INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Área Departamental de Engenharia de Electrónica e Telecomunicações e de Computadores

Classificação e identificação de eventos sonoros em ambiente urbano

PEDRO MIGUEL SANTANA GUERREIRO

(Licenciado)

Dissertação para obtenção do Grau de Mestre
em Engenharia Electrónica e Telecomunicações

Orientadores : Doutor Gonçalo Marques
Doutor Joel Paulo

Júri:

Presidente: Carlos Meneses

Vogais: Rui Jesus
Gonçalo Marques

Dezembro, 2018

Agradecimentos

Ao decorrer da dissertação, tive a oportunidade de verificar que apesar de ser um processo solitário, a que qualquer investigador está destinado, reúne contributos e apoios de várias pessoas externas ao mesmo, em que sem estes esta investigação teria sido mais difícil.

Primeiro que tudo quero agradecer aos meus orientadores, Eng. Gonçalo Marques e Eng. Joel Paulo. Foi com grande entusiasmo que me orientaram neste período da minha vida universitária. Estiveram sempre conectados, preocupados e presentes sobre o que ia fazendo e o que estava por fazer, possibilitando sempre o diálogo entre as duas partes de maneira a permitir que pudesse propor novos desafios e ideias a acrescentar à dissertação.

De seguida quero agradecer a todos os meus colegas e amigos, especialmente a todos aqueles que se mostraram preocupados e me ajudaram nesta etapa, que de forma anímica me motivaram e animaram para que conseguisse levar a dissertação adiante.

Depois agradecer ao meu grande pilar, que por muitas vezes foi o meu refúgio e teve a paciência de aguentar todas as barreiras e dificuldades que me foram apresentadas, transmitindo-me a confiança e motivação necessárias. Desta forma quero agradecer à minha família, especialmente aos meus pais e à minha irmã.

Por último e não menos importante que os anteriores, quero agradecer a Deus pela sanidade mental e espiritual concedida durante todo este processo, inclusive na escrita da mesma.

Resumo

O som ainda é uma medida ambiental pouco explorada, que transporta grande quantidade de informação sobre o ambiente em que se insere, o que leva à oportunidade de investigar novas dimensões e interações em ambientes urbanos.

A necessidade de detetar e classificar, automaticamente, sons urbanos complexos e dinâmicos levou ao surgimento de novos métodos, que dependem exclusivamente de características extraídas do sinal de áudio, conhecidos como métodos baseados em conteúdo. Consequentemente as pesquisas efetuadas nesta área, proporcionaram o surgimento de inúmeras aplicações urbanas, tais como: controlo eficiente do ruído, gestão de tráfego, vigilância e mapeamento da paisagem sonora. Porém, ao ser uma área relativamente nova, a sua limitação deve-se ao facto de não possuir uma taxonomia comum e bem definido.

Deste modo, o objetivo da dissertação é desenvolver um sistema eficiente, baseado em aprendizagem automática, para classificar e identificar eventos sonoros urbanos em condições reais de ruído. Para este propósito, será descrito em detalhe o esquemático do sistema em termos dos seus blocos de processamento e apresentados os resultados obtidos, para diferentes classificadores e atributos.

Para avaliar o sistema apresentado, foram usadas duas bases de dados disponíveis ao público, a UrbanSound4k e a ESC-50.

Relativamente aos resultados obtidos, o sistema supera as implementações efetuadas em *baseline* e atinge resultados comparáveis a outras abordagens similares.

Palavras-chave: Classificação e identificação de som urbano, aprendizagem automática, processamento de sinal, análise do conteúdo do sinal de áudio, etiquetagem automática.

Abstract

The sound is still an unexplored environmental measure, which carries a lot of information about the environment in which it is inserted, which leads to the opportunity to investigate new dimensions and interactions in urban environments.

The need to automatically detect and classify complex and dynamic urban sounds led to the emergence of new methods, which depend only on characteristics extracted from the audio signal, known as content-based methods. Consequently, the researches carried out in this area have provided the emergence of numerous urban applications, such as: efficient noise control, traffic management, surveillance and mapping of the sound landscape. However, as a relatively new area, its limitation is due to the fact that it does not have a common and well-defined taxonomy.

Thus, the aim of the dissertation is to develop an efficient system, machine learning system, to classify and identify urban sound events in real noise conditions. For this purpose, it will be described in detail the design of the system in terms of its processing blocks and presented the results obtained, for different classifiers and attributes.

To evaluate the system presented, were used two databases available to the public, the UrbanSound4k and the ESC-50.

Regarding the results obtained, the system overcome the implementations performed at baseline and achieves results comparable to other similar approaches.

Keywords: Urban sound classification and identification, automatic learning, signal processing, audio signal content analysis, autotaggings.

Índice

Lista de Figuras

Lista de Tabelas

Lista de Acrónimos

IoT – Internet of Things

HRQOL – Health-Related Quality of Life

ASR – Automatic Speech Recognition

MIR – Music Information Retrieval

CASA – Computational Auditory Scene Analysis

FFT – Fast Fourier Transform

DFT – Discrete Fourier Transform

iFFT – Inverse Fast Fourier Transform

DTMF – Dual Tone Multi Frequency

STFT – Short-Time Fourier Transform

MFCCs – Mel Frequency Cepstral Coefficients

PCA – Principal Component Analysis

LDA – Linear Component Analysis

kNN – k-Nearest Neighbors

SVM – Support Vector Machine

MLP – Multilayer Perceptron

TP – True Positives

FN – False Negatives

FP – False Positives

TN – True Negatives

ROC – Receiver Operating Characteristics

AUC – Area Under the Curve

BoF – Bag-of-Frames

ESC – Environmental Sound Classification

FG – Foreground

BG – Background

SNR – Signal-to-Noise Ratio

Capítulo 1

Introdução

Este capítulo tem como objetivo revelar os motivos que levaram à realização da dissertação, destacando a sua importância na classificação e identificação de eventos sonoros em ambiente urbano. Além disso, também serão abordadas algumas das aplicações que têm por base o som urbano e recorrem a metodologias concorrentes à aqui apresentada. Por fim, será apresentada a estrutura da dissertação.

1.1 Motivação

A rápida urbanização mundial representa um sério desafio para a sociedade humana, pois as cidades estão a ficar cada vez mais populosas levando a impactos de ruído ambiental, gerando assim uma preocupação crescente [World Health Organization, 2011]. Hoje, 54% da população mundial vive em cidades e até 2050, segundo as Nações Unidas, estima-se que esse número chegue a 68%, com cidades da China, Sudeste Asiático e América Latina a apresentarem um enorme crescimento populacional [United Nations, 2018]. Ao ocorrer este aumento populacional, o ruído torna-se cada vez mais uma preocupação, não só a nível de saúde pública [Ising and Kruppa, 2004] como também a nível da segurança e meio ambiente.

Para a maioria das pessoas, a capacidade de ouvir e identificar eventos sonoros é tão natural que é dado como certo. No entanto, esta é uma tarefa muito desafiadora para os computadores, pois a criação de máquinas que reconhecem automaticamente os eventos sonoros, continua a ser um problema em aberto.

O conceito de paisagem sonora tem atraído uma atenção especial na última década, envolvendo investigadores de diversas áreas disciplinares [Steele et al., 2013], sendo composta por diferentes sons que compõe um determinado ambiente, sejam estes de origem natural, humana, industrial ou tecnológica, produzindo assim um sinal sonoro complexo. A análise deste sinal complexo, juntamente com informações de tempo e localização, pode fornecer uma melhor compreensão sobre que aspetos sonoros mais afetam a rotina dos cidadãos e assim poder melhorar a sua qualidade de vida.

Com o intuito de promover serviços urbanos eficientes com baixos custos e menor consumo de recursos, começou a surgir o conceito de *smart city*, que está a ser desenvolvido a nível interdisciplinar e que conta com as tecnologias de informação e comunicação, *machine learning* e *Internet of Things* (IoT) para otimizar as funções da cidade e atender a questões convencionais relacionadas à cidade, meio ambiente e saúde [Zheng et al., 2005, The Economist, 2012]. Uma cidade é uma entidade complexa e dinâmica, cheia de movimentação, interações e fluxos, ficando inegavelmente ligada ao fenómeno físico do som.

Nos dias de hoje, os principais métodos para caracterizar as paisagens sonoras, são baseados na medição dos níveis de pressão sonora [Liu et al., 2014] e a informação da fonte de ruído é frequentemente negligenciada. Contudo, a redução do nível de ruído não conduz necessariamente a uma melhoria da qualidade de vida nas zonas urbanas, sendo essencial conhecer a fonte de ruído. De acordo com o estudo apresentado em [Héritier et al., 2014], sons urbanos típicos como o ruído do tráfego ferroviário, rodoviário, aéreo, os ruídos da vizinhança e da indústria apresentam diferentes impactos na qualidade de vida relacionada à saúde, ou do inglês *Health-Related Quality Of Life* (HRQOL).

Para além das questões convencionais relacionadas com o ruído e a sua identificação, há outros interesses que advêm da análise da paisagem sonora. Um dos interesses poderia ser a qualificação das cidades dum ponto de vista sonoro, isto é, identificar quais os sons mais peculiares e representativos de cada cidade e assim fornecer um indicador cultural, através do som. A título de exemplo, a cidade de Lisboa possui muitos sons característicos que não se encontram em muitas outras cidades, como o som dos elétricos, dos amola tesouras, entre outros. Outro interesse poderia ser o reconhecimento de situações de alerta, ou seja, identificar eventos sonoros como tiros, acidentes de carro, sirenes, gritos, vidros a partir e assim informar de imediato as autoridades mais próximas.

Os eventos sonoros geralmente têm uma duração bem definida e breve no tempo. Desta maneira, o objetivo principal da análise computacional de cenas e eventos sonoros urbanos [Virtanen et al., 2018], é extrair características do áudio recebido, por métodos computacionais, de modo a ser possível detetá-lo e de seguida classificá-lo. Estas tarefas exigem não só o uso de várias técnicas relacionadas ao processamento de sinais de áudio, mas também de aprendizagem automática.

Esta dissertação foca-se principalmente nas técnicas de classificação automática, que dependem exclusivamente de informações extraídas do sinal de áudio – conhecidas como métodos baseados em conteúdo, tendo como objetivo criar um sistema capaz de “escutar” e “entender” o som recebido por conta própria, apesar desta finalidade ainda ser uma utopia.

1.2 Estado de arte

A análise computacional de cenas e eventos sonoros urbanos é uma área relativamente nova, no qual a omissão de dados públicos e um vocábulo comum e bem definido têm proporcionado a sua limitação [Salamon et al., 2014]. Contudo não começou do zero, pois utiliza técnicas que advém do processamento de fala, Automatic Speech Recognition (ASR) [Kindcaid, 2018], e do processamento de música, Music Information Retrieval (MIR) [Klapui and Davy, 2007]. Onde se evidencia a extração de características baseadas nos coeficientes espetrais de frequência Mel (MFCC's), a técnica *bag-of-words*, a classificação *multi-class*, etiquetagem automática (*multi-label*), etc. Tudo conceitos que serão abordados nos próximos capítulos em pormenor.

Deste modo, apesar de assentar nas áreas acima citadas, beneficiou não só da evolução da inteligência artificial como do *Computational Auditory Scene Analysis* (CASA) [Wang and Brown, 2006], que visa modelar partes funcionais do sistema auditivo, computacionalmente, para produzir sistemas cada vez mais capazes de separar misturas de fontes sonoras, da mesma maneira que os ouvidos humanos o fazem.

Desta forma, começou a surgir a existência de aplicações inovadoras que usam o som urbano, por exemplo para fazer vigilância por áudio [Crocco et al., 2016, Radhakrishnan et al., 2005], gestão de tráfego [Nagy et al., 2014], mapeamento da paisagem sonora [Rychtáriková and Vermein, 2013, Toriza et al., 2014], controlo de ruído [Agha et al., 2017, Kivela et al., 2011, Mydlarz et al., 2017], etc. Progressos significativos também foram feitos pelo projeto "*Soundscapes of European Cities and Landscapes*", que se dedicou à pesquisa de paisagens sonoras em diferentes vertentes, sendo elas definição, avaliação e modelação psico-acústica [Botteldooren et al., 2013]. O projeto "EART-IT" [Pham and Cousin, 2013], baseia-se no som para produzir aplicações de alto valor social, que fazem parte sistemas de vigilância e controlo de emergências.

Por sua vez, a análise dos eventos sonoros pode ser ainda separada em dois problemas: a deteção e a classificação. A deteção de eventos sonoros visa identificar as marcas iniciais e finais de um determinado evento, enquanto que na classificação o propósito é classificar cada evento em diferentes classes.

Em suma, pode aferir-se que numa cidade o som torna-se numa fonte rica de informação e muito importante para muitas aplicações, as quais têm como foco o som urbano, pois apesar de conseguir ser monitorizado em não linha de vista, também pode ser utilizada para inúmeros fins.

