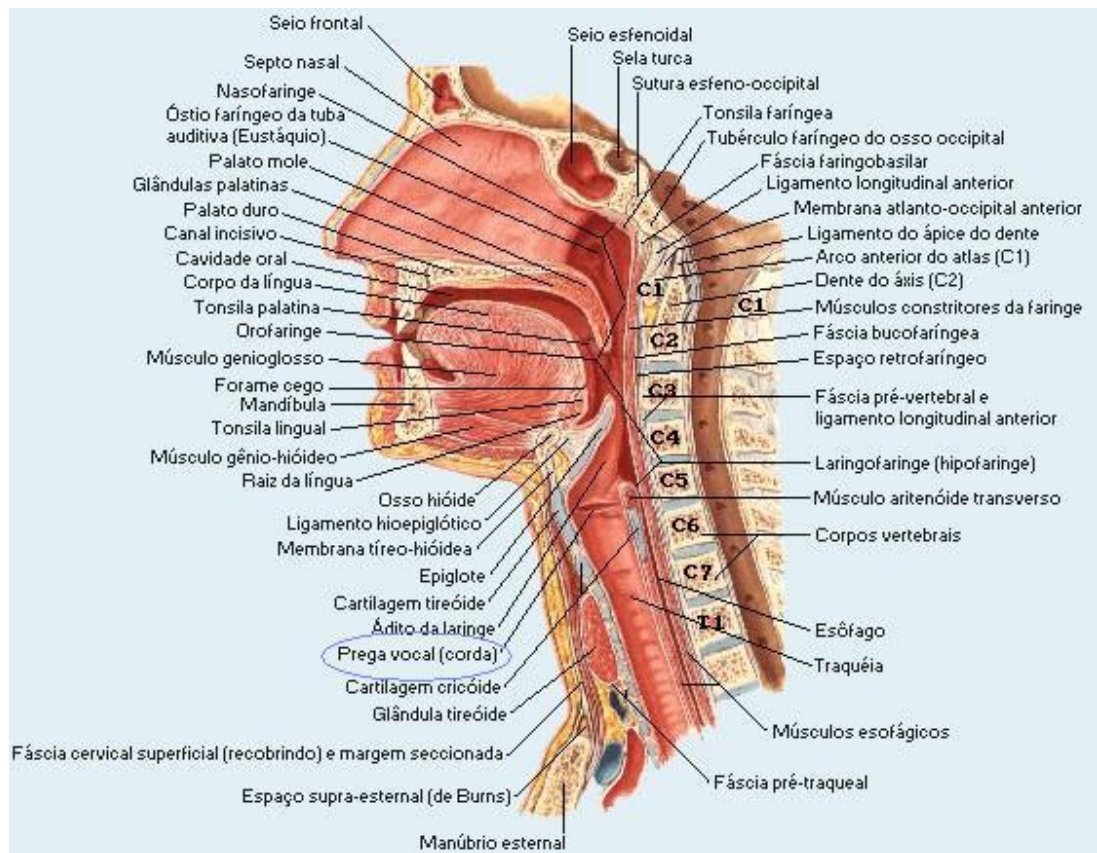


Figura 2 - Fisiologia do conjunto cabeça, pescoço e extremidade superior.

Durante a respiração normal, as pregas vocais estão bastante espaçadas entre si e, ao contrário do que se afirma com frequência, esse espaçamento não é menor durante a expiração do que durante a inspiração. Em qualquer velocidade, a corrente de ar não sofre resistência quando flui para dentro ou para fora dos pulmões. De forma breve, a laringe produz sinais sonoros glóticos gerando uma série rápida de pulsos de ar de curta duração, que excitam a coluna de ar supra-laríngea para produzir um sinal sonoro complexo.

O ciclo de vibração das pregas vocais pode ser explicado da seguinte forma: as cordas são aduzidas, parcial ou totalmente, de forma a restringir o fluxo do ar que sai dos pulmões; ao mesmo tempo, as forças de expiração produzem uma quantidade crescente de pressão de ar abaixo das pregas e, quando ela se torna suficiente, elas são literalmente sopradas para longe uma da outra, liberando assim um sopro de ar para o trato vocal. A liberação do ar provoca uma queda na pressão abaixo das pregas

vocais; juntamente com a elasticidade do tecido das pregas, esta queda de pressão permite que elas retornem à posição aduzida, prontas para serem “sopradas” novamente, uma vez que o pico de pressão se forme.

As pregas vocais, que primordialmente compreendiam apenas as funções de respiração, alimentação e esfícteriana, com o passar dos tempos se desenvolveram e passaram a ser usadas na produção da voz. Podem vibrar a frequências surpreendentemente altas, indo de 125 ciclos por segundo para homens com a voz mais grossa até 250 ciclos por segundo para mulheres de voz mais aguda. Outra diferença existente entre homens e mulheres é que as pregas masculinas têm mais massa e são menos esticadas que as pregas femininas. É possível inclusive a analogia das pregas às cordas de um violão, que emitem um som tão grave quanto menos esticada a corda estiver, ou mais massa ela apresentar.

A laringe, que é o espaço existente entre a epiglote e as pregas vocais, é o canal por onde o ar passa durante toda a respiração. Ela contém músculos que esticam ou relaxam as pregas vocais. De acordo com o grau de abertura das pregas e a quantidade de ar exalada, o som produzido será de diferentes qualidades (*Figura 3*).

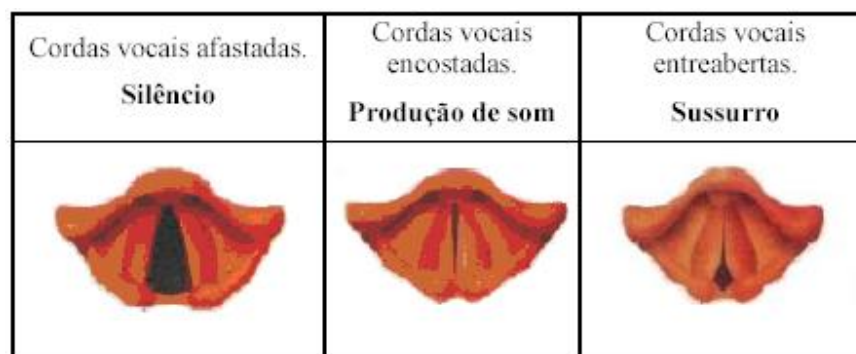


Figura 3 - Diferentes configurações das cordas vocais.

Sendo assim, o cérebro, ao receber um determinado estímulo externo, reage emitindo várias mensagens em forma de impulsos elétricos a diversas regiões do corpo humano, entre elas os músculos da laringe, do tórax e diafragma. Estes músculos, por sua vez, se contraem ou relaxam, de acordo com o impulso recebido, produzindo as referidas alterações fisiológicas. A movimentação deste conjunto de forma coordenada

é o que faz possível com que o ser humano se manifeste de acordo com o estímulo externo recebido.

2.2. As Propriedades da Voz

2.2.1. Extensão Fonatória

A voz normal tem variabilidade da frequência fundamental (que nada mais é do que a frequência média de vibração das pregas vocais) durante a conversação casual. Os sons da fala seriam muito monótonos sem ela.

Um adulto normalmente pode produzir tons que se estendem em uma faixa de frequência de uma a duas oitavas acima do tom mais baixo sustentável. Usando um teclado ou um diapasão, é possível determinar o tom mais grave que o indivíduo pode sustentar. O tom mais agudo, geralmente, atinge de uma a duas oitavas acima do mais baixo.

2.2.2. Frequência Fundamental

É a frequência média de oscilação das pregas vocais. Alguns fonoaudiólogos baseiam-se no conceito de que a frequência fundamental (ou ideal, ou habitual) situa-se em um quarto da extensão fonatória total [6]. A frequência ideal está entre 17% e 29% a partir dos graves, na extensão fonatória total.

2.2.3. Intensidade

Nas frequências médias, o NPS - Nível de Pressão Sonora - está em torno de 50 dB. Esta propriedade diz respeito à potência, ou volume, conferida ao som produzido.

2.2.4. Ruído Espectral

Também chamado de aspereza, é a qualidade vocal como consequência da periodicidade ou da distribuição aleatória da energia acústica no espectro vocal. Diz respeito à qualidade vocal detectada, definida e descrita com base nas impressões auditivas do ouvinte. As qualidades da voz associadas à aspereza são voz rouca, aguda ou chiada.

2.2.5. Ressonância

É um aspecto da voz que não está associado à laringe ou às pregas vocais, mas sim às vias de acesso entre a própria laringe e a abertura da boca.

O grau de nasalidade da voz, que é o fator que está mais diretamente relacionado ao fenômeno da ressonância na fala, depende da extensão do acoplamento das passagens nasais para a cavidade oral e faríngea. A caixa craniana exerce também a função de amplificador acústico, para o som gerado nas pregas vocais.

3. Psicologia e Modelagem de Emoções

3.1. O Cérebro e as Emoções

As determinações emocionais nos remetem tanto ao plano motor como também ao plano lingüístico. Antes de analisá-las, é preciso considerar que as emoções se manifestam a partir das condições do próprio organismo (sono, fome, dor, doença, etc.), ou ainda daquilo que a pessoa pensa sobre si mesma e sobre as situações sociais que enfrenta, em função dos seus próprios valores e crenças sobre o mundo e as coisas.

Devemos considerar, de acordo com Sherer(1981), que as emoções experimentadas pelo indivíduo manifestam-se sempre por meio de modificações fisiológicas. Essas manifestações podem variar desde a aceleração do batimento cardíaco e do ritmo respiratório até frio no peito ou na barriga, ou ainda tensão em diferentes partes do corpo (musculatura abdominal, do pescoço, dos ombros, do rosto, entre outras). De acordo com o tipo e intensidade da emoção experimentada, a atividade da fala pode ser consideravelmente influenciada, favorecendo a produção de quebras, repetições e hesitações. Observa-se, por exemplo, de modo consistente, que certas emoções, como vergonha ou medo, têm um potencial para produzir gagueira, enquanto outras, como a raiva, para produzir fluência.

Do ponto de vista cognitivo, devemos considerar que emoção e cognição são inversamente proporcionais, ou seja, quanto mais uma dada emoção está ativada, mais os seus pensamentos assumem o seu “colorido”, e por sua vez, menos o indivíduo estará envolvido em atividades racionais.

Há uma indiscutível relação entre a linguagem e o cérebro, baseada nas atividades cognitivas e as interações nas áreas do córtex. No córtex cerebral essas áreas estão separadas devido as suas funções. A fala é controlada pelo hemisfério esquerdo do cérebro em uma porcentagem dos seres humanos maior que noventa e cinco por cento, segundo Lent (2001), enquanto o hemisfério direito é o encarregado de fazer a organização da prosódia.

A prosódia é um componente afetivo da linguagem que consiste na entonação da fala e em gesticulações emocionais. Estes aspectos afetivos da linguagem estão

representados no hemisfério direito, responsável pelo caráter emocional do discurso, assim como pela ironia, sarcasmo e humor aplicados a ele, e que sua organização anatômica espelha aquela para a linguagem cognitiva no hemisfério esquerdo [3].

Assim, a relação entre os dois hemisférios cerebrais possibilita que um mesmo fonema possa ser produzido de várias formas, dependendo do contexto da frase e do conteúdo emotivo.

3.2. Os Parâmetros do Som escolhidos e As Emoções modeladas

Foi determinado seis categorias de parâmetros para comparação, que parecem ser muito eficientes como referência para a determinação da emoção. São eles: *freqüência fundamental, intensidade, duração do som, dinâmica da freqüência fundamental, freqüência fundamental normalizada e esforço vocal*.

Para obter resultados satisfatórios e confiáveis ao final do reconhecimento do estado emocional, optamos por emoções bem distintas e que apresentem diferenças notáveis em cada um desses parâmetros estudados.

Assim, as emoções que modelamos são: *neutro, alegre, triste e bravo*.

4. Algoritmos Desenvolvidos

4.1. Estrutura dos Algoritmos

Para realização deste trabalho, foram desenvolvidos diversos “algoritmos principais”, cada um com uma função específica. Abaixo segue a sequência e funcionalidade de cada um, para que se possa compreender de maneira macro como o trabalho foi estruturado.

calcula_parametros: Analisa a amostra de voz de um indivíduo e calcula os parâmetros estatísticos, relacionados as características acústicas da amostra.

Inputs: amostra de som (data) e frequência de amostragem (fs)

Outputs: vetor com todos os parâmetros estatísticos (P)

analisa_populacao: Este algoritmo é o que realiza o aprendizado. Com base em uma população ele obtém a distribuição dos parâmetros por estado emocional, construindo uma base de comparação

Inputs: Conjunto de amostras com os parâmetros já calculados (Ma)

Outputs: Níveis em que os parâmetros estão distribuídos (Nível) e a frequência em que eles ocorrem para cada nível por estado emocional (Dist)

analise_individuo: Compara os parâmetros de uma amostra de som com uma população previamente analisada e determina as probabilidades para cada estado emocional. Indica posteriormente o de maior probabilidade. Este algoritmo também considera os pesos atribuídos para cada parâmetro.

Inputs: parâmetros (P), nível e distribuição da população de referência (Nível Dist) e os pesos atribuídos para cada parâmetro (Pesos)

Outputs: as probabilidades para cada estado emocional (emocao) e a indicação do estado emocional de maior probabilidade (resposta)

teste_populacao: Este algoritmo é o que realiza o teste do aprendizado. Baseado nos níveis e distribuições de uma população previamente analisada, classifica uma outra população (podendo ou não ser a mesma do aprendizado) nos estados emocionais de maior probabilidade. Ao final ele retorna a comparação por estado emocional, do resultado desejado contra o resultado obtido. Este resultado é conhecido como “Matriz de Confusão”, um conceito bastante divulgado na discussão de resultados obtidos na aplicação de redes bayesianas.