1.3 Publicação

A pesquisa apresentada nesta dissertação, resultou na publicação de um artigo para o “Jornal Académico ISEL de Eletrónica, Telecomunicações e Computadores” (i-ETC). Esta revista apresenta contribuições originais e artigos de revisão de pesquisa teórica e experimental, nos amplos campos de eletrónica, telecomunicações e computadores:

• J. Alves, P. Guerreiro, G. Marques and J. Paulo. A Low-Cost Urban Sound Event Detection and Identification System for Urban Environments. Audio and Acoustics Laboratory of ISEL, Lisbon, Portugal, 2018

1.4 Estrutura da dissertação

Esta dissertação está estruturada da seguinte forma: O Capítulo 2 introduz e formaliza alguns aspetos importantes do processamento de sinal, necessários na criação dum sistema para classificação e identificação de eventos sonoros urbanos. Descreve também as características de nível baixo, usualmente usadas neste cenário, baseadas em informações retiradas do sinal de áudio. Por fim, é dada uma breve explicação sobre os algoritmos de classificação usados e a introdução de dois cenários de etiquetagem, o *multi-class* e o *multi-label*. O Capítulo 3 apresenta a implementação adotada, fornecendo os procedimentos e abordagens que foram aplicados ao sistema desenvolvido. O Capítulo 4 tem como propósito demonstrar uma ampla avaliação dos métodos utilizados para várias tarefas de classificação e numa variedade de conjuntos de dados, comparando-os com resultados alcançados por outros investigadores. O Capítulo 5 conclui a dissertação e discute futuras direções de pesquisa.

Capítulo 2

Extração de Informações Sonoras Baseadas em Conteúdo

Este capítulo é uma introdução aos métodos baseados em conteúdo. Aqui apresenta-se a terminologia e o processamento de sinal que é efetuado. Primeiramente, é dada uma breve visão geral da extração de informações sonoras, de seguida descreve-se os fundamentos do processamento de sinais de áudio digital, em particular as representações do domínio de tempo/frequência, que são a base da maioria das características de nível baixo. Posteriormente, abordam-se métodos com o objetivo de normalizar as características, recorrendo a transformações lineares.

2.1 Origem da extração de informações sonoras – Breve visão geral

Como foi dito anteriormente, a classificação e identificação de eventos sonoros urbanos assenta em duas áreas mais tradicionais e que começaram as suas pesquisas há mais de meio século. Deste modo, neste subcapítulo é dada uma breve introdução de ambas as áreas, sendo que falaremos do ASR e do MIR, respetivamente.

A fala é o principal meio de comunicação entre as pessoas, o que tem intrigado engenheiros e cientistas em compreender e automatizar as capacidades mecânicas da fala humana, de forma que o ASR tem atraído muita atenção nas últimas décadas. É necessário recuar aos anos 1930, quando Homer Dudley, da Bell Laboratories, propôs um modelo de sistema para análise e síntese de fala [Dudley et al., 1939]. O reconhecimento automático de fala tem evoluído progressivamente, transitando de uma máquina simples que responde a um pequeno conjuntos de sons, a sistemas mais sofisticados capazes de responder à linguagem natural humana. Baseados em grandes avanços nos modelos estatísticos de fala, da década de 1980, os sistemas atuais encontram amplas aplicações tais como o processamento automático de chamadas de rede telefónica, sistemas de informações, entre outros.

Relativamente ao MIR, as primeiras publicações mencionando recuperação da informação musical datam de meio século [Kassler, 1966], mas durante décadas a área não recebeu muita atenção. Hoje em dia, as coisas mudaram consideravelmente, e o interesse da pesquisa expandiu-se significativamente, em grande parte devido ao sucesso dos media e novas formas de adquirir, ouvir e processar sinais sonoros.

No entanto, na classificação de eventos sonoros é preciso atribuir rótulos a um excerto de áudio, sendo necessário recorrer às tarefas de reconhecimento de um locutor e género musical, usadas no ASR e MIR, de forma a simular o mecanismo do sistema auditivo humano. A audição humana é sem dúvida um prodígio da evolução natural, pois é capaz de realizar ações que máquinas modernas são incapazes de realizar, ela é capaz de distinguir diversos eventos sonoros numa paisagem sonora, isto é, tem a capacidade de num excerto de som ser capaz de ouvir individualmente pessoas a falar, carros a apitar, cães a ladrar, sirenes, etc. Desta forma, apesar de uma "entrada" de sons tão complexa, é capaz de ouvir seletivamente o que é desejado.

A primeira pessoa a tentar dar resposta ao funcionamento do sistema auditivo humano foi Bregman através do seu livro, Auditorial Scene Analysis [Bregman, 1990], em 1990. As suas ideias sobre a análise de cenas auditivas, forneceram não só uma nova estrutura para pesquisas sobre os sistemas auditivos humanos, como também para estudos comportamentais e neurológicos da perceção da fala, aparelhos auditivos, etc. Em reconhecimento destas contribuições, foi batizado como o "pai da análise de eventos auditivos".

Desta forma, e inspirados pelas teorias de Bregman, muitos sistemas computacionais têm sido propostos com o intuito de se assemelharem, o melhor que possível, ao sistema auditivo humano, por forma a criar sistemas de escuta cada vez mais eficientes.

Em suma, verifica-se um grande paralelismo entre a análise de eventos sonoros urbanos com as áreas ASR e MIR, pois partilham muitas características e métodos de análise de som.

2.2 Descritores de conteúdo

2.2.1 Atributos do som

A maioria das técnicas de reconhecimento automático do som são baseadas em vários conceitos e a sua eficácia depende de como estes conceitos são modelados. Antes de serem discutidas as técnicas utilizadas na extração da informação sonora, é necessário abordar os atributos percetivos do som. O som pode ser caracterizado em três atributos subjetivos [Moore, 1995, Rossing, 1990]:

Pitch: O *pitch* é o atributo humano que permite que os sons sejam ordenados numa escala de frequência, variando de baixas para altas. O tom está intimamente relacionado com à frequência fundamental (F0), que é a sua contraparte física. A frequência fundamental é definida por sinais periódicos ou quase periódicos e é o inverso de um período.

Loudness: O *loudness* é como os humanos percebem o nível de amplitude do som e, portanto, a energia presente no sinal. No contexto do processamento sonoro, é comum expressar o nível de som em decibéis, que resulta da aplicação de uma escala logarítmica à potência quadrática média do sinal, principalmente para lidar com a grande variedade de faixas dinâmicas envolvidas. O volume sonoro é por vezes referido como intensidade.

Timbre: O timbre é a qualidade percetiva que permite discriminar entre diferentes sons com o mesmo tom e intensidade. Por exemplo, uma nota tocada com a mesma intensidade por dois instrumentos diferentes (por exemplo, uma flauta e um violoncelo) é facilmente distinguível.

A conexão destes três atributos à sua contraparte física não é trivial e constitui alguns dos aspetos fundamentais da psico-acústica. No entanto, os dois primeiros, *pitch* e *loudness*, podem ser razoavelmente aproximados pela frequência fundamental e pela energia do sinal. Por outro lado, o timbre não tem uma contraparte física simples, nem pode ser facilmente codificado num único valor escalar. Desta maneira, faz com que a definição para o timbre seja mais pela negativa, isto é, o que não é *pitch* nem *loudness*, pois estas apenas dependem principalmente da distribuição de energia espetral do som e a sua evolução temporal.

2.2.2 Hierarquia das características

Para implementar métricas baseadas em conteúdo, é necessário extrair do sinal de áudio um conjunto de características adequadas. Num sentido muito amplo, uma característica é uma descrição compacta de uma informação particular presente no sinal de áudio. Deste modo, é importante abordar a questão de quais características utilizar. Apesar do conteúdo poder ser dividido em três camadas de descrição estruturadas hierarquicamente, sendo elas o nível baixo, nível médio e nível alto [Bello, 2017], nesta dissertação apenas foram utilizados os descritores de nível baixo, pois os restantes são descritores mais relacionados com informações musicais, pertencentes à área de pesquisa MIR.

Os descritores de nível baixo são aqueles que podem ser calculados diretamente a partir do sinal de áudio, ou derivados após alguma transformação do sinal, como as transformadas de Fourier ou Wavelet. Esta classe de características também é denominada de descritores centrados no sinal e a maioria é relacionada ao conteúdo espetral do sinal, podendo também ser obtidas através do conteúdo temporal do sinal. Uma revisão mais detalhada de alguns desses recursos, principalmente os usados nesta dissertação, será dada na próxima seção.

Em suma, as características de nível baixo são o ponto de partida para a grande maioria dos métodos apresentados na literatura, são fáceis de extrair e mostram bons desempenhos em uma ampla gama de tarefas de classificação. As características usadas neste projeto estão diretamente relacionadas ao espetrograma do sinal (com a exceção do *zero crossing rate*). Em seguida, apresenta-se uma breve visão geral dos conceitos de processamento de sinal necessários para calcular essas características.

2.2.3 Descritores de nível baixo

Existe uma ampla gama de características de nível baixo, que podem ser calculadas a partir dos sinais de áudio, a sua maioria tem origem no campo de pesquisa bem estabelecido do processamento de sinal e reconhecimento automático de fala. Para extrair um conjunto de características a partir de um som, este precisa de ser convertido num formato digital. Além disso, uma etapa da re-amostragem é frequentemente realizada para garantir que todos os sinais tenham a mesma frequência de amostragem. Neste projeto, trabalhou-se com uma frequência de amostragem, para todos os sinais de áudio, de 44100 Hz.

Representações na frequência

Análise espetral é o processo de caracterização de sinais no domínio da frequência em termos de componentes individuais de frequência. Qualquer sinal real temporal pode ser convertido no domínio da frequência e voltar à sua forma original, sem qualquer perda de informação, através da transformada de Fourier. Entende-se por espetro as variações de amplitude e fase do sinal versus frequência, o que em muitos casos é uma descrição mais simples e mais intuitiva do que à contrapartida no domínio do tempo. As representações espetrais são particularmente úteis para sinais de áudio, pois destacam características distintas do som. Consequentemente, não é surpreendente que a grande maioria das características de nível baixo sejam derivadas de representações do domínio da frequência.

O ponto de partida é dividir o sinal de áudio em segmentos de curta duração, janelas, e de seguida calcular o espetro para cada janela. O espetro do sinal é facilmente obtido, usando os algoritmos *Fast Fourier Transform* (FFT), que são implementações mais eficientes do que a *Discrete Fourier Transform* (DFT).

A DFT é uma transformação usada na análise de Fourier, que projeta um sinal discreto do domínio do tempo, numa representação discreta no domínio da frequência (e de volta via o inverso iDFT). De um ponto de vista matemático, esta transformação é exata para sinais periódicos em tempo discreto, mas também uma ferramenta importante de análise para sinais de tempo discreto de comprimento finito que possuem uma representação espetral contínua. Neste caso, o uso da DFT pode ser interpretada como uma amostragem do espetro contínuo, em intervalos espaçados regularmente. Formalmente, a DFT de um sinal periódico de tempo discreto $x[n]$ com um período de N amostras de comprimento, é representada por:

$$X[k] = DFT\big[x[n]\big] = \sum_{n=0}^{N-1} x[n] e^{-\frac{j2\pi kn}{N}} \qquad k \in \{0, 1, \dots, N-1\} \tag{2.1}$$

Onde j e $X[k]$ representam a magnitude e a fase em cada um dos valores de k (intervalos de frequência). Geralmente $X[k]$ é apresentado separadamente como o espetro de magnitude, $|X[k]|$, e espetro de fase φ[k]:

$$X[k] = Re(X[k]) + jIm(X[k]) = |X[k]| e^{j\phi[k]} \tag{2.2}$$

Com:

$$|X[k]| = \sqrt{Re(X[k])^2 + Im(X[k])^2} \tag{2.3}$$

$$\phi[k] = \arctan\frac{Re(X[k])}{Im(X[k])} \tag{2.4}$$

Em termos práticos, o espetro de magnitude contém a maioria das informações relevantes. Através da análise da Figura 1 e da Tabela 1, podemos verificar frequências de amostragem, para 4 valores de N.