Inputs: nível e distribuição da população de aprendizado (Nivel, Dist), os pesos atribuídos para cada parâmetro (Pesos) e a população em que se deseja realizar o teste (Ma).

Outputs: matriz de confusão comparando o resultado obtido com o resultado desejado (resposta)

Posteriormente, serão explorados os detalhes de cada algoritmo, as subfunções que os compõem e também as variáveis descritas acima.

4.2. Cálculo dos Parâmetros do Sinal de Voz

4.2.1. Aquisição do Sinal

O sinal de áudio será captado por um microfone conectado diretamente com o computador. Utilizando o software Audacity (freeware) o sinal será gravado digitalmente em um arquivo tipo wave, em canal mono e taxa de amostragem de 8kHz (taxa recomendada e suficiente para análise de sinais vocais humanos). Na figura abaixo (*Figura 4*), podemos visualizar um sinal de áudio gravado neste software.

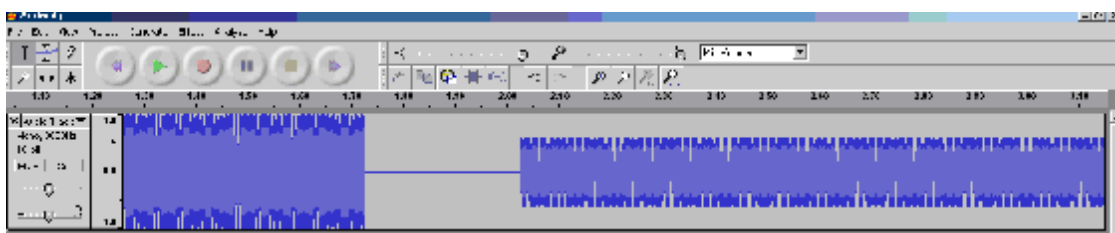


Figura 4 - Exemplo de sinal de áudio capturado pelo software Audacity.

Este software de fácil manuseio possui inúmeras ferramentas para o tratamento do sinal adquirido. Com ele é possível eliminar ruídos indesejáveis e adicionar filtros de frequência quando necessários, além de muitos outros plug-ins que podem ser úteis no tratamento do som. O software permite ainda exportar o arquivo para diversos formatos.

4.2.2. Importação dos Dados para o MatLab

No menu File do MatLab, pode-se importar diretamente o arquivo no formato wave. O programa retorna um vetor data – contendo números entre -1 e 1 com precisão double – e um número fs que é a frequência de amostragem do arquivo original (8 kHz no caso).

No poderoso ambiente do MatLab, é possível processar o sinal de voz para obter os parâmetros necessários.

4.2.3. Segunda Discretização – Transformação dos Dados

Os parâmetros calculados serão determinados para cada décimo de segundo da amostra, este período será denominado “período de discretização” - $TD = 0,1$ segundos. Para melhor organizar os dados e facilitar a manipulação, uma matriz \underline{S} é criada da seguinte forma:

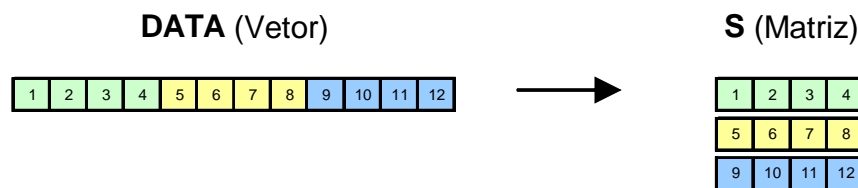


Figura 5 – Transformação do vetor em matriz.

O algoritmo **segmentar** transforma o vetor data em uma matriz \underline{S} , onde cada linha corresponde a um período de discretização TD . Portanto se a taxa de amostragem f_s do sinal é de 8kHz, a matriz \underline{S} terá sempre 800 colunas. ($f_s \times TD = 800$).

O número de linhas da matriz pode ser obtido dividindo-se o tempo amostrado pelo período de discretização.

No caso de um sinal de 10 segundos de duração teremos uma matriz \underline{S} de dimensão 100 x 800.

A coerência deste algoritmo pode ser facilmente verificada. Basta escolher um ponto de data e analisar o ponto em \underline{S} correspondente. Exemplo: data(801) deve ser igual a $\underline{S}(2,1)$.

O código pode ser visto em anexo (*Anexo 1*)

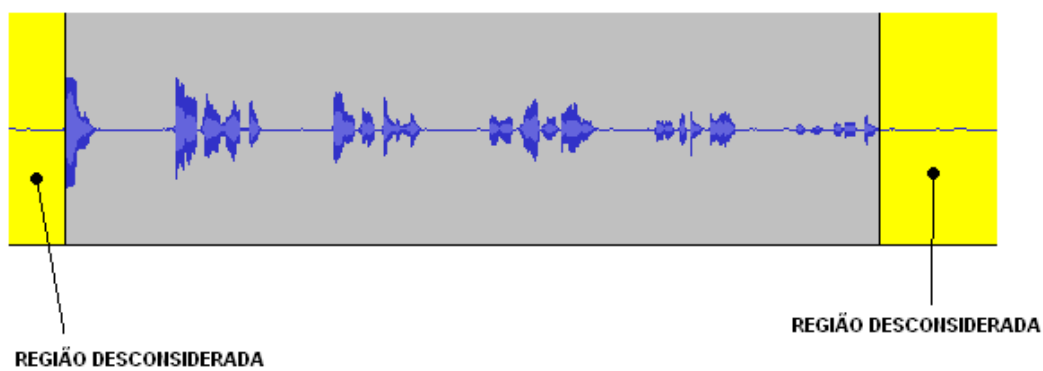
4.2.4. Descarte dos Períodos Iniciais

Para equalizar todas as amostras, antes dos parâmetros serem calculados, criou-se uma função que descarta o período inicial e final de silêncio, correspondente ao “delay” entre o gravador e locutor.

O objetivo aqui é que este delay não afete as variáveis relacionadas à dinâmica e velocidade do som.

Segundo os critérios das funções **intensidade** e **silêncio**, que serão descritos adiante, podemos observar na *Figura 6* abaixo, o funcionamento da função **descarta**. O código pode ser visto em anexo (*Anexo 2*).

AMOSTRA ORIGINAL



AMOSTRA CONSIDERADA

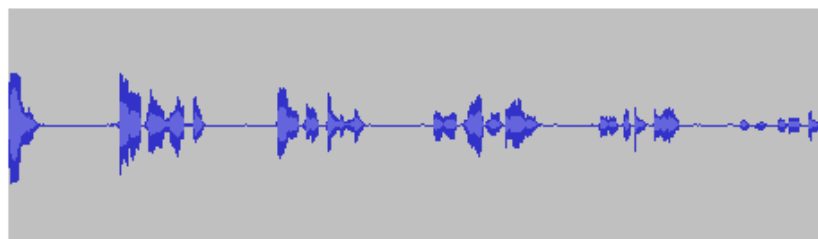


Figura 6 – Períodos desconsiderados.

4.2.5. Cálculo da Freq. Fundamental e Transf. De Fourier

O parâmetro mais importante analisado é a frequência fundamental F_0 (também é o de cálculo mais complexo). Para uma onda senoidal perfeita poderíamos facilmente determinar a sua frequência olhando simplesmente seu gráfico no tempo, assim obteríamos seu período e frequência correspondente. Porém, para sinais de áudio do mundo real, não temos apenas uma frequência, e sim uma mistura de várias frequências e ruídos. Olhando seu sinal através do tempo é praticamente impossível determinar sua frequência fundamental, conforme pode-se observar na *Figura 7*.

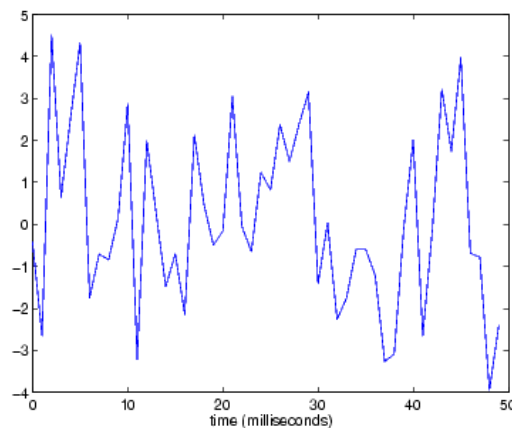


Figura 7 - Sinal composto por diferentes frequências e ruído adicionado

A solução é partir para uma análise no domínio da frequência [5], utilizando uma ferramenta poderosa que é a Transformada de Fourier. Esta transformada recebe um sinal de entrada no domínio do tempo, e retorna um sinal (número imaginário) no domínio da frequência. Conforme equação:

$$F(\omega_p) = \frac{1}{2N} \sum_{p=0}^{2N-1} f(t_k) e^{i\omega_p t_k}$$

O princípio da Transformada de Fourier é tentar compor o sinal de entrada por uma soma de sinais periódicos. Através da função no domínio da frequência, podemos determinar o módulo e fase para cada valor de frequência.

A frequência fundamental é a de maior amplitude. Na *Figura 8* pode-se visualizar o espectro de frequências de um sinal adquirido. As frequências principais desse sinal são 50 Hz e 120 Hz:

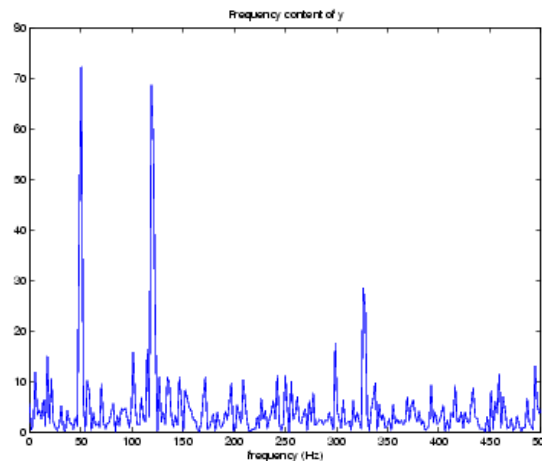


Figura 8 - Espectro de frequências

O algoritmo **frequencia** encontra através da Transformada de Fourier a frequência fundamental, retornando um vetor F0.

Através de alguns testes verificamos, que para as frequências amostradas, o período mínimo de observação para aplicação da Transformada de Fourier é de 1 segundo. Desenvolvemos o algoritmo de maneira a “varrer” toda amostra, analisando sempre um período de 1 segundo, porém com incrementos de 0,1 segundos.

Resumidamente, o algoritmo executa a transformada numa frequência de 10Hz, porém analisando sempre um período de 1 segundo. Desta maneira neutralizamos distúrbios pontuais que poderiam causar resultados indesejados e inconsistentes. A resposta final (vetor de frequências F0) torna-se mais “suave” – menores oscilações entre os instantes vizinhos. O código pode ser visto em anexo (*Anexo 3*).

4.2.6. Cálculo da Intensidade – RMS (Root Mean Square)

A função **intensidade** calcula a raiz da média dos quadrados para cada linha de S e retorna um vetor RMS com os correspondentes valores. Este tipo de cálculo é feito

também na análise de tensões elétricas senoidais de correntes alternadas, sendo que sua intensidade (RMS) é o valor da tensão eficaz.

Traduzindo para linguagem matemática:

$$RMS_i = \sqrt{\frac{\sum_{j=1}^c S_{ij}^2}{c}}$$

O código está disponível em anexo (*Anexo 4*).

4.2.7. Identificação do Silêncio

Nesta análise, para o cálculo de alguns parâmetros, é fundamental classificar os segmentos em duas categorias: voz ativa e silêncio. Com este objetivo foi criado um vetor *flag*.

Para cada segmento do áudio, a função **silencio** atribui zero ou um, para o silêncio e voz ativa respectivamente. O período será considerado silencioso quando sua intensidade for inferior 0,05. Resumidamente temos:

$$RMS_i < 0,05 \Rightarrow flag_i = 0$$

$$RMS_i \geq 0,05 \Rightarrow flag_i = 1$$

O código está disponível em anexo (*Anexo 5*).

4.2.8. Identificação de Segmentos Contínuos de Voz

Através da função **continuidade** podemos detectar quando há uma transição do silêncio para voz e vice-versa (fenômenos de borda). Possibilitando a análise dos parâmetros em segmentos contínuos de voz.

O algoritmo recebe o vetor flag, e retorna um vetor transicao que atribui 1 quando há transição do silêncio para voz ativa, -1 quando há transição da voz ativa para o silêncio e zero quando não há transição alguma. A tabela abaixo (*Figura 9*) ilustra a lógica implementada:

flag(i-1)	flag(i)	transicao(i)
0	0	0
0	1	1
1	0	-1
1	1	0

Figura 9 – Tabela lógica da função transição.

O código está disponível em anexo (*Anexo 6*).

4.2.9. Parâmetros Observados

Agora que já somos capazes de obter as principais características sonoras (frequência, intensidade e duração), é possível calcular os parâmetros mais significativos a serem utilizados na posterior inferência estatística do estado emocional.