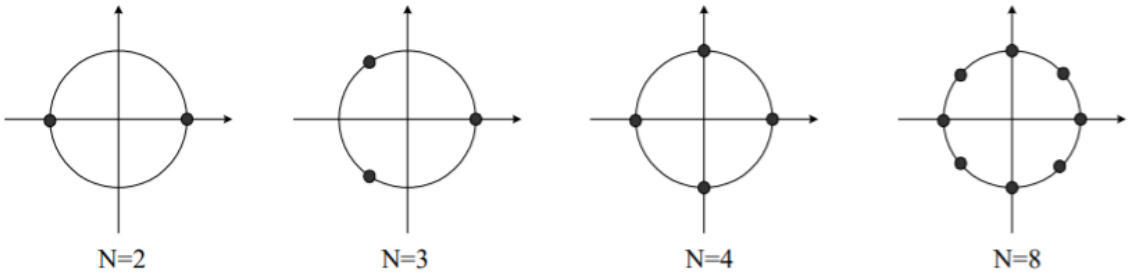

Figura 1 - Frequências de amostragem para 4 valores de N. Fonte: Ferreira, 2017

Tabela 1 - Frequências digitais [rad/s] para 4 valores de N

N	Frequências digitais [rad/s]
2	$\{0, \pi\}$
3	$\{0, \frac{2\pi}{3}, -\frac{2\pi}{3}\}$
4	$\{0, \frac{\pi}{2}, \pi, -\frac{\pi}{2}\}$
8	$\{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}, \pi, -\frac{3\pi}{4}, -\frac{\pi}{2}, -\frac{\pi}{4}\}$

Por sua vez, a IDFT é descrita por:

$$x[n] = IDFT\big[X[k]\big] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{\frac{j2\pi kn}{N}} \quad n \in \{0, 1, \dots, N-1\} \tag{2.5}$$

Apesar da DFT ser uma ferramenta essencial para converter um sinal do domínio do tempo numa representação no domínio da frequência, ao analisar um sinal apenas em termos de frequência não é o suficiente. Isto acontece porque os sinais são compostos por uma combinação de eventos sonoros localizados no tempo, que também possuem assinaturas de

frequências específicas. A DFT dá-nos um conteúdo espetral de todo o sinal, fazendo com que as características de frequência de partes individuais do sinal, assim como a sua evolução entre diferentes eventos sonoros, sejam perdidas. A solução é "cortar" o áudio numa série de segmentos de curta duração, janelas, e observar a progressão do espetro de frequência dos segmentos individuais. Isto é chamado de *Short-Time Fourier Transform* (STFT) e a sua magnitude ao quadrado produz o espetrograma. A Figura 2 mostra exemplos de espetrogramas tempo-frequência de sinais de voz, música, sinusoide a 2 kHz e ruído.

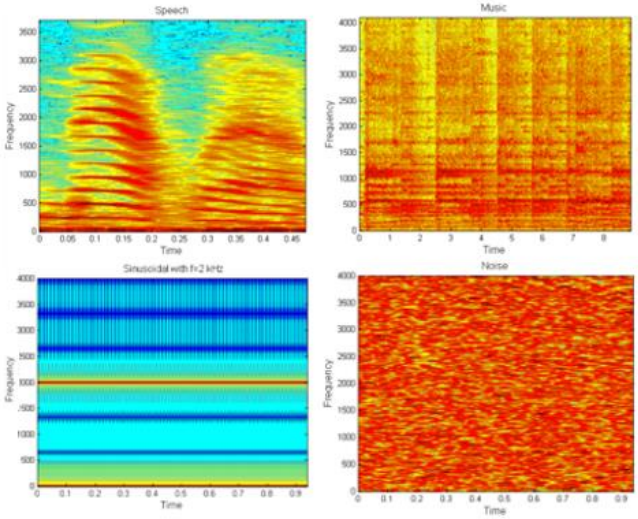

Figura 2 - Espetrogramas tempo-frequência para diferentes sinais. Fonte: Ferreira, 2017

Formalmente, a STFT de qualquer janela de um sinal $x[n]$, pode ser expressa em termos de um produto desse sinal com uma função de janela:

$$\mathrm{X[k,m]} = \sum_{\mathrm{n}} \mathrm{x[n]w[n-m]e}^{-\frac{\mathrm{j2\pi nk}}{\mathrm{N}}} \qquad \mathrm{n,k} \in \{0,1,\dots,\mathrm{N}-1\} \tag{2.6}$$

Onde m é o índice de tempo da janela e $w[n]$ é a função de janela de comprimento N. Esta equação fornece a representação de frequência discreta de um segmento do sinal $x[n]$. Para obter o espetrograma, é necessário efetuar este cálculo para todo o sinal. A função de janela tem um impacto direto na STFT, pois há limitações e compensações na resolução que são inerentes à escolha da forma e do tamanho da $w[n]$. Ao multiplicar o sinal $x[n]$ pela função de janela, no domínio da frequência, é o equivalente à convolução dos espetros de Fourier individuais do sinal, $X[k]$, e da janela $W[k]$. A convolução dos dois espetros pode criar componentes de frequência em $X[k,m]$, que não estavam presentes no sinal original. Este

efeito é conhecido como vazamento espetral ou *spectral leakage* [Higuti, 1989/1999]. Desta maneira, há funções janela com diferentes formas (Retangular, Hanning, Hamming, Kaiser, Gaussian, Blackman-harris, etc) que tentam de uma certa forma suavizar este efeito. A Figura 3 evidencia as funções de janela no domínio do tempo e a Figura 4 no domínio da frequência.

Contudo, escolher uma função janela requer um compromisso entre a resolução, no tempo e frequência, e a aplicação usada. De seguida, serão dados alguns exemplos deste compromisso:

• Se o sinal de áudio contiver frequências interferentes distantes da frequência de interesse, é preferível escolher uma função janela que apresenta maiores taxas de atenuação dos lóbulos laterais.

• Se o sinal contiver frequências interferentes próximas da frequência de interesse, deve-se optar por uma função janela que possua lóbulos laterais baixos.

• Se a frequência de interesse contiver dois ou mais sinais que estejam muito próximos um do outro, a resolução espetral é importante, deste modo a melhor solução é uma função janela com o lóbulo principal bastante estreito.

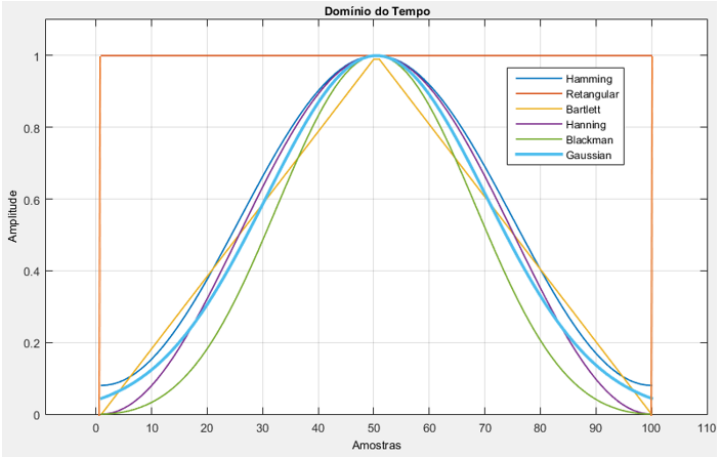

Figura 3 - Funções de janela no domínio do tempo

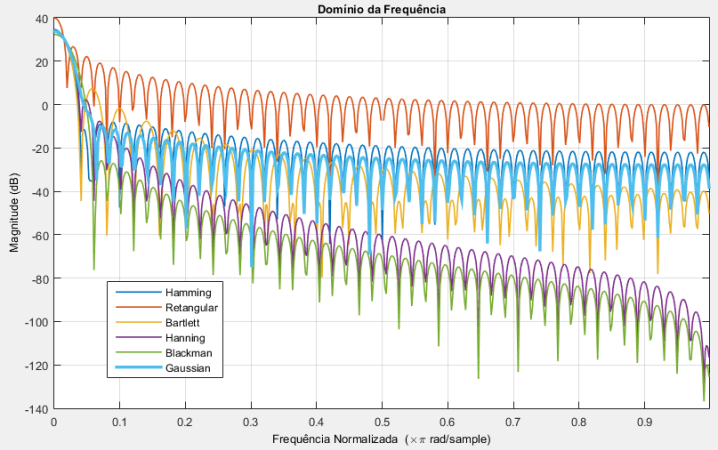

Figura 4 - Funções de janela no domínio da frequência

Ao longo desta dissertação, foi usada a janela Hanning, pois oferece boa resolução em frequência e dispersão espetral reduzida, como função de janela padrão:

$$w[n] = \frac{1}{2}(1 - \cos(\frac{2\pi(n-1)}{N})) \tag{2.7}$$

Outro fator que tem um impacto direto na STFT é o comprimento da função janela. Uma janela maior fornece uma boa resolução na frequência, apresentando linhas horizontais mais nítidas, contudo a sua resolução temporal é menos precisa. Enquanto uma janela pequena retrata o efeito oposto. Este compromisso está representado na Figura 5, onde vários espetrogramas de dois sinais de áudio são calculados com diferentes janelas de tempo. A resolução do espetrograma também é afetada pelo tamanho do salto: o número de amostras entre duas janelas consecutivas. Quanto menor o tamanho do salto, mais suave a resolução do espetrograma. Geralmente para economizar recursos de processamento e memória são usados tamanhos de salto de metade ou uma janela.

No entanto, a resolução de frequência do espetrograma definido pelo STFT é linear e não corresponde ao modo como os seres humanos percebem o som. Muitos estudos em psico-acústica mostram que o sistema auditivo humano tem uma resolução de frequência logarítmica. Para explicar este facto e com base em experiências sobre a audição humana, muitas escalas psico-acústicas foram propostas, como Mel, Bark, Equivalent Rectangular Bandwidth, entre outras. A escala Mel é a escolha habitual para métodos de classificação. O processo de mapeamento do espetro de frequências linear, obtido via STFT numa escala

psico-acústica, está intimamente relacionado com os bancos de filtros auditivos, que são filtros de passa-banda não uniformes para imitar o sistema auditivo humano. Uma nova representação de frequência pode ser obtida calculando a energia espetral em cada banda de filtros. Desta forma, uma correspondência mais próxima ao sistema auditivo humano é obtida e o número de coeficientes espetrais por janela é reduzido ao número de bandas.

Posteriormente, será descrito as características utilizadas neste trabalho que são usadas na classificação. Falar-se-á dos coeficientes espetrais de Mel (MFCCs) e de seguida outros descritores que complementam a informação dos MFCCs, a maioria baseada na magnitude espetral do sinal.

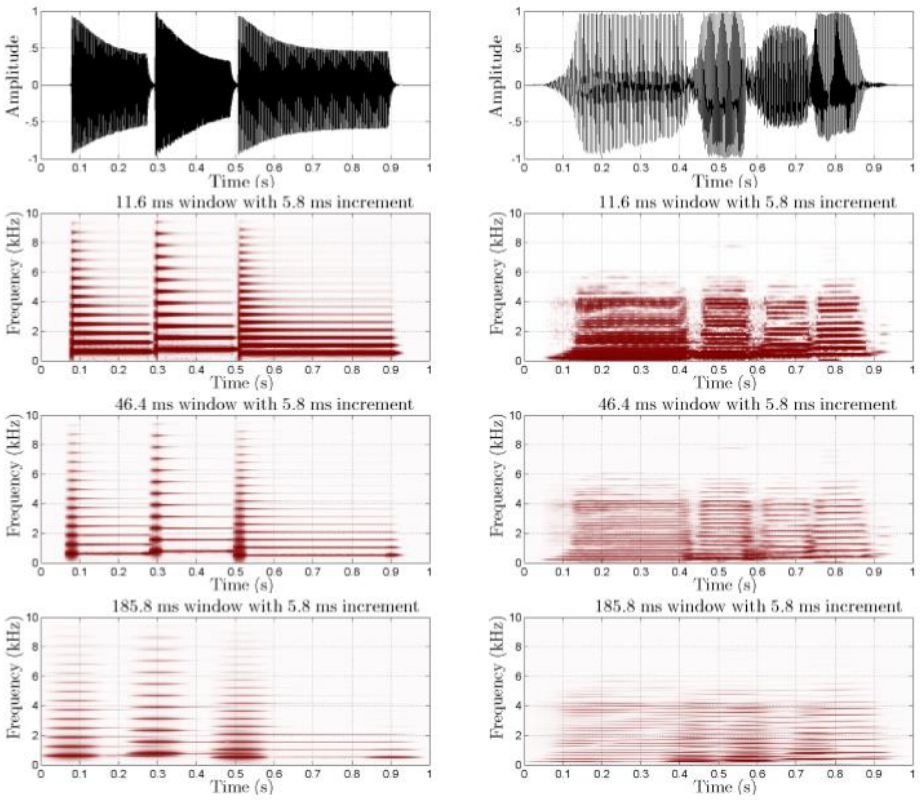

Figura 5 - Espetrogramas de dois sinais de áudio, com a variação das janelas de tempo. Fonte: Marques, 2014

Coeficientes espetrais de Mel

Os MFCCs são calculados janela a janela e estão relacionados com o espetro do sinal de áudio. Uma vez que sons com espetros semelhantes são geralmente percebidos como semelhantes. Os coeficientes são obtidos filtrando o espetro de potência de cada janela por uma série de filtros passa-banda que simulam a resposta auditiva da audição humana. Estes filtros são posicionados em intervalos regulares na escala Mel, que correspondem a intervalos logarítmicos na resolução de frequência linear. O primeiro filtro é o mais estreito e dá uma indicação de quanta energia existe perto de 0 Hz, à medida que as frequências aumentam a largura de banda também aumenta, Figura 6. Os seres humanos são melhores a discernir pequenas mudanças no tom em baixas frequências do que em altas, logo incorporar essa escala faz com que os recursos correspondam mais perto com aquilo que os humanos ouvem. O mapeamento entre frequências lineares nas frequências Hertz e Mel é dado por:

$$f_{Mel} = 2595 \log_{10}\left(\frac{f_{Hz}}{700} + 1\right) \tag{2.8}$$

A potência em cada banda de frequência (filtro) é:

$$X'[m] = 20 \log_{10}\left|\sum_{k} X[k]H[k,m]\right| \qquad m \,\epsilon\, \{1, 2, \dots, M\} \tag{2.9}$$

Onde M é o número total de bandas e $H[k,m]$ é o filtro, m, e k é o índex da frequência. Normalmente filtros triangulares são os mais usados, mas também podem ser empregues outras formas (por exemplo Retangular, Hanning, Gaussian). Finalmente, o logaritmo Mel é descorrelacionado através da transformada discreta de cosseno (DCT):

$$\phi[l] = \sum_{m=1}^{M} X'[m] \cos\left(\frac{\pi}{M}\left(m - \frac{1}{2}\right)\right) \tag{2.10}$$

Para l = 1, …, L, onde L $\leq M$ é o número de coeficientes após a DCT. Onde $\phi[l]$ são os coeficientes espetrais Mel. Os MFCCs estão associados ao conceito de timbre e são as características de escolha em sistemas baseados em áudio.