A função **parâmetros** recebe os vetores F0, RMS, flag e transicao e retorna um vetor P que armazena as variáveis indexadas abaixo. Estes parâmetros podem ser classificados em 6 grupos:

a) Relacionados à frequência fundamental:

- Média – F0mean (P1)
- Mediana – F0med (P2)
- Máximo – F0max (P3)
- Mínimo – F0min (P4)

- Fracional correspondente a 95% da amostra - F095th (P5)
- Fracional correspondente a 5% da amostra - F05th (P6)
- Amplitude – F0range (P7)
- Amplitude entre os fracionais 5%-95% – F05th-95thrange (P8)
- Desvio Padrão – F0stdev (P9)

b) Relacionados à intensidade:

- Média – RMSmean (P10)
- Mediana – RMSmed (P11)
- Máximo – RMSmax (P12)
- Fracional correspondente a 95% da amostra – RMS95th (P13)
- Fracional correspondente a 5% da amostra - FRMS5th (P14)
- Amplitude entre os fracionais 5%-95% – RMS5th-95thrange (P15)
- Desvio Padrão – RMSstdev (P16)

c) Relacionados à duração do som:

- Duração média de um segmento contínuo de voz – ONmean (P17)
- Duração média das pausas – OFFmean (P18)
- Desvio padrão das pausas – OFFstdev (P19)
- Razão voz/silencio – ONOFFratio (P20)

d) Relacionados à dinâmica da frequência fundamental:

- Incremento médio de um segmento contínuo de voz – F0UPmean (P21)
- Incremento máximo de um segmento contínuo de voz – F0UPmax (P22)
- Desvio padrão do incremento de F0UP – F0UPstdev (P23)
- Decremento médio de um segmento contínuo de voz – F0DOWNmean (P24)
- Decremento máximo de um segmento contínuo de voz – F0DOWNmax (P25)
- Desvio padrão do incremento de F0DOWN – F0DOWNstdev (P26)
- Razão entre incremento e decremento – UPDOWNratio (P27)

e) Relacionadas a frequência fundamental normalizada:

Dentro da função **parâmetros**, existe uma subrotina chamada **normalizar**, que é aplicada ao vetor das freqüências F0. Esta rotina tem o objetivo de colocar em um mesmo patamar a freqüência natural dos indivíduos (que costuma variar muito de um indivíduo para outro).

Por exemplo, comparar uma amostra de uma mulher de voz aguda com a de um tenor de ópera seria inviável olhando simplesmente o valor absoluto da freqüência fundamental.

Esta normalização descarta as frações de 5% inferiores e superiores da amostra para melhorar a homogeneidade da análise (*Figura 10*).

O código da função normalizadora pode ser consultado em anexo (*Anexo7*).

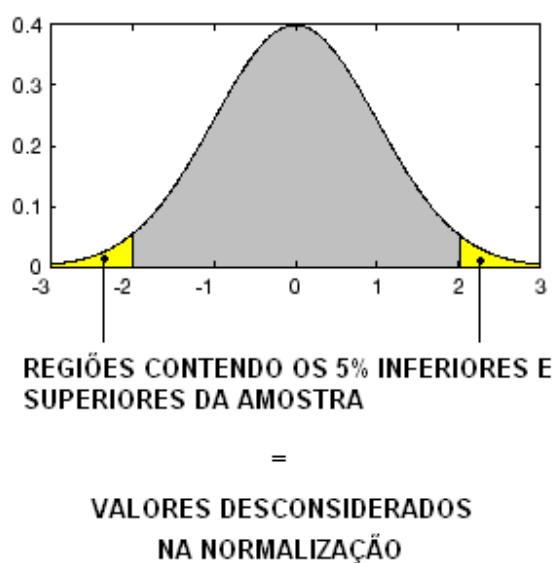


Figura 10 – Valores desconsiderados.

Depois disso, os valores que restarão são transformados aplicando-se a seguinte fórmula:

$$f0_n(i) = \frac{F0'(i) - F0'_{\min}}{F0'_{\max} - F0'_{\min}}$$

Para este novo vetor de frequência normalizada f0n, calcula-se as seguintes estatísticas.

- Média – f0n_mean (P28)
- Mediana – f0n_med (P29)
- Desvio Padrão – f0n_std (P30)

f) Esforço Vocal:

Este parâmetro é um indicativo da energia imposta na voz pelo locutor. Sua fórmula é seguinte:

$$EsforçoVocal = \frac{\sum_i F0_i \cdot RMS_i}{F0_{médio} \cdot RMS_{médio}} \quad P(31)$$

O código da função **parametros** pode ser visto em anexo (*Anexo 8*).

OBS: É importante ressaltar que os parâmetros dos itens 'a', 'b', 'd', 'e' e 'f' não levam em consideração os períodos de silêncio, apenas os segmentos com voz.

4.2.10. Fluxograma Resumido da Função

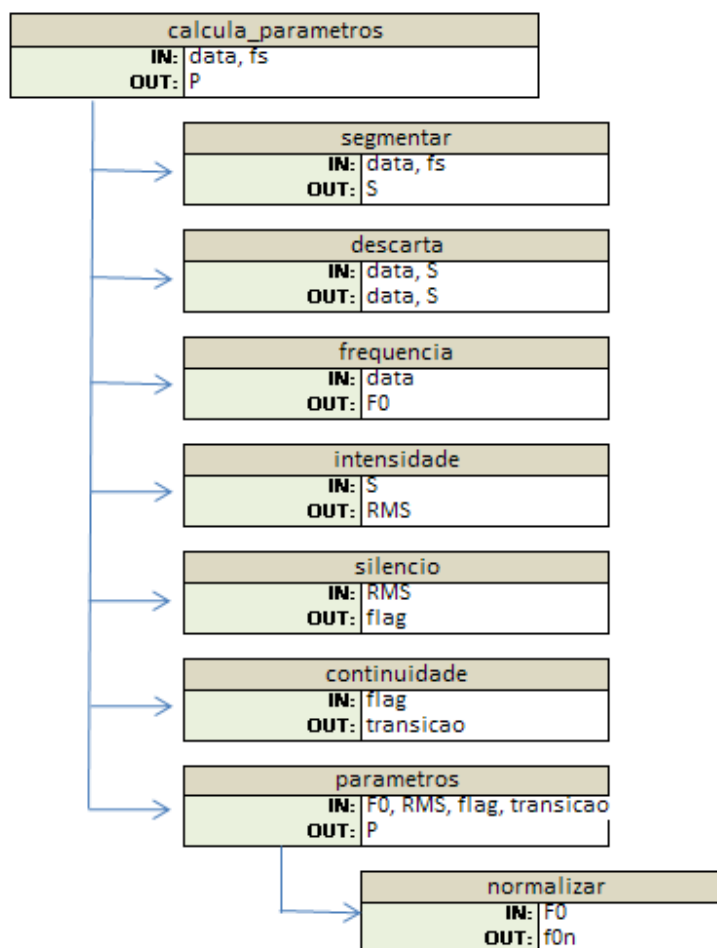


Figura 11 – Fluxograma: calcula_parametros

O código pode ser visto em anexo (*Anexo 9*).

4.3. Aprendizado

4.3.1. Conceitos de Redes Bayesianas

As redes Bayesianas [2] tem como objetivo estabelecer relações de dependências entre variáveis aleatórias para simplificar ao máximo problemas que envolvem um número grande de variáveis aleatórias. Por exemplo, considere as

variáveis binárias X , Y e Z . Precisariamos de 8 valores de probabilidade para analisar o problema, um para cada combinação de variáveis (na verdade 7 pois o 8º pode ser obtido através da propriedade de que a soma de todas as probabilidades deve ser igual a 1).

Suponha agora que Y depende única e exclusivamente de X , e que Z depende única e exclusivamente de Y . Representação:

$$X \rightarrow Y \rightarrow Z$$

Agora precisamos apenas de cinco valores para analisar o problema, um para $P(X)$, dois para $P(Y|X)$ e dois para $P(Z|Y)$. Perceba que Z é independente de X . Exemplo: Se X é a probabilidade de João jogar bola, e Y é a probabilidade de João quebrar a perna, portanto Y depende de X . Z é a probabilidade de João precisar de um hospital, portanto Z depende de Y . Note que a probabilidade de João precisar de um hospital não está mais relacionado com o fato dele ter jogado futebol ou não.

4.3.2. Modelagem do Problema

Para nossa aplicação, criaremos uma Rede Bayesiana em que os parâmetros observados são dependentes do estado emocional em que o indivíduo se encontra. Ou seja, se uma pessoa está furiosa esperamos encontrar uma intensidade alta de voz e um intervalo curto de pausas.

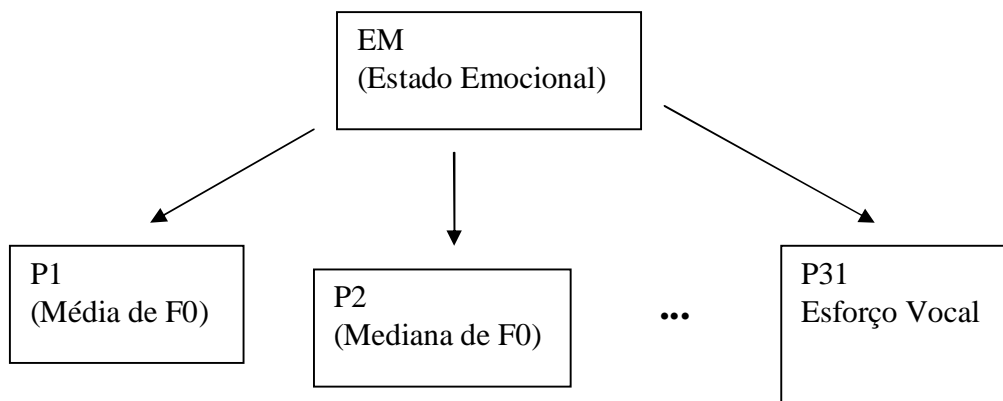


Figura 12 – Modelagem das Redes Bayesianas

Assim vinculamos todos os parâmetros ao estado emocional do indivíduo. Se soubermos todas as probabilidades $P (P_i | EM)$, será possível realizar uma inferência sobre EM ao observar todos os parâmetros de P.

4.3.3. Corpus

O corpus nada mais é do que um banco de dados robusto no qual o computador realiza observações para efetuar o aprendizado. Genericamente, fornecemos este conjunto de dados e determinamos para o computador, qual é o resultado desejado para cada conjunto de inputs.

Para nosso problema específico, em cada amostra de voz, tínhamos como dados de entrada os parâmetros \underline{P} e também o estado emocional. Assim o computador é capaz de reconhecer um padrão entre esses parâmetros para cada categoria de estado emocional.

Durante o segundo semestre de 2008, entramos em contato com a Casa de Cultura do Butantã e os alunos de Artes Cênicas do curso de graduação da USP. Ao total, 22 alunos entre homens e mulheres interpretaram textos de sua livre escolha para cada estado emocional, totalizando 88 amostras. Cada amostra tem aproximadamente 30 segundos de duração.

A tabela com o cálculo dos 31 parâmetros para cada uma das 88 amostras pode ser consultada nos anexos (*Anexo 10*).

É importante ressaltar que, dependendo da maneira em que o locutor utilizava o microfone, afetava diretamente a intensidade sonora registrada.

O assunto “emoções” é realmente muito complexo, pois para um mesmo estado emocional que foi solicitado que os alunos interpretassem, percebemos variações de um indivíduo para outro, mostrando que cada um tem sua maneira particular de expressar sentimentos.

Esta experiência foi de grande valor, pois pudemos conhecer ambientes totalmente diferentes dos quais estamos habituados, e também foi uma excelente oportunidade de divulgar a Engenharia Mecatrônica e suas inúmeras aplicações.

4.3.4. Algoritmo analisa_população

4.3.4.1. Objetivo da Função

O algoritmo **analisa_populacao** foi elaborado para analisar os parâmetros \underline{P} de uma população (conjunto de amostras \underline{Md}) e estabelecer uma base de comparação.

É necessário discretizar os parâmetros $\underline{P(i)}$ em níveis (faixas de valores) para podermos associar a cada faixa de valor e estado emocional uma probabilidade.

Temos duas etapas para esta função:

- Obtenção dos níveis para cada parâmetro
- Distribuição dos parâmetros nos níveis por estado emocional

4.3.4.2. Obtenção dos Níveis

Optamos em classificar os parâmetros em cinco níveis diferentes. Nesta obtenção dos níveis não é levado em consideração o estado emocional da amostra, a análise é feita para toda população.

Discretização e classificação dos parâmetros:

Nível I: De 0% à 20% da população

Nível II: De 20% à 40% da população

Nível III: De 40% à 60% da população

Nível IV: De 60% à 80% da população

Nível V: De 80% à 100% da população

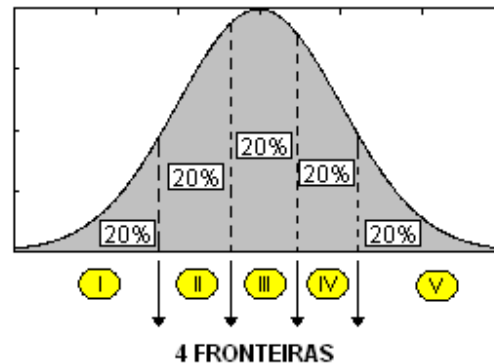


Figura 13 – Cálculo dos Níveis

Note que para separarmos os parâmetros em cinco níveis precisamos apenas de quatro valores.

O algoritmo ao receber uma população Md , retorna uma matriz $Nivel(4 \times 31)$, onde são armazenadas estas quatro fronteiras para cada um dos trinta e um parâmetros.

4.3.4.3. Distribuição dos parâmetros nos estados emocionais

Uma vez definido um domínio discreto para os parâmetros, é necessário verificar por estado emocional a sua distribuição. Em outras palavras, o programa verifica quantas vezes um nível ocorre para um determinado parâmetro para uma determinada variável. O estado emocional é conhecido pois está armazenado na coluna 32 da matriz Md , portanto sabemos qual o estado emocional de cada amostra.