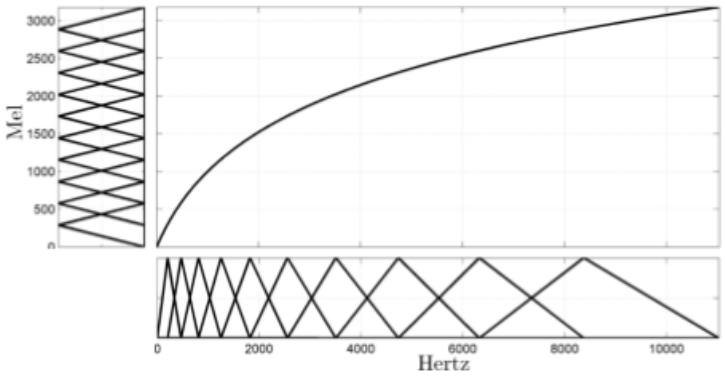

Figura 6 - Gráfico esquemático de filtros triangulares espaçados não uniformes no espetro de potência, em bandas de frequência Mel. Fonte: Marques, 2014

No entanto, também é comum estimar as diferenças de primeira e segunda ordem que os MFCCs possuem entre duas janelas consecutivas (MFCC-Δ e MFCC-Δ^2). Os MFCCs foram um dos tipos de características usadas nesta dissertação. Contudo, foi também testada a inclusão de diferentes números de coeficientes, mas a partir de vários testes, chegou-se à conclusão que usando 35 coeficientes chegava para atingir os melhores resultados no sistema implementado, verificando também que o uso de mais coeficientes não melhorava o desempenho dos métodos testados, nem a inclusão de MFCC-Δ ou MFCC-Δ^2.

Em síntese, com auxílio da Figura 7, podemos verificar o processo típico de extração destas características (este exemplo é feito usando 12 coeficientes).

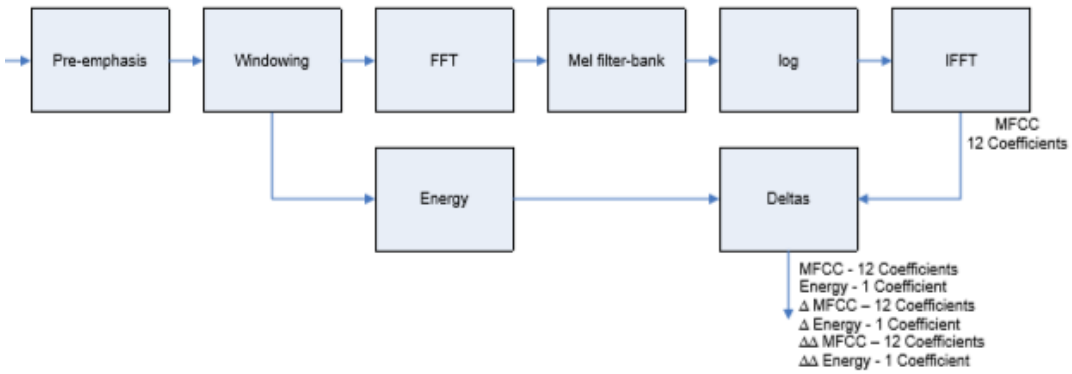

Figura 7 - Diagrama de blocos da extração das features MFCCs. Fonte: Burgos, 2014

Características espetrais

Além dos MFCCs, é comum incluir outros descritores extraídos do áudio. Há um grande número de características que podem ser usadas para fins específicos. Estas geralmente são agrupadas de acordo com alguma categorização, como energia, espetro, tempo, mas também outras taxonomias de características. Está além desta dissertação fornecer uma visão geral de todas as características que podem ser usadas em cenário de classificação baseada em áudio. Para descrever as características, primeiro é introduzida a energia normalizada do espetro de magnitude:

$$\tilde{X}[k] = \frac{|X[k]|}{\sum_{j \epsilon K} |X[j]|} \tag{2.11}$$

Onde $X[k]$ denota o espetro de Fourier discreto, k o índice de frequência, e $K = \{0, \dots, [\frac{N}{2}]\}$ o conjunto dos índices de frequência não negativos (para uma janela de áudio de N amostras).

De seguida, descreve-se brevemente os que foram usados. Estes são o centroide espetral, o *roll-off* e o *zero crossing rate*.

Centroide espetral

Esta característica indica onde o "centro de massa" do espetro está localizado. Está intimamente relacionada com o "brilho" dos timbres do som, que é derivado de uma analogia com o brilho visual. Os sons brilhantes estão associados a conteúdos de alta frequência e uma maneira de medir o brilho é através do centroide espetral, que é dada pela média ponderada das posições de frequência com as suas magnitudes:

$$SC = \sum_{k \epsilon K} k \tilde{X}[k] \tag{2.12}$$

Roll-of

Esta é a medida da quantidade de inclinação do espetro de potência. *Roll-off* é o ponto onde a frequência está abaixo de alguma percentagem (geralmente 95%, podendo também ser 85%) do espetro de energia. Isto é uma das maneiras de estimar a quantidade de alta frequência no sinal, consiste em encontrar a frequência de tal forma que uma certa fração da energia total seja contida abaixo dessa frequência.

$$SR = \lambda \sum |X[k]|^2 \tag{2.13}$$

Onde λ representa a percentagem, sendo que a usada nesta dissertação é de 85%.

Zero crossing rate

Esta medida, indica o número de vezes que o sinal cruza a amplitude zero, em qualquer das direções. Sendo assim, é calculado no domínio do tempo e embora não seja um recurso espetral, está intimamente relacionado com o conteúdo de alta frequência do sinal. Dando um exemplo, uma forma de onda quase periódica tende a ter valores baixos de *zero crossing rate*, enquanto sons não periódicos tendem a possuir valores altos. Para uma dada janela de N amostras de um sinal temporal $x[n]$, o *zero crossing rate* naquela janela é calculado por:

$$ZCR = \frac{1}{2N} \sum_{n=1}^{N-1} |sign(x[n]) - sign(x[n-1])| \tag{2.14}$$

Onde:

$$sign(x) = \begin{cases} +1 \; se \; x \; \geq 0 \\ -1 \; se \; x < 0 \end{cases} \tag{2.15}$$

2.3 Transformações lineares

Para ver como as características são representadas dimensionalmente, foram realizadas duas transformações lineares: análise de componentes principais, ou do inglês *Principal Component Analysis* (PCA), e análise discriminante linear, ou do inglês *Linear Discriminant Analysis* (LDA). Resumidamente, o LDA otimiza a matriz maximizando a distinção entre as classes e minimizando a variação dentro da classe, pois é um método supervisionado. Enquanto no PCA, como é um método não supervisionado, a matriz é obtida através das direções que possuem as maiores variações, pois nesta análise são desconhecidas as classes dos dados. Apesar de serem frequentemente usados para visualização de dados, também são usadas para pré-processar os dados, pois podem melhorar o desempenho do sistema. Nas subseções abaixo serão discutidas as metodologias usadas.

2.3.1 Análise de componentes principais

A análise de componentes principais é uma transformação linear ortogonal que projeta os dados nas direções de máxima variação – os componentes principais. A suposição subjacente é que as direções com maior variação transportam a maior parte da informação. Deste modo, os componentes principais formam um novo sistema de coordenadas, de maneira a que os dados projetados sejam linearmente descorrelacionados (isto é, tem uma matriz de covariância diagonal). A primeira componente principal possui a maior variação de dados, a segunda componente a segunda maior variação e assim por diante. PCA é uma técnica que pode ser obtida através da decomposição da matriz de covariância de dados, ou através da decomposição de valores singulares da matriz de dados. Formalizando, seja $P = [p_1, \dots, p_N]$ uma matriz dxN, contendo os dados: um total de N, d-vetores dimensionais. Por simplicidade, considera-se que os dados têm média zero. A matriz de covariância empírica $\hat{C}_p$ é:

$$\hat{C}_p = \frac{1}{N-1} XX^T = V \Delta V^T \tag{2.16}$$

Onde $V = [v_1, \dots, v_d]$ é uma matriz dxd que contém os vetores próprios, v_i de $\hat{C}_p$, e Δ é uma matriz diagonal com os seus valores próprios. De notar que matrizes simétricas possuem vetores próprios ortogonais: $VV^T = V^TV = VV^{-1} = I$, onde I é a matriz identidade.

O PCA é uma ferramenta essencial nos campos de análise exploratória de dados e previsão de modelos. Foi proposta pela primeira vez por Karl Pearson [Pearson, 1901] e mais tarde, independentemente, desenvolvido por Harold Hotelling [Hotelling, 1933] que o batizou com o seu nome. Portanto, o PCA ficou conhecido como transformação de Hotelling, mas dependendo do campo de aplicação, outros nomes foram usados, como transformação de Karhunen-Loéve no processamento de sinal, a decomposição ortogonal adequada em engenharia mecânica, etc.

2.3.2 Análise discriminante linear

A análise discriminante linear é uma projeção linear supervisionada. É uma generalização dos métodos discriminativos lineares Fisher [Fisher, 1936]. A sua finalidade é encontrar uma transformação que ao mesmo tempo maximize a separação de classes (dispersão entre classes) enquanto minimiza a variação dentro de cada classe (dispersão dentro da classe). Formalizando, seja P uma matriz dxN contendo os dados, onde cada vetor de dados tem um rótulo de classe conhecido e associado a ele: $p \in \gamma_i$ e com $\gamma_i \in \Gamma$, onde $|\Gamma|$ é o número total de classes. A matriz de dispersão dentro da classe, S_w é dada por:

$$S_w = \sum_{i=1}^{|\Gamma|} S_{\gamma_i} \tag{2.17}$$

Onde:

$$S_{y_i} = \sum_{p \in \gamma_i} (p - \mu_{\gamma_i})(p - \mu_{\gamma_i})^T \tag{2.18}$$

E:

$$\mu_{\gamma_i} = \frac{1}{N_i} \sum_{p \in \gamma_i} p \tag{2.19}$$

Onde N_i é o número de vetores na classe y_i. A matriz de dispersão entre classes, S_b é dada por:

$$S_b = \sum_{i=1}^{|\Gamma|} N_i(\mu_{\gamma_i} - \mu)(\mu_{\gamma_i} - \mu)^T \tag{2.20}$$

Onde:

$$\mu = \frac{1}{N}\sum_{\forall p} p = \frac{1}{N}\sum_{i=1}^{|\Gamma|} N_i \mu_{\gamma_i} \tag{2.21}$$

Com isto, pode-se definir os vetores projetados, y, em termos dos originais e uma matriz de projeção, W, de tamanho dx($|\Gamma| - 1$):

$$y = W^T p \tag{2.22}$$

As matrizes de dispersão ($\tilde{S}_w$ *e* $\tilde{S}_b$) dentro e entre classes do dados projetados são definidos por:

$$\tilde{S}_w = \sum_{i=1}^{|\Gamma|} \sum_{p \in \gamma_i} (y - \tilde{\mu}_{\gamma_i})(y - \tilde{\mu}_{\gamma_i})^T = W^T S_w W \tag{2.23}$$

Onde:

$$\tilde{\mu}_{\gamma_i} = \frac{1}{N_i}\sum_{y \in \gamma_i} y \tag{2.24}$$

$$\tilde{S}_b = \sum_{i=1}^{|\Gamma|} N_i(\tilde{\mu}_{\gamma_i} - \tilde{\mu})(\tilde{\mu}_{\gamma_i} - \tilde{\mu})^T = W^T S_b W \tag{2.25}$$

Onde:

$$\tilde{\mu} = \frac{1}{N}\sum_{\forall y} y \tag{2.25}$$

A matriz W é a projeção que maximiza a separação entre classes (a dispersão entre classes) e minimiza a variância em cada classe (a dispersão dentro da classe), que é obtida maximizando a seguinte função:

$$\mathcal{J}(W) = \frac{|\tilde{S}_b|}{|\tilde{S}_w|} = \frac{|W^T S_b W|}{|W^T S_w W|} \tag{2.26}$$

Capítulo 3

Classificação

Neste capítulo é dada uma breve explicação sobre os algoritmos de classificação utilizados, como também o desempenho e as vantagens/desvantagens de cada um.

Por último, o conjunto de dados será projetado para dois cenários de etiquetagem: o *multi-class* e o *multi-label*. Enquanto no cenário de *multi-class* cada evento de som apenas pode pertencer a uma das classes pré-definidas e mutuamente exclusivas, no *multi-label* cada evento pode ser identificado por mais do que uma classe (*tag*). Por conseguinte, faz com que este último seja uma configuração mais realista, pois um evento sonoro pode possuir várias classes a ocorrer em simultâneo.

3.1 Algoritmos de classificação

Com auxílio da redução da dimensionalidade efetuada e de um pré-processamento dos dados, o passo que se segue é recorrer a algoritmos de aprendizagem automática, com o objetivo de fazer predições.

Relativamente ao processo da classificação, foram usados métodos de aprendizagem supervisionada. Na aprendizagem supervisionada, os resultados são preditos a partir de probabilidades de acerto, que têm por base conjuntos de dados previamente treinados.

De seguida, serão abordados os algoritmos de classificação, explicando o seu funcionamento e discutindo a sua complexidade. Embora esteja além desta dissertação uma discussão mais aprofundada de cada algoritmo, tentar-se-á dar uma intuição sobre como cada algoritmo constrói o seu modelo, evidenciando os pontos fortes e fracos de cada algoritmo.

3.1.1 K-Nearest Neighbor

O algoritmo *k-Nearest Neighbors* (kNN) é possivelmente um dos mais simples algoritmos de aprendizagem automática. Construir o modelo consiste apenas em armazenar o conjunto de dados de treino. Para fazer uma previsão para um novo ponto de dados, o algoritmo encontra os pontos mais próximos do conjunto de treino, os "vizinhos mais próximos". Na sua versão mais trivial, o algoritmo considera apenas exatamente um vizinho mais próximo, a Figura 8 ilustra essa situação.

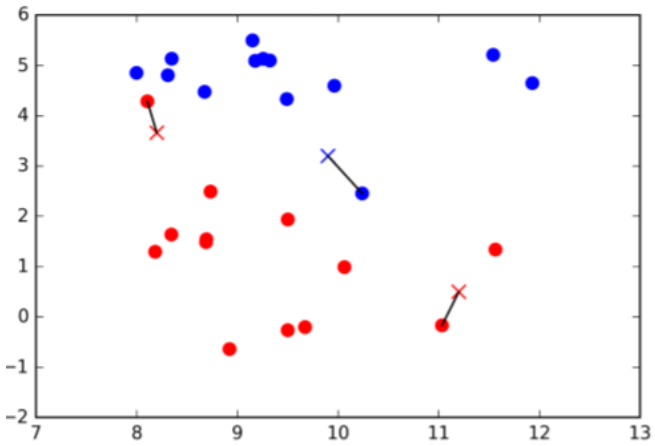

Figura 8 - Classificação KNN com apenas um vizinho. Fonte: Muller and Guido, 2016

Neste exemplo, Figura 8, adicionaram-se três novos pontos de dados, assinalados com cruzes, em que para cada um deles foi marcado o ponto mais próximo do conjunto de treino. A previsão deste algoritmo é a etiqueta desse ponto (mostrado pela cor da cruz). Em vez de se considerar apenas um vizinho mais próximo, também se pode escolher um número arbitrário de vizinhos, sendo esta a origem do seu nome.

Ao ser considerado mais de um vizinho, é usada uma espécie de votação para atribuir uma classe, isto significa que para cada ponto de teste é contado quantos vizinhos são vermelhos e quantos são azuis (isto para o exemplo acima retratado) e em seguida a classe com maior número de vizinhos dita o seu rótulo. De seguida, para o mesmo conjunto de dados, são agora usados três vizinhos em vez de um, Figura 9. Nesta situação podemos verificar que a previsão foi alterada para o ponto que se situa no canto superior esquerdo. Desta forma, conclui-se que ao aumentar o número de vizinhos nem sempre melhora a previsão deste classificador.

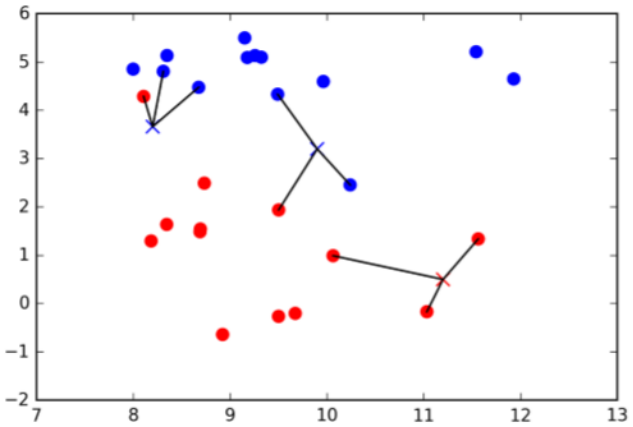

Figura 9 - Classificação KNN com o recurso a 3 vizinhos. Fonte: Muller and Guido, 2016

Em síntese, o parâmetro mais importante para o classificador kNN é o número de vizinhos que são escolhidos. Na prática, usar um pequeno número de vizinhos geralmente funciona bem, mas é um parâmetro que tem de ser ajustado. Contudo, também podem ser usadas diferentes métricas de distância sobre os pontos de dados, de forma a aperfeiçoar a previsão. Relativamente às métricas de distância, a utilizada foi a distância euclidiana, equação 3.1, pois foi aquela onde se obtiveram os melhores resultados. Contudo, também foram testadas as distâncias de *manhattan*, equação 3.2, e *minkowski*, equação 3.3.

$$D(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} \tag{3.1}$$

$$D(p,q) = |p_1 - q_1| + |p_2 - q_2| + \cdots + |p_n - q_n| \tag{3.2}$$

$$D(p,q) = (|p_1 - q_1|^q + |p_2 - q_2|^q + \cdots + |p_n - q_n|^q)^{\frac{1}{q}} \tag{3.3}$$

Onde $P = (p_1, p_2, \ldots, p_n)$ e $Q = (q_1, q_2, \ldots, q_n)$ representam os pontos, nos quais se pretende calcular a distância. O n é o tamanho do espaço dimensional das características. Todavia, pode-se verificar que a distância de *minkowski* é a generalização das outras duas distâncias, pois quando $q = 1$ representa a distância de *manhattan* e quando $q = 2$ a distância euclidiana.

O kNN apesar de ser um modelo muito fácil de compreender proporciona um desempenho razoável sem muitos ajustes. Ou seja, é um bom modelo de referência para ser testado antes de se considerar técnicas mais avançadas, servindo de *baseline*.

Contudo, não apresenta um bom desempenho para conjuntos de dados muito assimétricos, sendo importante realizar um pré-processamento. Portanto, embora seja um algoritmo fácil de realizar não é usado com muita frequência na prática, devido à sua lenta previsão e à incapacidade de lidar com muitos dados.

3.1.2 Random Forest

O algoritmo *random forest* é um método que opera através da construção de múltiplas árvores de decisão [Mueller and Guido, 2016], durante a fase de treino. Uma árvore de decisão é uma estrutura parecida a um fluxograma, a qual possui uma raiz e vários nós. Os nós representam as características do sinal de áudio, sendo que em cada um deles é aplicado um *if-else* (instrução condicional que executa um conjunto diferente de instruções, dependendo se uma expressão é verdadeira ou falsa), fazendo com que cada ramificação retrate o resultado do "teste". A Figura 10 retrata um exemplo simples duma árvore de decisão, em que neste caso o que se pretende saber é se um aluno passou numa disciplina. Quanto mais profunda a árvore, mais complexa é a decisão e mais apto o modelo.

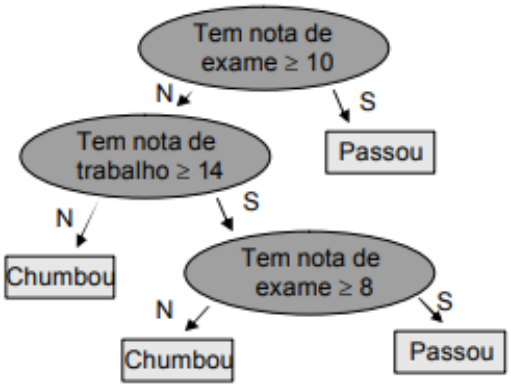

Figura 10 - Exemplo duma árvore de decisão, em que o objetivo é saber se o aluno passou numa disciplina. Fonte: V. Lobol, 2010

Apesar das árvores de decisão serem de fácil visualização e interpretação, apresentam a grande desvantagem de não conseguirem criar um modelo genérico, isto é, tendem para a sobre-aprendizagem. A sobre-aprendizagem é um termo usado em estatística para descrever quando um modelo se ajusta bem a um conjunto de dados observado, mas não mostra capacidade de generalização para outros conjuntos de dados, fazendo com que apresente bons resultados no conjunto de treino, porém não é uma boa representação da realidade.

Esta limitação é superada pelo *random forest*, pois apesar de possuir um conjunto de árvores de decisão para classificar o som recebido, também possui aleatoriedade em cada uma das árvores, o que permite que sejam diferentes umas das outras e assim se possa obter uma predição mais correta. A título de exemplo, é preferível que um excerto de som seja classificado por um grupo de pessoas do que apenas por uma, pois esta pode induzir em erro. De seguida, falar-se-á do seu funcionamento.

Para construir o modelo *random forest*, é necessário decidir sobre o número de árvores a serem criadas, qual o tamanho dos conjuntos aleatórios de características e se é usado *bootstrap*. O *bootstrap* é uma técnica que envolve amostragem aleatória de um conjunto de dados com substituição (a mesma amostra pode ser usada várias vezes), havendo por isso a criação de "amostras fantasmas" conhecidas como amostras de *bootstrap*. Ao utilizar esta técnica, o conjunto de características usado em cada árvore de decisão é diferente do inicial, isto porque algumas características podem ser repetidas enquanto outras omitidas. Desta forma, reduzir a variação de características ajuda a evitar a sobre-aprendizagem.

No que se refere ao número de árvores criadas, quanto maior o seu número mais robusto será o algoritmo, em contrapartida mais vagaroso será o seu desempenho. Desta maneira, é necessário haver um equilíbrio.

Outro parâmetro a ter em conta é o tamanho do conjunto aleatório de características a ser selecionado. Se o seu valor for baixo, faz com que as árvores sejam bastantes diferentes umas das outras e assim ajustarem melhor a predição, em contrapartida se for elevado há o risco de serem muito semelhantes. No caso da classificação, o tamanho padrão é igual à raiz quadrada do número de características dos dados.

Para fazer uma previsão, usando *random forest*, é utilizada uma estratégia de "votação", em que cada árvore fornece uma probabilidade para cada classe e depois é predita aquela que apresentar maior probabilidade.

Estes algoritmos, geralmente funcionam bem com conjuntos de dados grandes, não sendo preciso o ajuste de parâmetros e escalonamento de dados. O treino pode ser facilmente paralelizável com o auxílio dos núcleos de CPU. No entanto, exigem mais memória, podendo ser lentos a treinar e para conjuntos de dados demasiadamente grandes podem não obter o melhor desempenho.

3.1.3 Support Vector Machines

A SVM permite a construção de modelos mais complexos, sendo baseados no conceito de planos de decisão para definir os limites entre classes. Um plano de decisão é aquele que separa um conjunto de dados em duas classes.

Os classificadores mais simples deste algoritmo são os lineares, onde os conjuntos de dados são separados através de uma linha, isto é, possuem fronteiras de decisão lineares. Contudo, estes classificadores são limitados por espaços dimensionais baixos e muitas vezes são necessárias estruturas mais complexas para realizar uma separação quase ideal entre classes. A título exemplificativo, a Figura 11 mostra um exemplo em que a classificação linear não apresenta bons resultados.

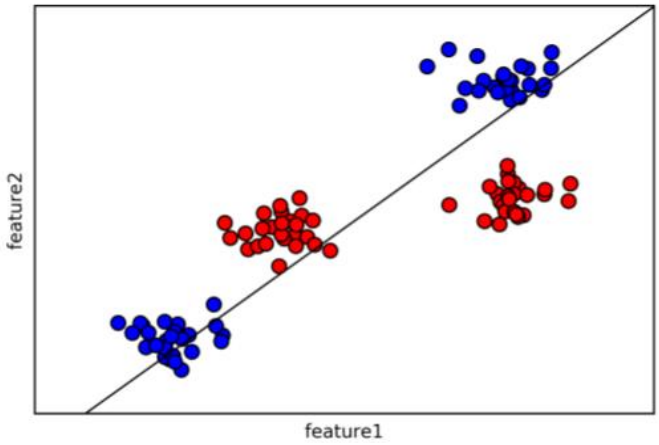

Figura 11 - Classificação SVM com modelo linear para classificação.
Fonte: Muller and Guido, 2016

Por conseguinte, ao realizar o quadrado da primeira característica, expandindo o conjunto de características, deixamos de representar cada ponto de dados como um ponto bidimensional, mas tridimensional. Assim sendo, com o aumento do espaço dimensional torna-se possível separar os pontos vermelhos e azuis, Figura 12, deixando de ser uma linha, para se assemelhar a uma elipse.