Nesta etapa o algoritmo retorna uma matriz $Dist(4 \times 155)$. As linhas representam os quatro estados emocionais, neutro, alegre, triste e bravo, respectivamente nas linhas 1, 2, 3 e 4. As colunas de 1 à 5 representam a distribuição dos parâmetro $P(1)$ nos cinco níveis. De 6 à 10 representam a distribuição de $P(2)$. Segue-se assim sucessivamente até no final onde as colunas de 151 à 155 representam a distribuição de $P(31)$.

Estado Emocional	P(1) - F0 Mean					P(2) - F0 Median					...	P(31) - Esforço Vocal				
	1	2	3	4	5	6	7	8	9	10	...	151	152	153	154	155
Neutro	7	10	3	2	0	7	6	6	3	0	...	3	7	6	5	1
Alegre	2	1	6	4	9	2	4	3	6	7	...	9	4	1	3	5
Triste	6	3	6	6	1	6	3	7	3	3	...	3	3	5	5	6
Bravo	3	3	3	5	8	4	3	2	5	8	...	3	3	6	4	6

Figura 14 – Exemplo de distribuição dos parâmetros nos níveis.

Uma simples verificação do correto funcionamento do algoritmo é fazer a soma de cada linha. O valor obtido deve ser igual para todas e igual também ao número de amostras analisadas multiplicado por trinta e um (número de parâmetros).

4.3.4.4. Fluxograma resumido da função

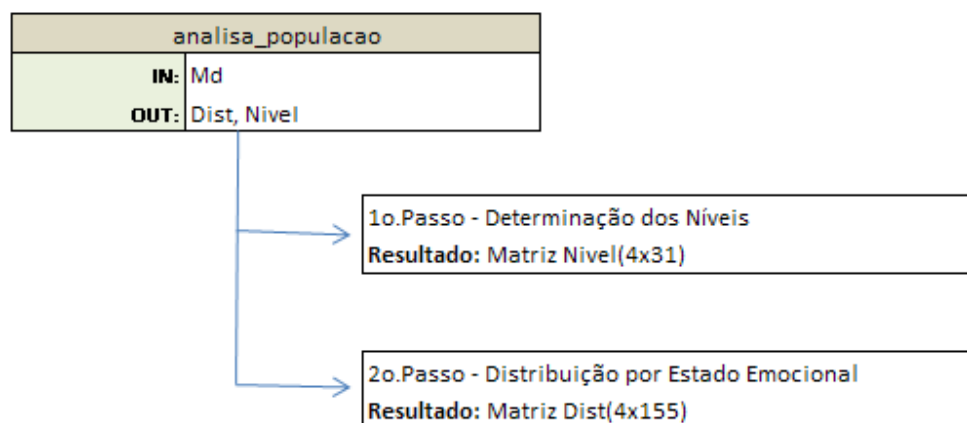


Figura 15 – Fluxograma: analisa_populacao.

O código pode ser visto em anexo (*Anexo 11*).

4.4. Análise de um indivíduo

4.4.1. Princípio de Funcionamento

Este algoritmo recebe como entrada os parâmetros P, os níveis e distribuições da população previamente analisada (Nivel, Dist) e o peso atribuído a cada variável (Pesos). Inicialmente manteremos peso unitário para todos os trinta e um parâmetros.

A saída desta função é uma matriz (emoção) que contém a “pontuação” de cada estado emocional e a probabilidade de cada estado emocional. O algoritmo também retorna um número resposta que indica o estado emocional de maior probabilidade. Todas essas saídas são mostradas ao usuário no display do programa.

O princípio de funcionamento é bastante simples. Cada parâmetro $P(i)$ é analisado e classificado em um dos níveis conforme a matriz Nível. Depois de classificado, o computador busca na matriz de distribuição Dist, qual a frequência que o parâmetro ocorre no nível classificado. O algoritmo faz isso individualmente para cada parâmetro e cada estado emocional. Os pesos são considerados nesta etapa. No término o algoritmo indica o estado emocional que obteve o maior número de ocorrências (considerando os pesos) para os parâmetros da amostra.

4.4.2. Exemplo

Na amostra em questão o parâmetro $P(1)$ obtido foi igual 200. O programa então analisa as fronteiras da matriz Nível abaixo:

Parâmetro	Fronteiras			
	1	2	3	4
P(1)	188	230,7	263,2	323,8
P(2)	168,9	209,6	238,6	275,8
...
P(31)	1	1,008	1,018	1,032

Figura 16 – Matriz Nível

Portanto, o parâmetro $P(1)$ seria classificado como **Nível II** – pois está entre a primeira e segunda fronteira.

Em sequência, o algoritmo irá buscar para o parâmetro $P(1)$, no Nível II, quais as ocorrências para cada estado emocional na matriz de distribuição Dist abaixo:

Estado Emocional	P(1) - F0 Mean				
	1	2	3	4	5
Neutro	7	10	3	2	0
Alegre	2	1	6	4	9
Triste	6	3	6	6	1
Bravo	3	3	3	5	8

Figura 17 – Matriz Distribuição

Analisando a coluna do Nível II, teríamos as seguintes pontuações para cada estado emocional, considerando o peso do parâmetro:

$$N = Pesos(1) \cdot Dist(1,2) = 1 \cdot 10 = 10$$

$$A = Pesos(1) \cdot Dist(2,2) = 1 \cdot 1 = 1$$

$$T = Pesos(1) \cdot Dist(3,2) = 1 \cdot 3 = 3$$

$$B = Pesos(1) \cdot Dist(4,2) = 1 \cdot 3 = 3$$

O mesmo procedimento é feito para todos os parâmetros. No final, o estado emocional que obtiver maior pontuação é indicado como o de maior probabilidade. A probabilidade de cada parâmetro é calculada dividindo a pontuação do estado emocional pela soma das pontuações.

4.4.2. Fluxograma Resumido da Função

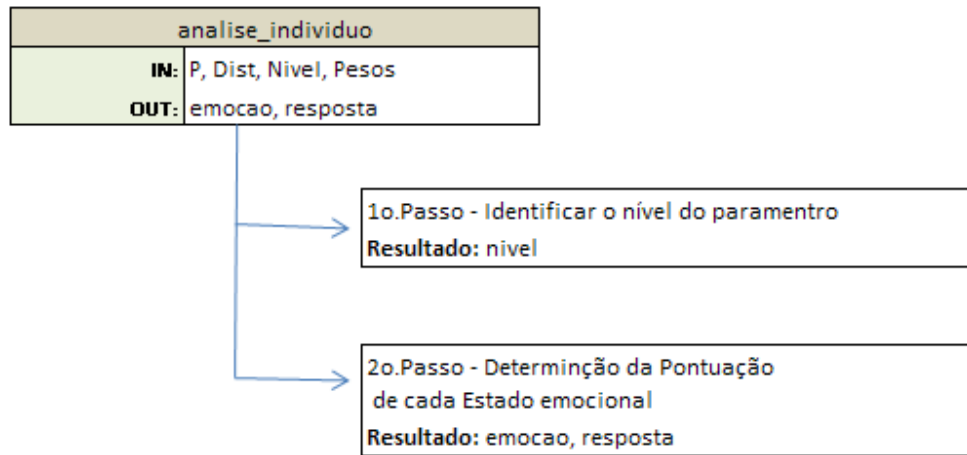


Figura 18 – Fluxograma: análise_individuo.

O código pode ser visto em anexo (*Anexo 12*).

4.5. Teste

4.5.1. Princípio de Funcionamento

O objetivo deste algoritmo é determinar a acurácia do aprendizado (previamente realizado). Ele recebe como entrada o conjunto de amostras *Ma* no qual o programa irá calcular o resultado emocional de maior probabilidade. Também são necessários os inputs dos níveis e distribuições (*Nivel*, *Dist*) no qual aprendizado foi realizado e também o peso de cada variável (*Pesos*). É importante notar que podemos aprender com um determinado conjunto de amostras e testar com um conjunto distinto de amostras, inclusive de tamanhos diferentes.

A saída é a matriz *confusao* que compara o resultado obtido com o resultado desejado, determinando a eficiência do aprendizado para um conjunto de amostras.

A desvinculação entre o algoritmo de aprendizado e algoritmo de teste proporcionou uma grande agilidade no processamento dos dados. Desta maneira conseguimos

selecionar com facilidade qual parte do corpus queríamos “aprender” e com qual parte “testar”.

Primeiramente, o computador identifica qual o estado emocional real da amostra (resultado desejado). Essa informação está armazenada na coluna 32 da matriz Md.

Na sequência, o algoritmo chama a rotina **analise_individuo** e calcula o estado emocional de maior probabilidade (resultado obtido). A comparação é armazenada na matriz confusao. O procedimento é repetido para todas as amostras do conjunto.

4.5.2. Fluxograma resumido da função

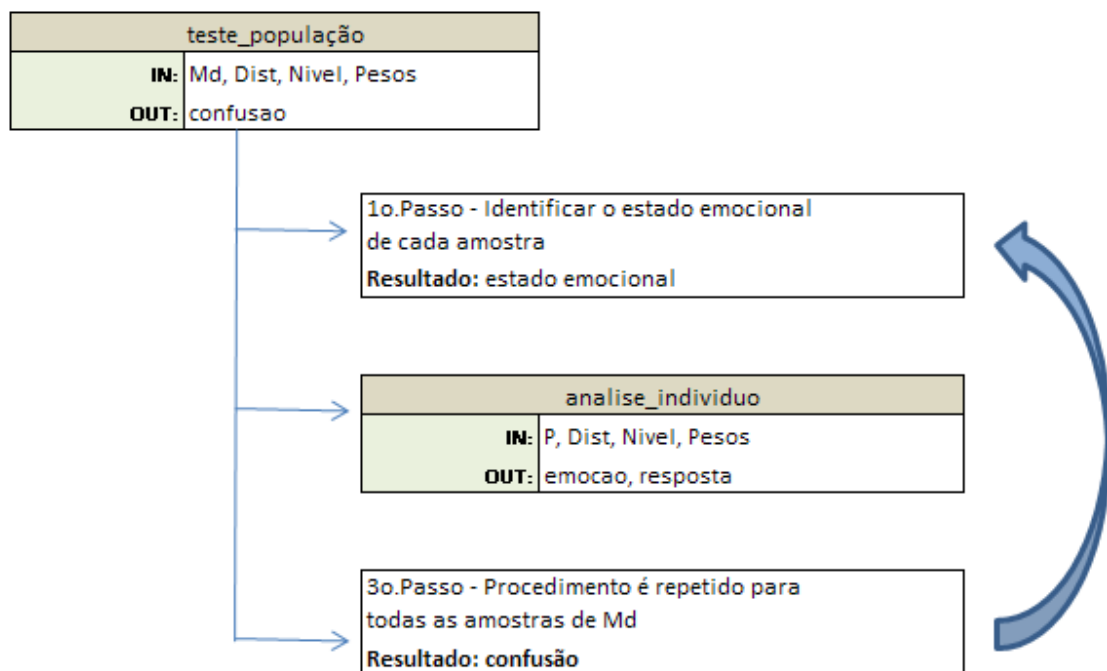


Figura 19 – Fluxograma: teste_populacao.

O código pode ser visto em anexo (Anexo 13).

5. Resultados e Discussão

5.1. Aprendizado e Teste

Em problemas em que usam técnicas de Redes Bayesianas, é muito comum utilizar uma parte do corpus para que o computador realize o aprendizado e outra para o teste.

Foram feitos diversos experimentos para aprendizado e teste, alternando as amostras do corpus. Vale a pena ressaltar que todos os testes foram feitos considerando peso unitário para todos os parâmetros.

Os experimentos foram realizados para três casos distintos:

Caso I

Aprendizado: corpus inteiro

Teste: corpus inteiro

Caso II

Aprendizado: 50% do corpus

Teste: 50% complementar do corpus

Caso III

Aprendizado: 75% do corpus

Teste: 25% complementar do corpus

5.1.1. Caso I

Neste caso, o corpus inteiro foi utilizado para observação, para que o computador realiza-se o aprendizado. Ou seja, todas as amostras foram submetidas à função **analisa_populacao** para obtermos a distribuição e nível correspondente.

Posteriormente, com as informações obtidas no aprendizado, submetemos toda as amostras para o teste. Obtivemos o seguinte resultado (*Figura 20*):



Figura20 – Resultado para o Caso I.

Analisando a matriz de confusão acima, podemos concluir que quando o locutor encontra-se no estado emocional neutro ou bravo, a porcentagem de acerto do programa é bastante elevada – 90,9% e 77,3% respectivamente.

Quando o locutor está no estado alegre, o algoritmo o confunde muitas vezes com o estado bravo. É importante notar que a recíproca não é verdadeira. Uma das causas é o fato de que em ambos os estados emocionais, o locutor tem intensas variações de frequências e intensidades. São estados de muita variação na voz.

O resultado global é bastante satisfatório, comprovando que apesar da lógica implementada ser bastante simples, o algoritmo é uma ferramenta poderosa e eficiente.