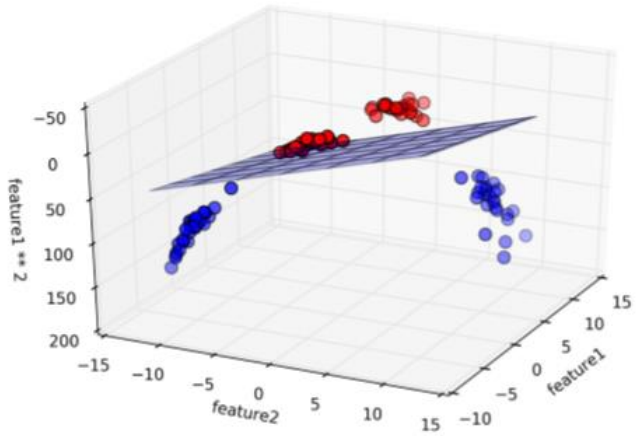

Figura 12 - Classificação SVM a nível tridimensional. Fonte: Muller and Guido, 2016

Assim verificamos que ao adicionar características não lineares à representação dos nossos dados, os modelos tornam-se mais eficientes. No entanto, muitas vezes não se sabe quantas características adicionar e ao adicionar muitas pode tornar o cálculo muito demorado. Contudo existe o chamado *kernel trick*, que tem como objetivo calcular diretamente as distâncias (mais precisamente, os produtos internos) dos pontos de dados para uma representação expandida das características, permitindo aprender um classificador num espaço dimensional superior sem realmente estimar a sua representação.

Nos últimos anos, os métodos *kernel* têm recebido uma grande atenção, particularmente devido à crescente popularidade do SVM. As funções do *kernel* podem ser usadas em muitas aplicações, pois fornecem uma ponte de ligação entre a linearidade e a não-linearidade [Zhang, 2007, Souza, 2010]. De seguida, serão apresentadas algumas delas: o *kernel* linear, caracteriza-se como a função mais simples e é dada pelo produto interno mais uma constante opcional (v), equação 3.4; o *kernel* polinomial, calcula todos os polinómios possíveis até um certo grau, adequado para dados de treino normalizados. Possui o parâmetro ajustável gama (γ) e o grau do polinómio (d), equação 3.5; o *kernel* da função de base radial (RBF) também conhecido como *kernel* gaussiano, considera todos os possíveis polinômios de todos os graus, na qual a importância das características diminui para graus mais altos, equação 3.6; o *kernel* sigmoid, também conhecido como o *kernel* da tangente hiperbólica, vem do classificador redes neuronais, abordado na próxima subsecção, equação 3.7.

$$K(x, z) = < x, z > + v \tag{3.4}$$

$$K(x, y) = (\gamma < x, z > +v)^d \quad (3.5)$$

$$K(x, z) = \exp(-\gamma||x - z||^2) \quad (3.6)$$

$$K(x, z) = \tanh(\gamma < x, z > +v) \quad (3.7)$$

O objetivo principal deste método é maximizar a margem de separação entre classes. Normalmente, apenas um subconjunto de pontos de treino é suficiente para definir esse limite, principalmente aqueles que estão na fronteira entre classes. Estes são chamados de vetores de suporte e dão por isso o nome a este algoritmo de *support vector machine.*

Todavia, existem dois parâmetros que são ajustados de forma a delimitar da melhor maneira a decisão de fronteira, sendo eles o gama e um parâmetro que controla o termo de regularização, representado neste dissertação como C. A Figura 13, exibe o que sucede com a variação destes parâmetros.

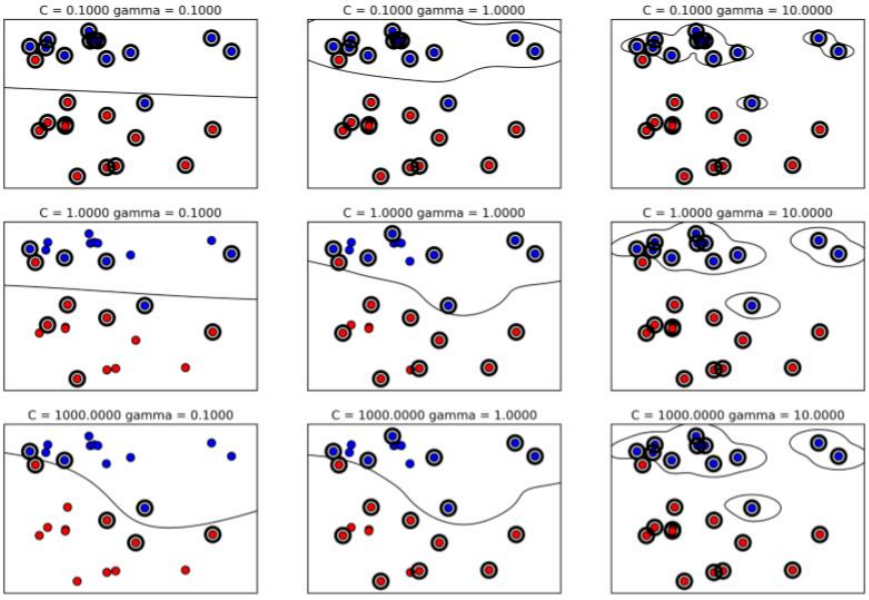

Figura 13 - Classificação SVM com variação dos parâmetros C e gama. Fonte: Muller and Guido, 2016

Analisando a Figura 13 e indo da esquerda para a direita, aumenta-se o parâmetro gama de 0.1 para 10. Um pequeno gama, significa um grande raio para o *kernel*, o que faz com que muitos pontos sejam considerados próximos. Isto reflete-se com limites de decisão muito suaves à esquerda e limites mais concentrados à direita. Por sua vez, um alto valor de

gama significa que os limites de decisão são mais restritos, o que produz modelos mais complexos.

Examinando de cima para baixo, aumenta-se o parâmetro C de 0.1 para 1000. Um parâmetro C pequeno significa um modelo mais restrito, em que cada ponto de dados tem uma influência muito limitada, enquanto com o aumento do C, é permitido que certos pontos tenham uma influência mais forte no modelo, fazendo com que o limite de decisão se curve para evitar a sobre-aprendizagem.

Em síntese, SVM são modelos muito poderosos e que funcionam bem em muitas variedades de conjunto de dados, permitindo limites de decisão muito complexos, mesmo se os dados possuírem apenas algumas características. Como limitações exigem um pré-processamento dos dados antes do treino e um cuidado ajuste dos parâmetros, propriamente o C e o gama.

3.1.4 Redes Neuronais

Uma rede neuronal artificial é um modelo computacional que se inspira na forma como as redes neuronais biológicas processam as informações. Estas redes são compostas por camadas de unidades computacionais interconectadas chamadas perceptrão [Vojt, 2016], que são algoritmos que colecionam e classificam informações de acordo com uma determinada arquitetura, transformando os dados até que possam ser preditos como uma saída. Cada perceptrão multiplica um valor inicial por um peso, depois soma os resultados com outros valores que entram no mesmo perceptrão e de seguida são introduzidas componentes de não linearidade com auxílio das funções de ativação.

Um perceptrão de múltiplas camadas (MLP) contém uma ou mais camadas escondidas (além da entrada e da saída). Enquanto um perceptrão de uma camada única só pode aprender funções lineares, MLP pode aprender funções não lineares. A Figura 14 ilustra um exemplo dessa rede. A rede MPL é constituída por três tipos de camadas: a entrada, que fornece informações do exterior para a rede, em que contém as características do sinal; as escondidas, que executam cálculos e “transferem” a informação da entrada para a saída; a saída, que utilizam somas ponderadas para produzir o resultado final.

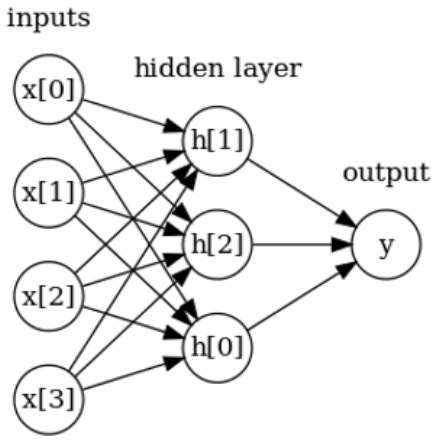

Figura 14 - Classificação redes neuronais, exemplo de uma rede. Fonte: Muller and Guido, 2016

Posto isto, a saída de cada camada escondida é obtida através da aplicação de funções de ativação, ou às vezes chamadas de funções de transferência, sobre a soma ponderada de todas as características de entrada, sabendo que cada camada escondida no processo de treino define os pesos, para cada entrada, de maneira a melhorar o desempenho. As funções de ativação introduzem uma componente não linear aplicada ao resultado. Nesta dissertação as duas funções utilizadas foram a não-linearidade retificadora (também conhecida como unidade linear retificada ou relu) e a tangente hiperbólica (tanh). A Figura 15 evidencia o comportamento de ambas as funções, verificando-se que a relu corta valores abaixo de zero, enquanto a tanh satura para valores abaixo de -1 e acima de 1.

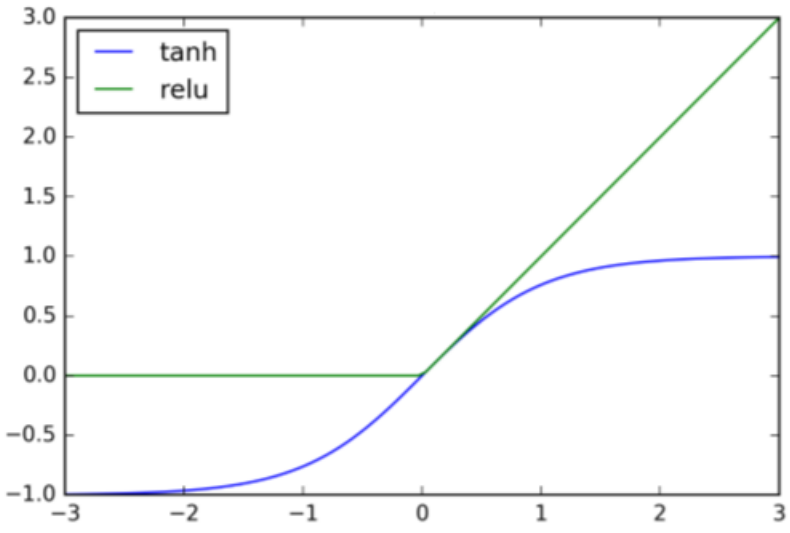

Figura 15 - Classificação redes neuronais, comportamento das funções de ativação, relu e tanh. Fonte: Muller and Guido, 2016

Desta maneira, há muitas alternativas para controlar a complexidade de uma rede neuronal, como o número de camadas escondidas, o número de unidades em cada camada escondida, a função de ativação, etc. Uma propriedade importante nas redes neuronais é que os pesos são definidos aleatoriamente antes do início da aprendizagem, e esta inicialização aleatória afeta o modelo que é aprendido. Isto significa que mesmo usando exatamente os mesmo parâmetros, podemos obter modelos muito diferentes ao usar diferentes inicializações.

As redes neuronais são vantajosas, pois apesar de serem capazes de adquirir informações contidas em grandes quantidades de dados e construir modelos muito complexos, possuem capacidade de processamento paralelo. Como inconvenientes, exigem um pré-processamento dos dados, pois funcionam melhor com dados "homogêneos", requerem um ajuste personalizado dos seus parâmetros de forma a atingir os melhores resultados e o treino pode demorar tempo dum ponto de vista computacional.

3.2 Etiquetação automática

As etiquetas (*tags*) são anotações textuais que formam uma descrição semântica que pode ser usada, por exemplo, para marcar aquilo que sentimos/ouvimos.

Idealmente, os sons devem ser etiquetados por especialistas que usam um vocabulário bem definido (taxonomia) e que, para cada som, confirme se todas as *tags* são aplicáveis ou não. Este não é o caso da maioria dos conjuntos de dados disponíveis publicamente. Os que estão disponíveis podem possuir muitos problemas, especialmente quando as *tags* são recolhidas de forma não estruturada, fazendo com que se crie uma má etiquetação, pois apesar de certos sons não apresentarem uma determinada *tag*, não significa necessariamente que esta não esteja presente.