5.1.2. Caso II

A única diferença entre este caso e o primeiro, é que o corpus foi segmentado em duas partes. Primeiramente, uma parte foi utilizada para aprendizado e outra para teste. Na sequência, alternou-se as partes e executou-se novamente o aprendizado e teste. O resultado obtido pode ser visto na *Figura 21* a seguir:

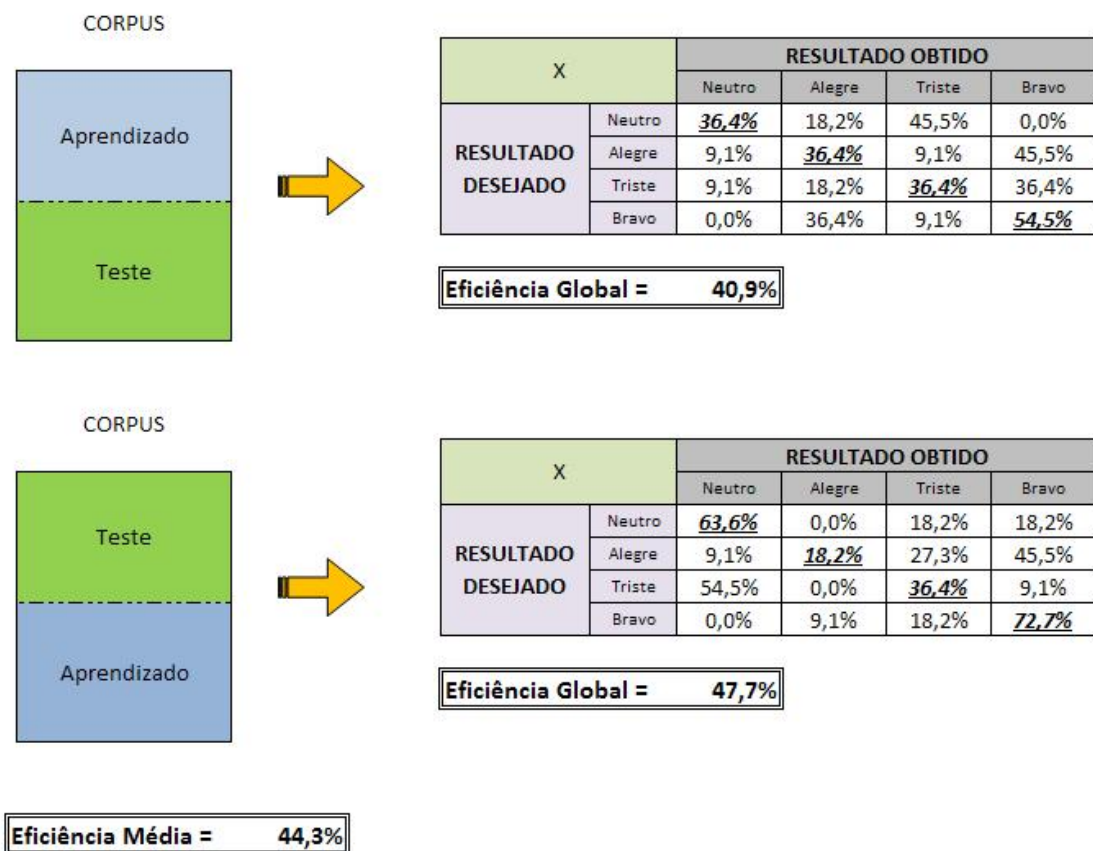


Figura 21 – Resultado para o Caso II.

Dividindo o aprendizado desta maneira, notamos uma sensível queda na eficiência (de 62,5% para 44,3%), porém, de maneira geral, as maiores eficiências se mantiveram para os estados emocionais neutro e bravo.

A queda na eficiência pode ser atribuída ao fato do corpus não ser muito extenso, tornando as sub-amostras utilizadas para aprendizado e teste insuficientes.

5.1.3. Caso III

Neste caso, utilizamos 75% do corpus para aprendizado e 25% para teste. Esta situação foi repetida quatro vezes, alternando as partes de aprendizado e teste. O resultado obtido pode ser verificado na *Figura 22* adiante:

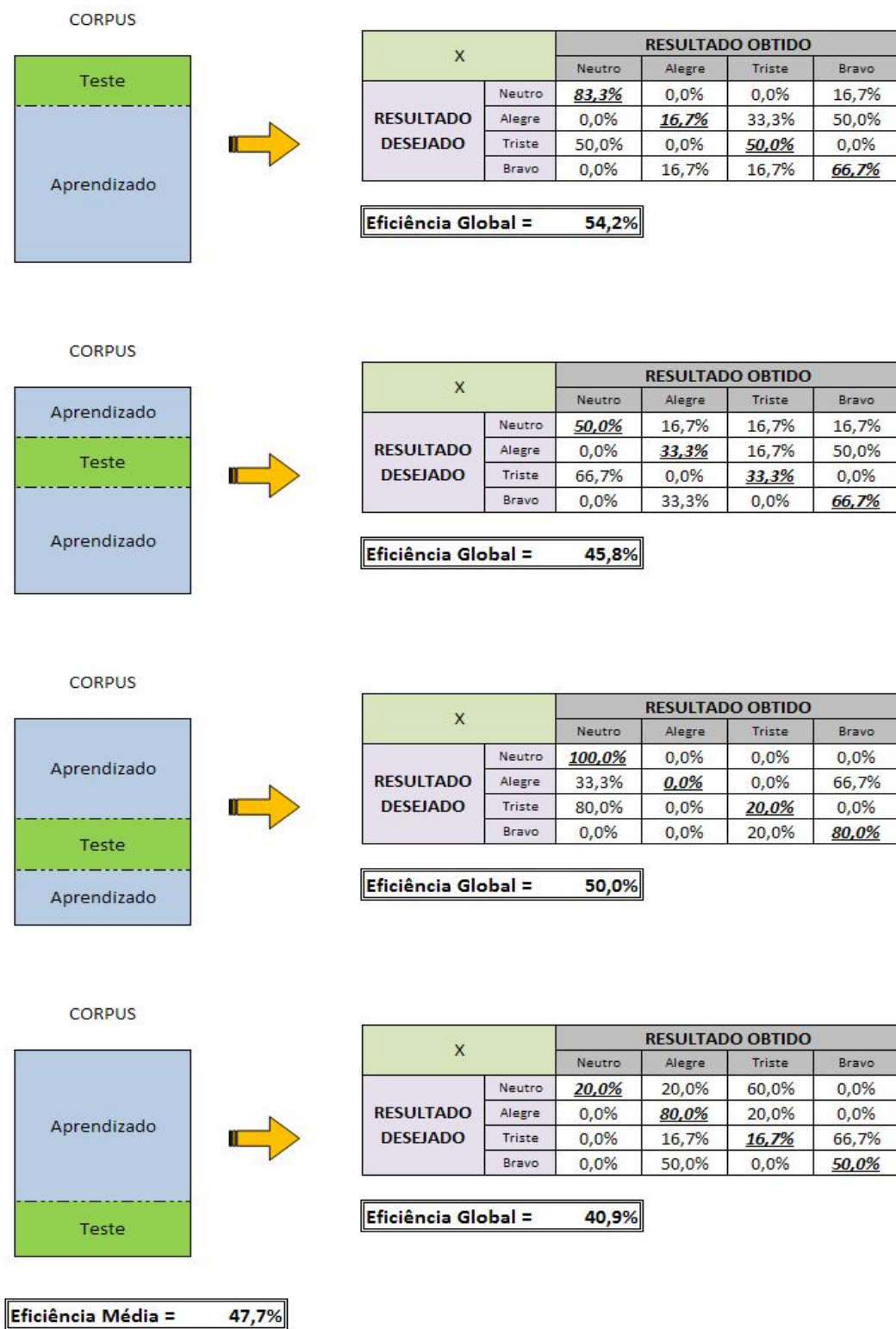


Figura 22 – Resultado obtido para o Caso III.

Dividindo o aprendizado desta maneira, notamos uma sensível queda na eficiência (de 62,5% para 47,7%), porém, de maneira geral, as maiores eficiências se mantiveram para os estados emocionais neutro e bravo.

A queda na eficiência pode ser atribuída ao fato do corpus não ser muito extenso, tornando as sub-amostras utilizadas para aprendizado e teste insuficientes.

5.2. Aplicação de Pesos aos Parâmetros

Com o objetivo de melhorar a eficiência do algoritmo, foi desenvolvido um modelo para o cálculo dos pesos dos parâmetros. Assim, os parâmetros que tenham maior relevância na identificação do estado emocional tornam-se mais decisivos.

A linha de raciocínio adotada foi: atribuir um peso maior para os parâmetros que apresentam as médias mais distintas possíveis quando classificadas por estado emocional. Traduzindo para uma linguagem matemática temos:

$$Pesos(i) = \frac{DesvioPadrão(P(i)_{N-média}, P(i)_{A-média}, P(i)_{T-média}, P(i)_{B-média})}{P(i)_{média}}$$

Após aplicar esses pesos, refizemos o aprendizado utilizando todo o corpus e testamos em todo o corpus, similarmente ao Caso I. O resultado obtido pode ser visto na *Figura 23* abaixo:

X		RESULTADO OBTIDO			
		Neutro	Alegre	Triste	Bravo
RESULTADO DESEJADO	Neutro	<u>86,4%</u>	4,5%	4,5%	4,5%
	Alegre	9,1%	<u>54,5%</u>	0,0%	36,4%
	Triste	31,8%	4,5%	<u>50,0%</u>	13,6%
	Bravo	0,0%	4,5%	9,1%	<u>86,4%</u>

Eficiência Global =	69,3%
----------------------------	--------------

Figura 23 – Resultado após aplicação de pesos.

Após a aplicação dos pesos, a eficiência subiu consideravelmente de 62,5% para 69,3%. Este resultado comprova que o raciocínio utilizado para atribuição dos pesos, apesar de ser simples, é bastante eficiente.

A matriz de nível, distribuição e os pesos para o algoritmo final podem ser vistos nos *Anexos, 14, 15 e 16* respectivamente.

Considerando o sistema de aprendizado e teste do Caso I, com a aplicação de pesos aos parâmetros, obtivemos uma eficiência de aproximadamente 70%.

O algoritmo identificou com grande eficiência o estado emocional neutro. Ao identificar o estado emocional triste, algumas vezes o computador o confundiu com o neutro. É importante notar que a recíproca não é verdadeira neste caso, ou seja o triste se confunde com o neutro mas neutro não se confunde com o triste. Uma das causas é que para ambos os estados, os níveis de intensidade de voz e variação de frequência são menores. Uma maneira de melhorar este ponto seria uma investigação sobre os parâmetros mais significativos na distinção destes dois estados.

Para o estado emocional bravo o algoritmo obteve bastante eficiência. Ao identificar o estado emocional alegre, algumas vezes o computador o confundiu com o bravo. É importante notar que a recíproca não é verdadeira neste caso, ou seja, o alegre se confunde com o bravo, mas o bravo não se confunde com o alegre.

Uma das causas é que para ambos os estados, os níveis de intensidade de voz e variação de frequência são maiores. Ambos são estados de bastante fluência e vivacidade na voz. Uma maneira de melhorar este ponto seria uma investigação sobre os parâmetros mais significativos na distinção destes dois estados.

Apenas para estabelecer uma base de comparação, um estudo similar realizado por finlandeses na *University of Oulu* [1], obteve uma eficiência média de 60%. Um fato curioso é que o estado emocional em que o estudo referido obteve maior dificuldade em reconhecer foi o bravo, enquanto que o triste foi o de maior eficiência, contrastando com o nosso estudo. Podemos concluir também que os aspectos fonéticos e culturais de cada idioma e região, influencia diretamente na maneira em que expressamos nossas emoções através da voz.

6. Conclusão

Este resultado é bastante satisfatório, comprovando que apesar da lógica implementada ser bastante simples, o algoritmo obteve alta eficiência na resolução de um problema complexo. É importante ressaltar que a utilização de redes bayesianas mostrou-se uma ferramenta poderosa ao reconhecer tendências nos parâmetros medidos para cada estado emocional. Levando também em consideração de que os parâmetros acústicos variam muito não apenas de acordo com estado emocional mas também com as características da voz de cada indivíduo, afetando diretamente os cálculos das variáveis.

7. Considerações Finais

Segue abaixo alguns pontos em que o trabalho pode ser melhorado em futuros estudos sobre o tema:

- a) Correlação entre variáveis: Pode-se realizar uma análise da correlação entre todos os trinta e um parâmetros, com o objetivo de reduzir o número de variáveis, eliminando assim os parâmetros redundantes.
- b) Análise focada: Para melhorar a eficiência do algoritmo no reconhecimento dos estados alegre e triste, pode-se fazer um estudo focado para obter os parâmetros de maior relevância na distinção destes estados emocionais.
- c) Aumento do corpus: Com o aumento da dimensão do banco de dados, é possível realizar experimentos mais variados e obter resultados mais consistentes (os resultados do Caso II e Caso III não foram satisfatórios), possibilitando o aprendizado e teste em diferentes configurações.

Anexo 1 – Função: segmentar

```
function S = segmentar(data, fs)

c = 800;          % Numero de colunas de S = frequencia de amostragem (8kHz) x
                  periodo de discretizacao (0,02s)
l = ceil(length(data)/(fs*0.1)); % Calcula o numero de linhas necessarios, a
funcao ceil arredonda para cima
S = zeros(l,c);
for i=1:(l-1);    % ate (l-1) para que na ultima linha o parametro nao exceda
                  a dimensao de data
    S(i,1:c) = data((i*800-799):(i*800));
end
final = length(data) - (l-1)*c; %Calcula o numero de elementos a serem
colocados na ultima linha de S
S(l,1:final) = data(((l-1)*c+1):length(data)); %Realiza a segmentacao da
ultima linha de S

end
```

Anexo 2 – Função: descarta

```
function [Slinha, datalinha] = descarta(S, data);

[1, c] = size(S);
aux = 0;
i = 0;

rmss = intensidade(S); % Calcula a intensidade para poder discernir o
silencio do som
flagg = silencio(rmss); % Atribui 0 para o silencio e 1 para o som - critério
> 0.05

while (aux == 0) % Retorna o indice i do flag onde começa a amostra
    i = i + 1;
    if (flagg(i) == 1)
        aux = 1;
    end
end

aux = 0;
j = 1 + 1;

while (aux == 0) % Retorna o indice j do flag onde termina a amostra
    j = j - 1;
    if (flagg(j) == 1)
        aux = 1;
    end
end

Slinha = S(i:j,:); % Retorna a nova matriz S desconsiderando os periodos de
silêncio iniciais e finais
datalinha = data((i-1)*800+1:j*800); % idem acima para o vetor data

end
```

Anexo 3 – Função: frequencia

```
function F0 = frequencia(data)

dataa = zeros(1,length(data)+ 8000);    % A frequencia é calculada para cada
0.1s, mas utiliza um período de observação
dataa(1,1:length(data)) = data';    % de 1s. É adicionado 1s no final da
amostra para que não haja problema com o indice do vetor  dataa

l = length(data)/800;
F0 = zeros(1,l);

for i=1:l
    sample = dataa((i-1)*800+1:(i-1)*800+8000);
    Y = fft(sample,8192);    % Calculo a Transformada de Fourier
    Pyy = Y.*conj(Y);    % Calcula a magnitude da Transformada
    [ps, indice] = max(Pyy(1:4095)); % Encontra o indice de maior magnitude
    F0(i) = 8000*indice/8192; % Atraves do indice, pode-se encontra a
    frequencia fundamental F0 para cada TD
end

end
```

Anexo 4 – Função: intensidade

```
function RMS = intensidade(S)

[l,c] = size(S); % Obtem as dimensoes da matriz S
for i=1:l
    RMS(i) = (sumsqr(S(i,1:c))/c)^.5; % Calcula a raiz da media dos quadrados
    - RMS
end

end
```

Anexo 5 – Função: silencio

```
function flag = silencio(RMS);

l = length(RMS);
flag = zeros(1,l);           % Cria o vetor flag contendo zeros
for i=1:l
    if (RMS(i) >= 0.05)
        flag(i) = 1;        % Atribui o valor um para RMS >= 0.05
    end
end

end
```

Anexo 6 – Função: transicao

```
function transicao = continuidade(flag);

l = length(flag);
transicao = zeros(1,l);
if (flag(1) == 1) % Faz a verificacao se a amostra inicia com
    silencio ou nao
        transicao(1)=1;
end
for i=2:l
    if ( flag(i-1)==0 && flag(i)==1 ) % Verifica se houve a transicao
        silencio-voz
        transicao(i) = 1;
    end
    if ( flag(i-1)==1 && flag(i)==0 ) % Verifica se houve a transicao voz-
        silencio
        transicao(i) = -1;
    end
end
end
```


Anexo 7 – Função: normalizar

```
function F0normal = normalizar(F0)

lowbond = prctile(F0,5);    % Estes limites são adotados com base nos percentil
de 5% e 95%
highbond = prctile(F0,95); % com o objetivo de descartar os 5% inferiores e
superiores da amostra
l = length(F0);            % tornando os dados mais homogêneos
aux = 1;

for i=1:l
    if (F0(i) >= lowbond && F0(i) <= highbond)
        F0normal(aux) = (F0(i) - lowbond)/(highbond - lowbond);    % A
        normalização
        aux = aux + 1;
    end
end
```

Anexo 8 – Função: parametros

```
function P = parametros(F0, RMS, flag, transicao)

l = length(F0);
P = zeros(1,31);

% A) Parametros relacionados a frequencia fundamental

F0aux = F0.*flag; % Produto de cada valor entre os vetores com objetivo de
zerar as partes silenciosas

j = 1;
for i=1:l
    if (F0aux(i) > 0)           % Cria o vetor de frequencia fundamental
        f0(j) = F0aux(i);      % considerando apenas as partes de voz ativa
        j = j + 1;
    end
end

P(1) = mean(f0); % Retorna a media de f0
P(2) = median(f0); % Retorna a mediana de f0
P(3) = max(F0); % Retorna o máximo de F0
P(4) = min(F0); % Retorna o mínimo de F0
P(5) = prctile(f0,95); % Retorna o percentil de f0 para 95%
P(6) = prctile(f0,5); % Retorna o percentil de f0 para 5%
P(7) = P(3) - P(4); % Retorna a amplitude de F0
P(8) = P(5)-P(6); % Retorna a amplitude entre 5%-95% dos percentil
P(9) = std(f0); % Retorna o desvio padrao de f0

% B) Parametros relacionados a intensidade

RMSaux = RMS.*flag; % Produto de cada valor entre os vetores com objetivo de
zerar as partes silenciosas

j = 1;
for i=1:l
    if (RMSaux(i) > 0)           % Cria o vetor de intensidade
        rms(j) = RMSaux(i);      % considerando apenas as partes de voz ativa
        j = j + 1;
    end
end

P(10) = mean(rms); % Retorna a media de rms
P(11) = median(rms); % Retorna a mediana de rms
P(12) = max(rms); % Retorna o maximo de rms
P(13) = prctile(rms,95); % Retorna o perctile de 95% de rms
P(14) = prctile(rms,5); % Retorna o perctile de 5% de rms
P(15) = P(13) - P(14); % Retorna a amplitude entre os percentile 5%-95% de rms
P(16) = std(rms); % Retorna o desvio padrao de rms

% C) Parametros relativos a duracao do som
```

```

j=1;
fim=1; % Inicializa o silencio no primeiro segmento
for i=1:l
    if (transicao(i) == 1)
        inicio = i;
        quieto(j) = inicio - fim; % Vetor que armazena a duracao do silencio
    end
    if (transicao(i) == -1)
        fim = i-1;
        barulho(j) = fim - inicio; % Vetor que armazena a duracao dos
segmentos continuos de voz
        j = j + 1;
    end
end

P(17) = mean(barulho); % Duração média de um segmento contínuo de voz
P(18) = mean(quieto); % Duração média das pausas
P(19) = std(quieto); % Desvio padrão das pausas
P(20) = sum(barulho)/sum(quieto); % Razão voz/silencio

% D) Parametros relacionados a dinamica da frequencia fundamental

j=1;
for i=1:l
    if (transicao(i) == 1)
        F0inicio = F0(i);
    end
    if (transicao(i) == -1)
        F0final = F0(i-1);
        F0variacao(j) = F0inicio - F0final; % Armazena a variacao da
frequencia fundamental
        j = j + 1; % em um segmento contínuo de voz
    end
end

j = 1;
k = 1;
for i=1:length(F0variacao)
    if (F0variacao(i) >= 0)
        F0UP(j) = F0variacao(i);
        j = j +1;
    end
    if (F0variacao(i) < 0)
        F0DOWN(k) = F0variacao(i);
        k = k +1;
    end
end

if (j == 1)
    F0UP = 0; % No caso de não haver incrementos em F0
end
if (k == 1)
    F0DOWN = 0; % No caso de não haver decrementos em F0
end

```

```

P(21)=mean(F0UP); % Incremento médio de um segmento contínuo de voz
P(22)=max(F0UP); % Incremento máximo de um segmento contínuo de voz
P(23)=std(F0UP); % Desvio padrão do incremento
P(24)=mean(F0DOWN); % Decremento médio de um segmento contínuo de voz
P(25)=min(F0DOWN); % Decremento máximo de um segmento contínuo de voz
(INVERTE-SE AS FUNÇÕES MIN MAX POR CONTA DO SINAL)
P(26)=std(F0DOWN); % Desvio padrão do decremento
P(27)=length(F0UP)/length(F0DOWN); % Razão entre incremento e decremento

% E) Parametros relacionados a frequenncia fundamental após normailzação

f0n = normalizar(f0); % Aplica a normalização no vetor das frequencias

P(28) = mean(f0n); % Retorna a media do vetor normalizado
P(29) = median(f0n); % Retorna a mediana do vetor normalizado
P(30) = std(f0n); % Retorna o desvio-padro do vetor normalizado

% F) Parametro de esforço vocal - Considera a frequencia e intensidade no
mesmo instante

P(31) = mean(f0.*rms)/(mean(f0)*mean(rms));

end

```

Anexo 9 – Função: calcula_parametros

```
function P = calcula_parametros(data, fs)

S = segmentar(data, fs); % Funcao que segmenta o vetor data em uma matriz,
                        % onde os parametros de intensidade sao avaliados
para cada linha da matriz

[S, data] = descarta(S, data);
% Descarta os periodos iniciais e finais de silencio de cada amostra e retorna
o novo vetor data e matriz S

F0 = frequencia(data); % Acha a frequencia fundamental utilizando a
transformada de fourier - Período de observação de 2s

RMS = intensidade(S); % Calcula a intensidade da amostra para cada TD (0,1seg)

flag = silencio(RMS); % Identifica os segmentos silenciosos - RMS < 0,05

transicao = continuidade(flag); % Identifica os segmentos continuos de voz

P = parametros(F0, RMS, flag, transicao); % Calcula todos os parametros
acusticos definidos para a amostra

clear F0 RMS S data flag fs transicao; % Limpa as variaveis desnecessarias

end % Fim do algoritmo
```

Anexo10 – Tabela do corpus

		1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	17	18	19	2	21	22	23	24	2	26	2	2	2	3	3		
		25	20	73	13	51	14	59	36	13	0	0	0	0	0	0	0	0	2,9	2,8	1,5	0	59	50	12	5	-95	##	13	0	2	0	0	0	1,0
Allan	Alegre	7	0	40	0	12	26	13	28	12	0	0	0	0	0	0	0	0	3,4	3,4	2,0	0	24	20	8	49	-17	92	31	3	0	0	0	1,0	
Allan	Bravo	17	7	16	0	4	0	7	1	3	57	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1,0	
Allan	Neutr o	12	12	23	11	14	11	12	8	22	15	0	0	0	0	0	0	0	0	0	0	0	0	2	13	4	-23	##	46	4	0	0	0	0	1,0
Allan	Triste	18	18	41	13	23	14	28	4	87	42	0	0	0	0	0	0	0	4,8	3,8	2,3	2	15	69	18	-21	44	20	2	0	0	0	0	1,0	
Amanda	Alegre	40	34	95	2	70	22	94	48	16	0	0	0	0	0	0	0	0	1,4	4,8	4,2	3	55	45	12	6	0	-35	##	65	7	0	0	0	1,0
Amanda	Bravo	37	30	99	2	71	23	98	47	16	0	0	0	0	0	0	0	0	1,8	3,6	2,4	5	29	35	7	79	##	##	21	7	4	0	0	0	1,0
Amanda	Neutr o	23	23	67	7	25	20	67	4	58	29	0	0	0	0	0	0	0	1,1	5,8	5,6	2	4	47	10	-11	24	11	9	0	0	0	0	1,0	
Amanda	Triste	30	22	66	4	1	4	6	3	8	8	15	1	1	0	0	0	0	1,3	##	8,6	1	32	45	10	2	-5	-9	4	7	0	0	0	1,0	
Anna	Alegre	34	30	4	##	18	83	19	94	63	19	1	1	2	2	1	1	0	2,1	3,3	1,8	6	49	86	17	9	-85	##	22	2	3	0	0	0	1,0
Anna	Bravo	30	29	82	0	17	37	21	64	16	5	87	0	0	0	0	0	0	4,5	3,3	1,7	3	7	50	14	##	##	20	4	3	0	0	0	1,0	
Anna	Neutr o	22	21	64	15	28	16	49	11	0	58	1	1	1	1	1	1	0	1,5	3,0	2,1	5	6	81	15	-95	##	15	2	6	0	0	0	1,0	
Anna	Triste	28	29	70	19	33	20	50	12	9	63	0	0	0	0	0	0	0	2,4	2,9	1,6	8	7	70	15	-20	69	23	2	0	0	0	0	1,0	
Claudianne01	Alegre	42	45	72	22	61	24	50	37	13	0	0	0	0	0	0	0	0	3,1	2,6	1,4	1	58	35	11	6	-37	##	47	2	0	0	0	1,0	
Claudianne01	Bravo	30	26	5	##	18	51	20	94	30	12	0	0	0	0	0	0	0	3,9	2,7	1,4	4	28	14	45	-55	##	55	2	0	0	0	0	1,0	
Claudianne01	Neutr o	21	21	25	18	25	18	66	65	19	0	0	0	0	0	0	0	0	1,3	2,6	1,3	5	2	12	3	-3	-8	2	2	0	0	0	0	1,0	
Claudianne01	Triste	23	22	48	17	36	17	30	18	3	45	0	0	0	0	0	0	0	0,9	3,2	2,3	3	8	10	5	21	-24	31	10	16	0	0	0	1,0	
Claudianne02	Alegre	24	22	38	15	36	17	22	19	60	0	0	0	0	0	0	0	0	3,0	3,4	2,3	8	30	11	37	-34	##	45	2	0	0	0	0	1,0	
Claudianne02	Bravo	35	34	3	56	16	53	20	39	32	10	6	0	0	0	0	0	0	5,5	3,1	2,6	1	79	28	5	11	11	1	##	10	7	2	0	0	1,0
Claudianne02	Neutr o	19	20	26	15	21	17	11	0	12	0	0	0	0	0	0	0	0	1,3	4,1	3,7	3	3	29	7	-1	-1	0	17	0	0	0	0	1,0	
Claudianne02	Triste	20	19	35	4	25	17	35	1	83	35	0	0	0	0	0	0	0	0,7	5,2	5,4	1	7	30	11	-11	15	5	6	0	0	0	0	1,0	
Diego01	Alegre	26	23	65	12	51	14	52	36	8	97	0	0	0	0	0	0	0	1,8	4,4	4,6	4	19	22	5	44	-21	35	15	6	0	0	0	1,0	
Diego01	Bravo	47	53	90	4	73	19	90	53	21	0	0	0	0	0	0	0	0	4,5	6,9	5,3	6	12	45	18	0	##	##	86	3	0	0	0	1,0	
Diego01	Neutr o	22	18	61	5	61	11	51	50	14	0	0	0	0	0	0	0	0	0,9	6,5	6,4	1	79	51	19	7	-30	59	25	2	0	0	0	1,0	
Diego01	Triste	22	16	76	0	51	13	75	38	13	0	0	0	0	0	0	0	0	3,5	8,3	7,1	4	13	42	16	-27	93	39	3	2	0	0	0	1,0	
Diego02	Alegre	28	27	57	19	52	20	37	32	5	87	0	0	0	0	0	0	0	4,0	3,8	2,8	0	46	23	6	86	-19	32	13	2	0	0	0	1,0	
Diego02	Bravo	22	22	45	13	29	14	32	15	2	57	0	0	0	0	0	0	0	3,9	7,3	6,6	5	52	27	6	96	-70	##	78	3	0	0	0	1,0	
Diego02	Neutr o	24	23	47	7	41	13	47	28	4	92	0	0	0	0	0	0	0	1,3	5,3	4,9	2	40	29	1	78	-57	##	77	4	0	0	0	1,0	
Diego02	Triste	17	16	39	2	31	12	39	18	5	56	0	0	0	0	0	0	0	2,6	9,4	8,3	3	18	12	8	32	##	##	13	2	8	0	0	0	1,0
Felipe	Alegre	23	18	41	13	41	13	28	27	10	6	0	0	0	0	0	0	0	5,9	2,7	1,8	0	87	23	0	##	##	79	1	0	0	0	0	1,0	
Felipe	Bravo	20	17	50	3	34	12	50	21	9	75	0	0	0	0	0	0	0	9,3	2	##	7	57	27	3	96	##	##	0	7	0	0	0	1,0	
Felipe	Neutr o	17	14	45	6	37	11	45	26	4	78	0	0	0	0	0	0	0	2,5	5,5	4,9	4	36	26	4	73	##	##	80	6	0	0	0	1,0	
Felipe	Triste	13	13	26	0	2	7	1	8	5	26	0	0	0	0	0	0	0	3,0	7,6	5,4	3	5	25	10	##	##	0	6	0	0	0	0	1,0	
Janaina02	Alegre	61	7	55	1	##	5	##	27	8	##	82	27	0	0	0	0	0	2,1	4,6	5,2	4	62	40	11	6	##	##	20	1	5	0	0	0	1,0
Janaina02	Bravo	26	8	26	2	##	1	9	2	##	7	0	0	0	0	0	0	0	2,8	4,2	3,7	6	10	63	21	6	5	-96	##	0	3	0	0	0	1,0
Janaina02	Neutr o	21	20	44	1	3	8	7	8	1	37	0	0	0	0	0	0	0	2,4	4,6	5,1	5	14	78	24	-36	85	44	7	0	0	0	0	1,0	
Janaina02	Triste	25	22	80	3	60	16	79	43	13	0	0	0	0	0	0	0	0	1,2	7,1	6,8	2	18	1	53	-4	11	6	8	0	0	0	0	1,0	
Janaina01	Alegre	54	58	2	##	20	82	23	##	7	0	0	0	0	0	0	0	0	1,2	6,6	6,3	2	83	4	4	0	0	0	0	27	5	6	3	1,0	
Janaina01	Bravo	76	61	4	1	30	3	3	3	0	6	1	0	0	0	0	0	0	8,3	3,5	2,1	3	52	55	13	0	1	##	##	25	4	9	0	0	1,0
Janaina01	Neutr o	22	20	83	14	39	17	68	22	2	66	0	0	0	0	0	0	0	0,5	6,9	1	1	1	15	3	-1	-1	0	28	2	1	3	0	0	1,0
Janaina01	Triste	27	26	53	0	34	21	52	13	5	59	0	0	0	0	0	0	0	0,8	8,0	8,3	1	1	18	4	-15	32	14	9	4	4	2	0	0	1,0
Letidia	Alegre	24	23	39	19	30	20	19	0	33	0	0	0	0	0	0	0	0	0,9	3,5	2,8	2	5	45	12	-43	##	74	5	0	0	0	0	1,0	
Letidia	Bravo	36	35	78	20	51	26	58	24	1	79	0	0	0	0	0	0	0	3,8	2,7	1,3	4	50	24	9	69	##	##	14	6	2	0	0	0	1,0
Letidia	Neutr o	24	25	33	17	27	20	15	0	24	0	0	0	0	0	0	0	0	0,4	7,8	8,4	1	0	0	0	0	-4	-5	1	4	0	0	0	1,0	
Letidia	Triste	29	28	43	21	43	22	22	21	4	57	0	0	0	0	0	0	0	0,6	5,5	4,3	1	9	5	35	-16	46	16	2	0	0	0	0	1,0	
Letiricia	Alegre	27	23	68	2	52	18	67	34	10	0	0	0	0	0	0	0	0	1,3	4,6	4,1	3	19	29	6	54	##	##	13						