Desta forma, o primeiro cenário tido em conta foi o de *multi-class*, em que cada exemplo pertence a uma das classes pré-definidas e mutuamente exclusivas, isto é, cada *tag* não pode pertencer a mais de uma classe, fazendo com que cada trecho de som apenas apresente uma *tag*. Para avaliar o seu desempenho, é necessário saber qual a probabilidade total de errar (ou acertar) independentemente das classes, sendo que neste caso é recorrente utilizar a matriz de confusão para analisar o seu comportamento.

Contudo, os sons urbanos são dinâmicos e podem ser compostos por vários eventos em simultâneo, e o cenário de *multi-class* fica um pouco restritivo para esta gama dinâmica, assim sendo este cenário pode não ser a melhor estratégia para lidar com a tarefa de

classificação sonora urbana. Idealmente, o sistema tem como objetivo ser capaz de classificar eventos de áudio a ocorrer simultaneamente. Portanto, uma configuração mais realista é a etiquetagem de várias *tags*, num trecho de áudio. Este conceito é conhecido como *multi-label*. O seu problema é geralmente dividido num conjunto de problemas de classificação binária, um para cada *tag*, fazendo com que a sua avaliação seja geralmente baseada em métricas derivadas do problema da classificação binária. De observar que em muitos problemas de extração de informações no som, particularmente no *multi-label*, são utilizadas várias métricas de desempenho que refletem diferentes especificidades do desempenho do classificador. A escolha de quais as métricas que devem ser usadas, dependem da área de aplicação e da importância dos dois tipos de erros possíveis.

Para ajudar a consolidar o problema da classificação binária, a Figura 16 mostra a matriz de confusão e as equações de várias métricas geralmente usadas e que podem ser calculadas. Em linhas gerais, é habitual referir duas classes neste tipo de classificação, como os positivos e os negativos. Tipicamente a classe dos positivos representa a existência ou deteção de uma dada condição, situação, teste, etc. Em muitos casos práticos, o número de exemplos positivos são significativamente menor que os negativos, o que pode não representar uma boa medida de desempenho. Todavia, mais abaixo será dada uma abordagem mais pormenorizada sobre as classes e as métricas habitualmente utilizadas.

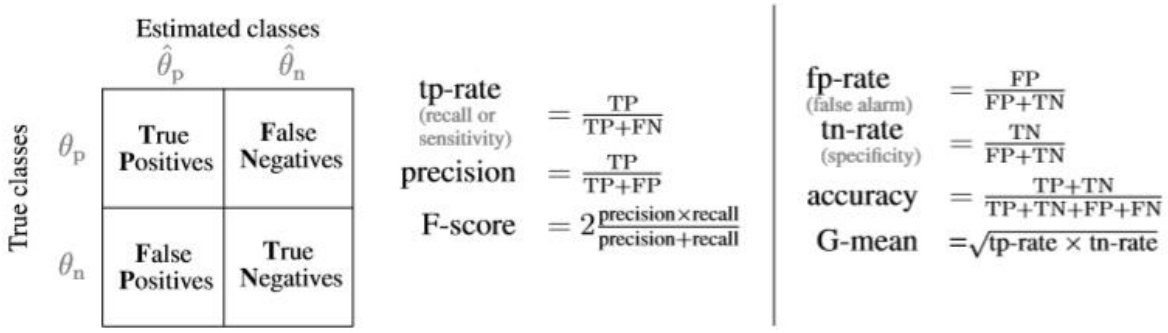

Figura 16 - Matriz de confusão e equações de várias métricas comummente usadas. Fonte: Marques, 2014

Através da figura acima, dadas duas classes, $\theta = \{\theta_p, \theta_n\}$, e as predições classificadoras, $\widehat{\theta} = \{\widehat{\theta}_p, \widehat{\theta}_n\}$, há quatro resultados possíveis representados na matriz de confusão: Verdadeiros Positivos (TP), Falsos Negativos (FN), Falsos Positivos (FP) e Verdadeiros Negativos (TN). Na diagonal desta matriz estão as decisões corretas, enquanto os outros elementos, FP e FN, representam os erros de classificação. No *multi-label*, e para cada som urbano, a classe positiva θ_p indica a presença da tag θ, enquanto o $\widehat{\theta}_p$ e $\widehat{\theta}_n$ são as predições positivas e negativas do classificador, respetivamente. As medidas mais comuns

são a *precision*, *recall* e o F-score, embora alguns autores também utilizem a G-mean [Seyerlehner, 2010].

A *precision* mede quantos sons urbanos previstos como positivos são realmente positivos e é usada como uma métrica de desempenho quando o objetivo é limitar o número de FP. Como exemplo, se imaginarmos um modelo para prever se um novo medicamento é eficaz no tratamento de uma doença em ensaios clínicos, em que notoriamente são caros, só é desejável realizar experiências em pacientes que tenham essa doença. Portanto, é importante que o modelo não produza muitos FP na seleção dos pacientes.

O *recall,* por outro lado, mede quantos sons urbanos positivos são capturados pelas previsões positivas e é usado como métrica de desempenho quando se precisa de identificar todos os sons positivos, ou seja, quando é importante evitar FN. A título de exemplo, num diagnóstico de cancro é melhor apresentar mais pessoas doentes, incluindo possíveis pacientes saudáveis na previsão, do que apresentar poucos doentes e correr o risco de pessoas que tenham cancro serem consideradas saudáveis.

O F-score é a média da *precision* e do *recall*. É comum avaliar com base apenas em F-scores, mas não é aconselhável apresentar medidas isoladas de *precision* ou *recall*. A razão é porque os valores de *precision* podem ser artificialmente alinhados com esquemas de atribuição de *tags* “conservadoras”: classificadores que atribuem uma classe positiva apenas com fortes evidências (e cometem poucos erros FP) têm tipicamente pontuações altas de *precision* e baixa *recall*. Por sua vez, o *recall* pode ser combinado escolhendo classificadores “liberais”: para o caso de um classificador positivo trivial (isto é, atribui *tag* a todos os sons), faz com que o *recall* seja igual a um. Enquanto modelos triviais podem alcançar resultados relativamente bons, em uma destas duas medidas, apenas um modelo verdadeiramente válido pode obter simultaneamente valores altos de *precison* e *recall*. Porém existe outra ferramenta geralmente usada para analisar o comportamento dos classificadores em diferentes limites, chamada curva de características operacionais, ou do inglês *Receiver Operating Characteristics* (ROC). A curva ROC considera de maneira similar todos os possíveis limiares para um determinado classificador, mas em vez de retornar *precision* e *recall*, mostra a taxa de FP em relação à taxa TP. Sabendo que a taxa TP é simplesmente outro nome para *recall*, enquanto a taxa FP é a fração de FP de todas as amostras negativas, a Figura 17 ilustra uma curva ROC.

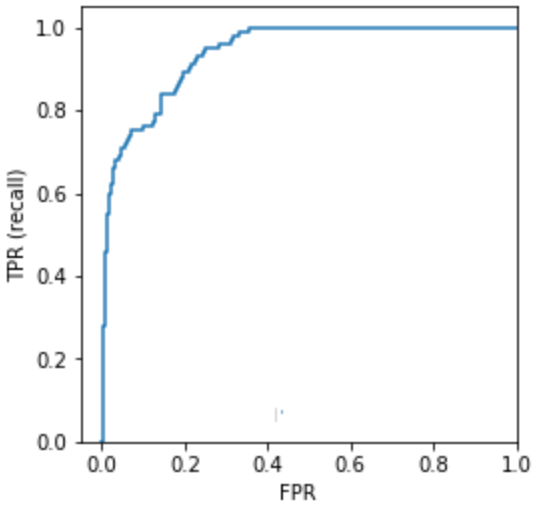

Figura 17 - Curva ROC

Neste caso, a curva ROC ideal é quando está sobre o canto superior esquerdo, pois o pretendido é possuir um classificador que apresente um alto *recall*, enquanto mantém uma taxa de FP baixa. Em adição da curva ROC, ainda é usualmente utilizada a área debaixo da curva, do inglês *Area Under the Curve* (AUC), para comparar classificadores. A AUC pode ser interpretada como a capacidade de discriminar corretamente as observações positivas das negativas, podendo variar de 0 a 1, portanto uma AUC de 1 significa que todos os pontos positivos têm uma pontuação maior do que todos os pontos negativos, condição perfeita.

Os resultados obtidos sobre *multi-class* e *multi-label* serão mencionados no Capítulo 5.

Capítulo 4

Sistema proposto

Este capítulo tem como objetivo analisar a implementação usada na criação dum sistema para classificação e identificação de eventos sonoros em ambiente urbano. Aqui é apresentado o sistema adotado, fornecendo os procedimentos e abordagens que foram aplicados.

4.1 Esquemático

O ambiente urbano apresenta sons compostos por sobreposições de eventos, que aparecem e desaparecem em determinados períodos, desta forma o foco deste sistema é classificar esses eventos sonoros ocorrendo em qualquer instância das gravações. Para este feito, a proposta é composta por cinco blocos principais de processamento: pré-processamento, extração de características, *pooling*, normalização de características e classificação. A ideia chave é "aprender" com os dados de treino, de uma maneira supervisionada.

O esquemático da representação de todo o processo é ilustrado na Figura 18. Nas subseções seguintes, descrever-se-á cada bloco em detalhe. Toda a implementação foi realizada, recorrendo à linguagem de programação *Python*.

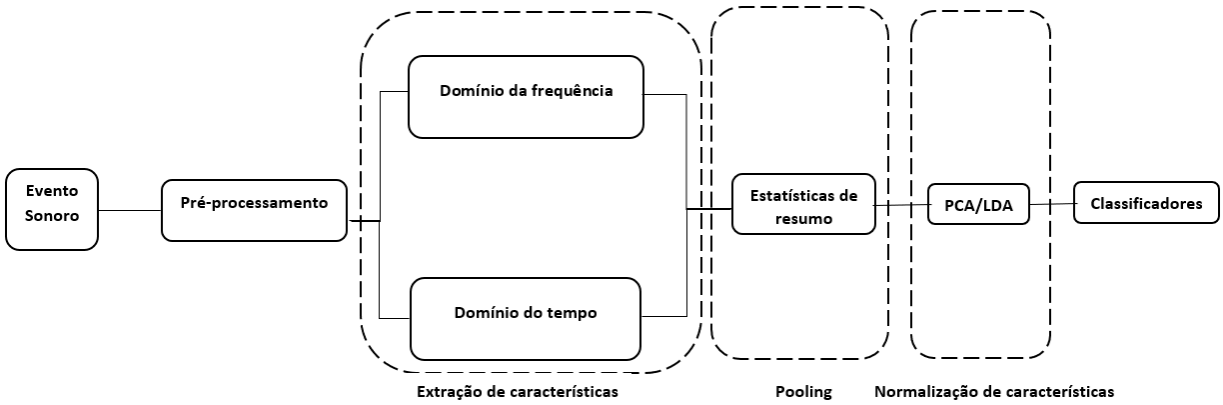

Figura 18 - Diagramas de blocos do sistema utilizado

4.1.1 Pré-processamento

Primeiramente os sinais de áudio têm de ser pré-processados, visto que após de serem digitalizados podem possuir variações de gravação não uniformes, isto é, variações na quantidade de canais de áudio recebidos e na frequência de amostragem. De maneira a maximizar o desempenho da análise de áudio e assim mitigar estas variações, todos os sinais de áudio foram convertidos num formato uniforme (convertendo neste caso para canal modo) e utilizada a técnica da re-amostragem, para que todos os sinais possuam a mesma frequência, que neste sistema a utilizada é de 44100 Hz.

4.1.2 Extração de características

De seguida segue-se o bloco da extração de características, em que para algoritmos de classificação é expetável haver baixa variabilidade entre características extraídas para exemplos da mesma classe e ao mesmo tempo, alta variabilidade para classes distintas.

Para isso extraiu-se características de tempo-frequência, que são obtidas dividindo o sinal de tempo em segmentos curtos e sobrepostos, dos quais descritores espetrais são calculados, estes descritores são mencionados com mais pormenor na subseção 2.2.3.

Um dos aspetos importantes a analisar na extração das características, é o tamanho das janelas e saltos entre as mesmas, visto que ao ser tomado em consideração a evolução temporal do sinal é necessário saber que parâmetros fornecem o melhor desempenho para o sistema proposto. Desta forma, foi analisado o comportamento do sistema, com a variação do tamanho das janelas e saltos entre as mesmas. Para a verificação deste comportamento, foi usada a base de dados do UrbanSound8k, PCA com branqueamento e o classificador redes neuronais. A Tabela 2 mostra os resultados obtidos.

Tabela 2 - Probabilidades de acerto [%], para diferentes tamanhos de janelas e saltos entre janelas

Janela	*Salto*	Probabilidade de acerto [%]
512	*256*	63,7
1024	*512*	64,8
2048	*512*	67,8
2048	*1024*	68,3
4096	*2048*	67,9
4096	*1024*	67,5
8192	*2048*	65,8

Por forma a maximizar o sistema proposto, foram utilizadas janelas de 2048 e saltos entre as mesmas de 1024. À vista disto, o sinal fica assim dividido em segmentos de 46 milissegundos (2048 amostras a 44100 Hz) com 50% *overlap*. Após este bloco de processamento, cada sinal sonoro fica convertido numa sequência vetorial de 38 dimensões, sendo que foram usadas 40 bandas Mel para extrair 35 MFCCs, mais outras três características utilizadas, como o *roll-off*, o centroide espetral e o *zero crossing rate*.