	o	2	3	7	2	2	4	4			1	1	1	1	1	1	0											3		5	5	2		0									
Letirida	Triste	23	22	36	3	2	28	20	36	1	0	0	0	0	0	0	0	0,6	##	##	0				4	30	8	-3	-8	4	6	0	0	0	1	0							
Lucas	Alegre	23	20	72	11	43	13	60	30	11	0	0	0	0	0	0	0	2,6	3,2	2,0	8				6	57	14	##	##	19	3	2	0	0	0	1	0						
Lucas	Bravo	25	18	66	6	96	1	2	57	46	14	1	0	0	0	0	0	3,4	3,7	3,0	9				33	4	80	-94	##	67	3	2	1	3	1	1	0						
Lucas	Neutr o	13	12	25	10	18	11	14			0	0	0	0	0	0	0	1,1	6,9	5,4	1				10	7	32	-23	90	38	5	2	2	1	0	0							
Lucas	Triste	18	17	46	6	4	25	12	46	12	0	0	0	0	0	0	0	1,5	4,7	3,7	3				4	43	10	-52	##	83	3	4	4	3	0	0	1	0					
Luis	Alegre	19	16	43			35	13	43	22	0	0	0	0	0	0	0	2,4	3,8	3,4	6				31	3	68	-31	##	52	4	2	1	3	0	0	1	0					
Luis	Bravo	22	16	86	6	4	7	2	2	6	0	0	0	0	0	0	0	6,6	4,0	3,5	6				37	0	67	##	##	22	1	2	3	1	3	0	0	1	0				
Luis	Neutr o	16	13	54			26	11	53	15	0	0	0	0	0	0	0	1,4	4,6	5,8	3				11	2	29	-82	##	64	6	0	0	0	0	1	0	0	1	0			
Luis	Triste	16	14	50			25	11	50	14	0	0	0	0	0	0	0	1,7	6,1	7,7	3				14	7	38	-39	##	55	5	3	2	2	0	0	1	0	0	1	0		
Marco01	Alegre	18	16	75	12	26	12	63	13		0	0	0	0	0	0	0	3,3	3,4	2,0	9				14	57	20	-50	##	47	2	0	0	0	0	1	0	0	1	0			
Marco01	Bravo	15	15	32	11	21	11	21		98	0	0	0	0	0	0	0	2,2	4,3	3,1	5				16	7	37	-25	85	40	6	3	3	2	0	0	0	1	0				
Marco01	Neutr o	14	12	25	1	5	24	11	24	12	0	0	0	0	0	0	0	0,6	11,7	9,9	0				1	11	3	-67	##	57	4	0	0	0	0	1	0	0	1	0			
Marco01	Triste	15	13	24	1	2	22	11	23	11	0	0	0	0	0	0	0	0,8	9,8	8,8	1				8	6	24	-6	13	4	4	3	2	3	0	0	0	1	0				
Marco02	Alegre	18	17	42	12	26	13	30	12		0	0	0	0	0	0	0	3,6	3,5	1,9	0				19	11	29	-50	##	54	2	0	0	0	0	1	0	0	1	0			
Marco02	Bravo	23	20	48	12	42	14	35	28		0	0	0	0	0	0	0	4,0	3,8	3,1	0				39	6	75	-54	##	72	2	0	0	0	0	1	0	0	1	0			
Marco02	Neutr o	18	13	34			31	11	34	19	0	0	0	0	0	0	0	1,4	4,5	3,1	3				26	8	56	-47	##	77	2	0	0	0	0	1	0	0	1	0			
Marco02	Triste	18	16	53			25	14	53	11	0	0	0	0	0	0	0	4,3	5,7	4,1	7				22	13	8	43	-15	36	11	1	3	2	2	0	0	0	1	0			
Mateus	Alegre	32	26	75	16	61	17	58	44	16	0	0	0	0	0	0	0	9,1	3,3	2,2	5				54	0	95	-74	##	61	2	0	0	0	0	1	0	0	1	0			
Mateus	Bravo	32	28	68	23	58	23	45	35	11	0	0	0	0	0	0	0	8,7	2,9	1,8	7				10	34	15	0	##	##	90	2	0	0	0	0	1	0	0	1	0		
Mateus	Neutr o	23	17	52	14	40	14	37	25	10	0	0	0	0	0	0	0	0,4	3,5	2,5	1				1	10	3	##	##	10	8	5	0	0	0	0	1	0	0	1	0		
Mateus	Triste	24	22	69			48	17	69	31	0	0	0	0	0	0	0	6,5	6,0	5,0	0				54	1	87	##	##	18	5	6	2	2	2	0	0	0	1	0			
Paulo	Alegre	27	25	85	13	48	16	71	32		0	0	0	0	0	0	0	2,2	2,8	1,6	8				35	0	77	-87	##	10	1	3	3	3	2	0	0	0	1	0			
Paulo	Bravo	27	26	66	16	50	19	50	31		0	0	0	0	0	0	0	1,7	3,0	1,5	6				27	48	5	82	-64	##	94	4	0	0	0	0	1	0	0	1	0		
Paulo	Neutr o	20	21	71	12	24	15	59			0	0	0	0	0	0	0	0,4	6,9	5,7	1				3	87	15	-14	19	6	15	0	0	0	0	1	0	0	1	0			
Paulo	Triste	28	27	64	17	35	18	47	17		0	0	0	0	0	0	0	3,5	3,0	2,0	1				26	7	51	-59	##	66	3	5	5	2	0	0	0	1	0	0	1	0	
Rafaela	Alegre	35	27		20	66	22		44	21	0	0	0	0	0	0	0	2,2	3,2	2,3	7				39	47	11	11		10	5	5	0	0	0	0	1	0	0	1	0		
Rafaela	Bravo	44	31	##	23	25	##	96	26		0	0	0	0	0	0	0	6,1	3,3	1,9	7				64	35	12	2	##	##	41	6	1	2	1	2	0	0	0	1	0		
Rafaela	Neutr o	32	26	53	19	53	20	34	33	11	0	0	0	0	0	0	0	0,4	8,1	8,5	0				29	29	5	93	-1	-1	0	5	3	2	3	0	0	0	1	0			
Rafaela	Triste	27	24	66	20	43	20	45	23		0	0	0	0	0	0	0	3,2	3,8	2,9	8				28	25	7	57	-36	69	48	11	2	2	2	0	0	0	1	0			
Renato	Alegre	33	25	75	14	62	17	60	44	15	0	0	0	0	0	0	0	3,7	2,8	1,6	2				85	50	15	8	##	##	13	4	5	3	2	3	0	0	0	1	0		
Renato	Bravo	33	24	67	12	57	17	54	40	15	0	0	0	0	0	0	0	6,2	3,0	1,6	0				74	31	10	9	##	##	17	2	2	4	2	3	0	0	0	1	0		
Renato	Neutr o	27	25	67	12	48	14	54	33	12	0	0	0	0	0	0	0	1,4	3,7	2,4	4				38	30	9	82	-75	##	67	3	3	3	3	0	0	0	1	0			
Renato	Triste	32	24	69	13	48	17	56	30	13	0	0	0	0	0	0	0	3,2	3,6	2,6	8				72	26	12	5	3	##	##	17	6	1	5	2	4	1	0	0	0	1	0
Richard	Alegre	30	28	68	18	51	21	49	29		0	0	0	0	0	0	0	11,			3				77	37	12	3	-16	23	11	4	0	0	0	0	1	0	0	1	0		
Richard	Bravo	25	24	61	15	33	18	45	14		0	0	0	0	0	0	0	##	3,1	1,9	2				57	67	11	-56	89	44	1	0	0	0	0	1	0	0	1	0			
Richard	Neutr o	18	16	64	13	23	13	51			0	0	0	0	0	0	0	2,1	3,9	2,3	5				52	49	12	6	9	-7	18	8	4	0	0	0	0	1	0	0	1	0	
Richard	Triste	23	22	71	11	26	15	60	10		0	0	0	0	0	0	0	6,4	4,9	3,6	2				29	64	25	-9	27	10	1	0	0	0	0	0	1	0	0	1	0		
Tuti	Alegre	35	26	96	17	74	22	79	52	19	0	0	0	0	0	0	0	1,0	9,5	##					19	4	81	-7	24	10	5	0	0	0	0	0	1	0	0	1	0		
Tuti	Bravo	52	44		24	98	29		68	21	0	0	0	0	0	0	0	2,9	2,9	1,8	0				81	66	15	0	##	##	16	5	4	0	0	0	0	0	1	0	0	1	0
Tuti	Neutr o	22	20	62		30	18	62	11		0	0	0	0	0	0	0	1,8	4,0	3,4	4				5	92	16	-29	##	33	2	0	0	0	0	0	0	1	0	0	1	0	
Tuti	Triste	23	22	79	17	26	20	62		66	0	0	0	0	0	0	0	1,7	4,6	4,0	4				13	34	6	50	-23	55	24	12	0	0	0	0	0	0	1	0	0	1	0

Anexo 11 – Função: analisa_população

```
function [Dist, Nivel] = analisa_populacao(Md)

[1, c] = size(Md); % Obtem o tamanho da Matriz de amostras

% 1o. Passo - Obtenção dos Níveis

Nivel = zeros(31,4); % Determina o limite entre os 5 níveis para os 31
parametros para toda a amostra

for i=1:31
    Nivel(i,1) = prctile(Md(:,i),20); % 20% da amostra
    Nivel(i,2) = prctile(Md(:,i),40); % 40% da amostra
    Nivel(i,3) = prctile(Md(:,i),60); % 60% da amostra
    Nivel(i,4) = prctile(Md(:,i),80); % 80% da amostra
end

% 2o Passo - Distribuição dos Parametros nos níveis por estado emocional

Dist = zeros(4,5*31);

for i=1:1
    if (Md(i,32) == 'N') linha = 1; end % Obtem o estado emocional da
    amostra
    if (Md(i,32) == 'A') linha = 2; end % em questão para que se possa
    alocar
    if (Md(i,32) == 'T') linha = 3; end % corretamente na matriz de
    distribuição
    if (Md(i,32) == 'B') linha = 4; end

    for j=1:31
        if (Md(i,j) <= Nivel(j,1)) % Contador de
        frequencia para o Nivel I
            Dist(linha,j*5-4) = Dist(linha,j*5-4) + 1; % para o j-
        esimo parametro
        end
        if ((Nivel(j,1) < Md(i,j)) && (Md(i,j) <= Nivel(j,2))) % Contador de
        frequencia para o Nivel II
            Dist(linha,j*5-3) = Dist(linha,j*5-3) + 1; % para o j-
        ésimo parametro
        end
        if ((Nivel(j,2) < Md(i,j)) && (Md(i,j) <= Nivel(j,3))) % Contador de
        frequencia para o Nivel III
            Dist(linha,j*5-2) = Dist(linha,j*5-2) + 1; % para o j-
        ésimo parametro
        end
        if ((Nivel(j,3) < Md(i,j)) && (Md(i,j) <= Nivel(j,4))) % Contador de
        frequencia para o Nivel IV
            Dist(linha,j*5-1) = Dist(linha,j*5-1) + 1; % para o j-
        ésimo parametro
        end
        if (Md(i,j) > Nivel(j,4)) % Contador de
        frequencia para o Nivel V
```



```
                Dist(linha,j*5-0) = Dist(linha,j*5-0) + 1;           % para o j-
ésimo parametro
            end
        end % Reinicia a contagem para o próximo parametro
    end    % Finaliza distribuição
end
```

Anexo 12 – Função: analise_individuo

```
function [emocao, resposta] = analise_individuo(P, Dist, Nivel, Pesos)

% Pesos atribuidos a cada parametro - Pode-se utilizar como input ou
% alterar aqui manualmente

%           1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
%Pesos(1,1:17) = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];

%           18 19 20 21 22 23 24 25 26 27 28 29 30 31
%Pesos(1,18:31) = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];

indicador = zeros(4,5*31);

for i=1:31
    if (P(i) <= Nivel(i,1)) % Classifica o i-esimo
parametro % como Nivel I
        nivel = 1;
    end
    if ((Nivel(i,1) < P(i)) && (P(i) <= Nivel(i,2))) % Classifica o i-esimo
parametro % como Nivel II
        nivel = 2;
    end
    if ((Nivel(i,2) < P(i)) && (P(i) <= Nivel(i,3))) % Classifica o i-esimo
parametro % como Nivel III
        nivel = 3;
    end
    if ((Nivel(i,3) < P(i)) && (P(i) <= Nivel(i,4))) % Classifica o i-esimo
parametro % como Nivel IV
        nivel = 4;
    end
    if (P(i) > Nivel(i,4)) % Classifica o i-esimo
parametro % como Nivel V
        nivel = 5;
    end

    indicador(:,5*i-5+nivel) = Pesos(i); % Obtem o nivel do
parametro do individuo % com relacao a
% amostra e considera o

peso da variavel
end
    emocao(1,:) = sum((indicador.*Dist)'); % Mostra a "pontuacao"
cada estado emocional de acordo com os pesos
    emocao(2,:) = (sum((indicador.*Dist)')/sum(sum((indicador.*Dist)')))*100;
% Mostra a probabilidade de cada estado emocional em porcentagem
    disp(' '); % Pula linha
    disp('      N      A      T      B'); % Cabeçalho da matriz
    disp(emocao); % Resultado numérico
    disp(' '); % Pula linha
    [probabilidade , resposta] = max(emocao(2,:)); % Encontra a maior
probabilidade e para qual emocao ela ocorre
```

```
if (resposta == 1) disp('Resp: Estado Emocional Neutro'); end %  
if (resposta == 2) disp('Resp: Estado Emocional Alegre'); end %  
if (resposta == 3) disp('Resp: Estado Emocional Triste'); end %  
if (resposta == 4) disp('Resp: Estado Emocional Bravo'); end %  
disp(' '); %pula linha  
  
end % Fim do algoritmo
```

Anexo 13 – Função: teste_populacao

```
function confusao = teste_populacao(Dist, Nivel, Md, Pesos)

[l, c] = size(Md);
confusao = zeros(4);

for i=1:l
    [emocao, resposta] = analise_individuo(Md(i,1:31), Dist, Nivel, Pesos);
    %Analisa as amostras da população
    if (Md(i,32) == 'N') linha = 1; end % Identifica a resposta desejada pelo
    aprendizado
    if (Md(i,32) == 'A') linha = 2; end
    if (Md(i,32) == 'T') linha = 3; end
    if (Md(i,32) == 'B') linha = 4; end

    if (resposta == 1) coluna = 1; end % Identifica a resposta obtida pelo
    computador
    if (resposta == 2) coluna = 2; end
    if (resposta == 3) coluna = 3; end
    if (resposta == 4) coluna = 4; end

    confusao(linha, coluna) = confusao(linha, coluna) + 1; % Monta a matriz de
    confusão
end

end % Fim do Algoritmo
```

Anexo 14 – Matriz de Nível (Algoritmo Final)

PARÂMETROS	FRONTEIRAS			
	1ª	2ª	3ª	4ª
P(1)	188,0447	230,7259	263,1834	323,7537
P(2)	168,9453	209,6191	238,5742	275,7813
P(3)	416,9922	556,4453	677,9297	793,2617
P(4)	2,929688	112,0117	139,4531	178,7109
P(5)	259,8438	341,9922	465,791	588,3545
P(6)	128,3203	153,2227	179,8242	207,251
P(7)	328,7109	470,1172	563,0859	685,6445
P(8)	110,4688	179,5117	302,4561	404,3701
P(9)	43,39926	68,38412	94,8248	138,4802
P(10)	0,073509	0,083263	0,099702	0,122167
P(11)	0,068065	0,076756	0,089771	0,113715
P(12)	0,135338	0,195526	0,251133	0,331124
P(13)	0,110178	0,142467	0,176148	0,240757
P(14)	0,051283	0,052304	0,054024	0,057382
P(15)	0,056033	0,090867	0,123378	0,185022
P(16)	0,018563	0,028778	0,040499	0,059155
P(17)	1,188889	1,790245	2,878497	3,996875
P(18)	3,233529	3,657576	4,57627	6,837143
P(19)	1,932324	2,5396	3,82312	5,767237
P(20)	0,190759	0,428448	0,729511	1,100601
P(21)	6,285341	18,38684	32,47559	56,68533
P(22)	56,64063	136,8164	273,4375	359,1797
P(23)	14,31143	36,88987	75,60922	112,3068
P(24)	-111,279	-68,0339	-34,3657	-15,4785
P(25)	-322,949	-190,137	-103,32	-31,3477
P(26)	10,1897	43,83538	72,45802	132,4519
P(27)	2,00625	2,9625	4,53	6,3
P(28)	0,236636	0,289483	0,351021	0,438243
P(29)	0,142596	0,206913	0,282211	0,386061
P(30)	0,199327	0,237476	0,275707	0,30429
P(31)	1,000059	1,008119	1,018205	1,031872

Anexo 15 – Matriz de Distribuição (Algoritmo Final)

Estado Emocional	P(1)					P(2)					P(3)					P(4)				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
Neutro	7	10	3	2	0	7	6	6	3	0	7	6	6	2	1	4	7	3	6	2
Alegre	2	1	6	4	9	2	4	3	6	7	3	2	2	8	7	1	3	7	4	7
Triste	6	3	6	6	1	6	3	7	3	3	6	6	3	5	2	9	4	3	3	3
Bravo	3	3	3	5	8	4	3	2	5	8	2	3	7	2	8	5	2	5	5	5

Estado Emocional	P(5)					P(6)					P(7)					P(8)				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
Neutro	9	6	4	2	1	7	4	6	4	1	6	5	6	4	1	9	5	5	2	1
Alegre	0	3	4	6	9	1	6	4	4	7	4	2	3	6	7	1	2	4	7	8
Triste	7	5	5	3	2	5	3	6	5	3	5	5	6	3	3	7	6	4	3	2
Bravo	2	3	5	6	6	5	4	2	4	7	3	5	3	4	7	1	4	5	5	7

Estado Emocional	P(9)					P(10)					P(11)					P(12)				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
Neutro	9	5	4	3	1	10	7	4	1	0	10	8	3	1	0	10	6	5	1	0
Alegre	2	1	5	6	8	3	2	8	3	6	3	3	5	5	6	2	4	5	7	4
Triste	6	9	3	2	2	5	7	1	8	1	5	5	4	7	1	6	6	4	2	4
Bravo	1	2	6	6	7	0	1	5	5	11	0	1	6	4	11	0	1	4	7	10

Estado Emocional	P(13)					P(14)					P(15)					P(16)				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
Neutro	10	7	4	1	0	10	6	3	2	1	10	7	4	1	0	9	7	5	1	0
Alegre	2	3	7	5	5	2	4	7	4	5	2	3	7	5	5	2	3	8	4	5
Triste	6	6	2	4	4	6	5	4	5	2	6	6	2	4	4	7	5	2	5	3
Bravo	0	1	5	7	9	0	2	4	6	10	0	1	5	7	9	0	2	3	7	10

Estado Emocional	P(17)					P(18)					P(19)					P(20)				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
Neutro	9	8	5	0	0	2	2	6	6	6	1	5	4	6	6	9	6	7	0	0
Alegre	2	4	6	6	4	6	8	3	4	1	7	6	3	4	2	2	4	4	7	5
Triste	7	4	2	5	4	2	2	3	7	8	1	3	4	6	8	7	7	0	5	3
Bravo	0	1	5	6	10	8	5	6	0	3	9	3	7	1	2	0	0	7	5	10

Estado Emocional	P(21)					P(22)					P(23)					P(24)				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
Neutro	12	3	2	4	1	9	6	2	3	2	9	6	2	3	2	1	5	4	4	8
Alegre	2	1	6	6	7	3	2	5	3	9	2	2	4	5	9	5	5	5	5	2
Triste	4	11	5	1	1	6	7	6	2	1	5	8	6	2	1	2	2	4	6	8
Bravo	0	2	5	6	9	1	1	6	8	6	2	1	6	7	6	10	5	5	2	0

Estado Emocional	P(25)					P(26)					P(27)					P(28)				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
Neutro	2	3	5	4	8	7	5	4	4	2	0	5	6	6	5	3	4	5	4	6
Alegre	4	5	8	2	3	2	3	7	5	5	6	4	4	5	3	4	6	4	6	2
Triste	1	4	2	8	7	8	7	3	2	2	3	4	3	4	8	6	1	6	4	5
Bravo	11	5	3	3	0	1	2	4	6	9	9	4	5	2	2	5	6	3	3	5

Estado Emocional	P(29)					P(30)					P(31)				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
Neutro	6	3	1	6	6	3	0	7	5	7	3	7	6	5	1
Alegre	4	4	7	5	2	5	4	2	6	5	9	4	1	3	5
Triste	3	4	7	3	5	6	8	4	1	3	3	3	5	5	6
Bravo	5	6	3	3	5	4	5	5	5	3	3	3	6	4	6

Anexo 16 – Pesos (Algoritmo Final)

PARÂMETROS	PESOS
P(1)	0,3064306
P(2)	0,287083494
P(3)	0,376655188
P(4)	0,44845935
P(5)	0,443993489
P(6)	0,127902529
P(7)	0,374570639
P(8)	0,653087042
P(9)	0,601386857
P(10)	0,372317227
P(11)	0,378053719
P(12)	0,546025829
P(13)	0,445543408
P(14)	0,099152065
P(15)	0,607289968
P(16)	0,627840674
P(17)	0,86075438
P(18)	0,38240764
P(19)	0,490702188
P(20)	1
P(21)	0,909131593
P(22)	0,739337838
P(23)	0,77001488
P(24)	0,870778226
P(25)	0,948240777
P(26)	0,869557649
P(27)	0,47196833
P(28)	0,113075893
P(29)	0,148868775
P(30)	0,115661247
P(31)	0,006399551

9. Referências Bibliográficas

- [1] TOIVANEN, J. - Automatic recognition of emotions in spoken Finnish. University of Oulu: 2005.
- [2] COZMAN, F. G. – An Introduction to the Theory of Sets of Probability Distributions. Escola Politécnica: 2008.
- [3] KANDEL, E. R.; SCHWARTZ, J. H. – Principles of Neural Sciences, Second Edition. Elsevier, 1985.
- [4] CHAPMAN, S. J. - Matlab Programming for Engineers. Thomson Engineering, 2005.
- [5] MARPLE, S. L. - Digital Spectral Analysis. New Jersey: Prentice-Hall, 1986.
- [6] FAIRBANKS, G. F. – Voice and Articulation Drillbook. 2nd ed. New York: Harper & Row, 1960.