Estas características foram calculadas através de funções da biblioteca librosa [McFee et al., 2015].

4.1.3 Pooling

Após obter a sequência vetorial de 38 dimensões, proveniente da extração de características, é necessário convertê-la num vetor único de características, isto porque os algoritmos *standard* de classificação, utilizados neste sistema, apenas lidam com vetores únicos e não com sequências vetoriais, Figura 19. Deste modo, a sequência vetorial obtida foi resumida ao longo do tempo usando estatísticas de resumo habitualmente usadas [Salamon et al., 2014]: mínimo, máximo, mediana, média, variância, desvio padrão e assimetria.

Este método é usualmente conhecido no MIR como *Bag-of-Frames* (BoF), um termo emprestado dos modelos "*bag of words*" da recuperação de texto. A razão para o nome vem do fato de que os métodos BoF desprezam a informação temporal do sinal de áudio. Apesar de ser um pouco contraditório a utilização deste método fase a separação do sinal de áudio por janelas, por forma a possuir a evolução temporal do sinal, recorreu-se ao método BoF

como primeira abordagem à área, análise computacional de cenas e eventos sonoros urbanos, por forma a torná-la mais simplista.

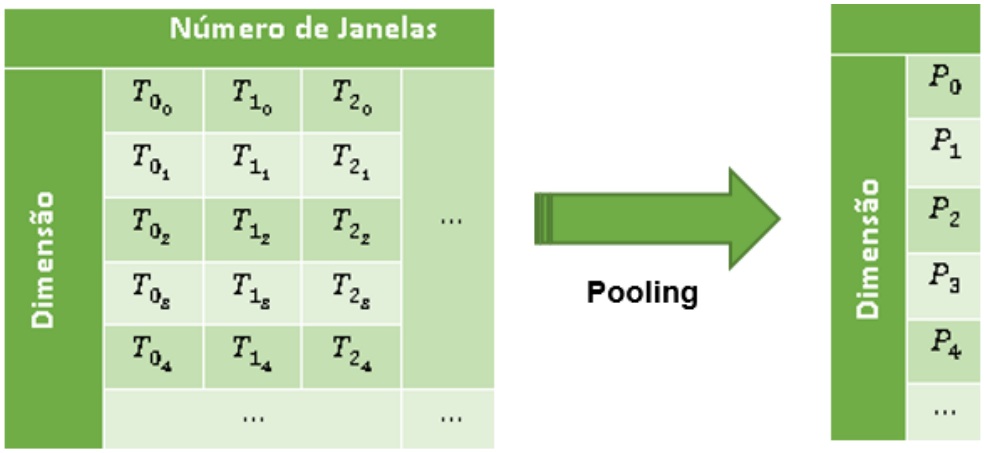

Figura 19 - Conversão da sequência vetorial de 38 dimensões para um vetor único de 266 dimensões, recorrendo a estatísticas de resumo

Deste modo, após a realização deste bloco de procedimento cada evento sonoro possui assim um vetor único de 266 dimensões, vindas da multiplicação das sete estatísticas de resumo pelas 38 dimensões da sequência vetorial.

4.1.4 Normalização de características

Embora pudéssemos utilizar diretamente os vetores únicos de características obtidos anteriormente, nos sistemas de classificação, comprovou-se que ao recorrer a transformações lineares havia melhorias nos resultados, o que é corroborado por outros investigadores [Coates and Ng, 2011, Ye et al., 2017]. Primeiramente, foi tirada a média de todos os dados e de seguida com base nas duas técnicas de transformação linear, aprofundadas nas subseções 2.3.1 e 2.3.2, testou-se quatro projeções e verificou-se qual delas proporcionava os melhores resultados. Para a realização de ambas as técnicas, recorreu-se à biblioteca scikit-learn [Pedregosa et al., 2011].

Desta forma as quatro projeções utilizadas foram o PCA com e sem branqueamento, o LDA e também a junção do PCA e LDA. O branqueamento realiza uma normalização de variância. Os resultados obtidos encontram-se no Capítulo 5.

4.1.5 Classificação

Ultimamente, para o processo da classificação foram utilizados quatro algoritmos mencionados na seção 3.1. O kNN é uma aprendizagem baseada em instâncias, em que a afiliação à classe é atribuída com base no voto da maioria dos seus vizinhos e é possivelmente um dos métodos de classificação mais simples, enquadrando-se num sistema *baseline*, pois é um bom ponto de partida para ser testado e comparado com os resultados existentes, antes de se considerar técnicas mais avançadas. O *random forest* foi utilizado, pois é um algoritmo muito robusto e poderoso, funcionando melhor que uma árvore de decisão, não precisando de escalonamento dos dados. O SVM foi escolhido pelo seu desempenho e pelas suas capacidades de generalização, particularmente em espaços de alta dimensão. Este classificador possui diferentes tipos de *kernel* a serem usados no algoritmo. Após vários testes sobre os mesmos, aquele que proporcionou melhores resultados foi o gaussiano e assim foi o escolhido para os testes que se seguem no Capítulo 5. Por último, as redes neuronais, que embora sejam muito sensíveis ao dimensionamento dos dados e à escolha dos parâmetros, conseguem construir modelos muito complexos, à vista disso tem sido um algoritmo usado por muitos investigadores [Salamon and Bello, 2016, Piczak, 2015, Ye et al., 2017, Cakir, 2014, Antich, 2017].

Para a realização dos resultados, com auxílio a estes algoritmos de classificação, usaram-se as implementações fornecidas na biblioteca *scikit-learn* [Pedregosa et al., 2011].

Capítulo 5

Avaliação

Este capítulo tem como propósito avaliar o sistema efetuado, para diferentes classificadores e conjuntos de dados, de forma a poder criar uma base de comparação com resultados alcançados por outros investigadores. Por fim, será demonstrada a dicotomia entre a classificação *multi-class* e *multi-label*.

5.1 Dados e metodologias de teste

Nos últimos anos, vários novos conjuntos de dados têm sido disponibilizados para classificação de eventos sonoros [Mesaros et al., 2017, Piczak, 2015, Salamon et al., 2014, Gemmeke et al., 2017]. Para avaliar a abordagem proposta, foram escolhidas duas bases de dados, a UrbanSound8k [Salamon et al., 2014] e a ESC-50 [Piczak, 2015].

O conjunto de dados UrbanSound8k inclui dez classes de sons urbanos com 8732 excertos de sons do mundo real, com duração máxima de 4 segundos. Uma breve taxonomia das classes, juntamente com o número de excertos de áudio por classe, é mostrada na Figura 19.

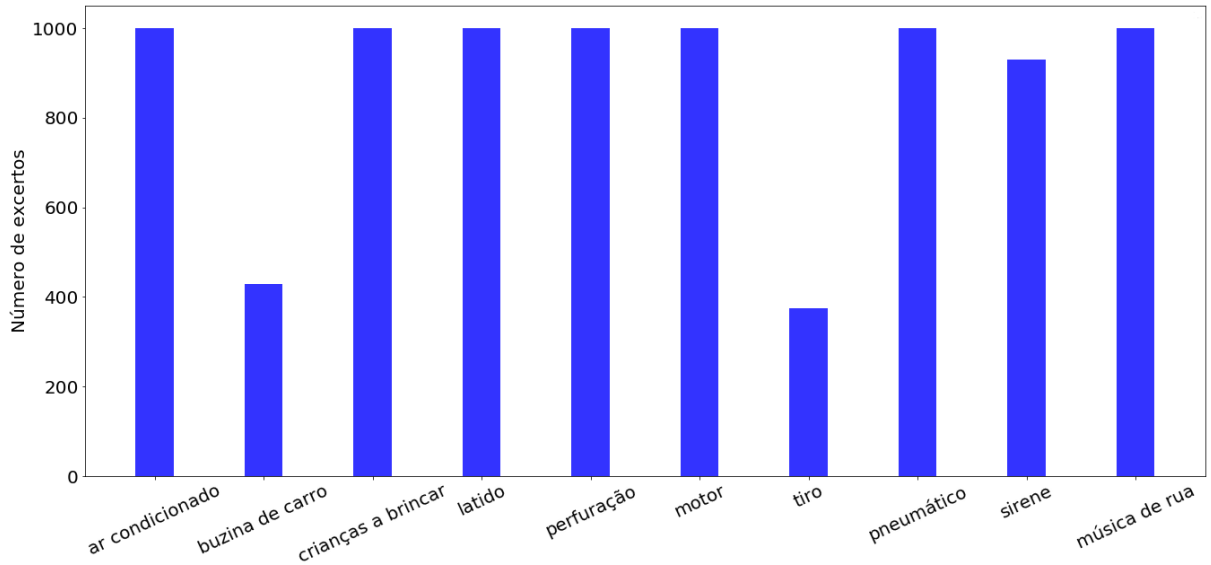

Figura 20 - Número de excertos de sons existentes em cada classe, para o conjunto de dados UrbanSound8k

Os excertos abrangem sons como: ar condicionado, buzina de carro, crianças a brincar, latido, perfuração, motor, tiro, pneumático, sirene e música de rua. A razão para escolher este conjunto de dados deve-se ao facto de englobarem sons relacionados à vida urbana e, portanto, adequados para testar algoritmos de classificação de som urbano. Os excertos de sons, nesta base de dados, são pré-agrupados em 10 *folds*, para assim ser possível comparar os resultados obtidos, com as abordagens publicadas sobre o mesmo conjunto de dados. Adicionalmente, este conjunto de dados ainda possui um parâmetro relativo à saliência do som, com o objetivo de informar se o som está em primeiro ou segundo plano.

Relativamente ao conjunto de dados ESC-50, este é uma coleção de 2000 excertos de sons, com duração máxima de 5 segundos, compreendendo 50 classes igualmente equilibradas de eventos sonoros, em que cada classe possui 40 excertos de áudio, divididos em 5 grupos principais: animais, paisagens sonoras naturais e sons de água, sons humanos não falados, sons domésticos e sons urbanos. Foi escolhido este conjunto de dados não só por incluir sons urbanos, mas também por possuir uma maior variedade de classes e assim ser possível verificar o comportamento do classificador proposto, quando se utiliza mais de 10 classes. Nesta base de dados, os excertos de sons foram pré-agrupados em 5 *folds*.

Desta forma, foi usada validação cruzada para avaliar a capacidade de generalização do modelo, a partir do conjunto de dados utilizados. No caso do UrbanSound8k, foram utilizados 10 *folds* para validação cruzada e no ESC-50 apenas 5 *folds*. O conceito central das técnicas de validação cruzada é a divisão do conjunto de dados em subconjuntos mutualmente exclusivos e, posteriormente, utilizar um dos *folds* para treino e os restantes para teste.

Deste modo, todos os valores relativos à probabilidade de acerto são obtidos pela média das probabilidades alcançadas, em cada *fold*.

5.2 Resultados: Multi-class

Primeiramente e antes de comparar os quatro classificadores mencionados na secção 2.3, realizaram-se duas transformações, habitualmente usadas para a visualização de dados: PCA e LDA, isto para ver como a diversidade espetral é refletida nas características. A Figura 20 mostra essas transformações em 2D, respetivamente. Na realização desta representação

foi utilizado o conjunto de dados do Urbansound8k, pois apresenta menos classes que o conjunto de dados ESC-50, o que possibilita melhor visualização dos dados.

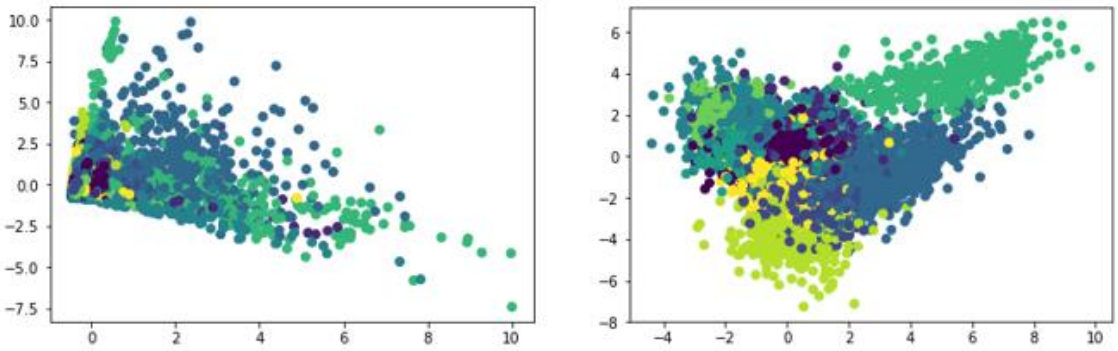

Figura 21 - Representação das transformações em 2D, recorrendo ao PCA (imagem da esquerda) e ao LDA (imagem da direita)

Ao analisar a Figura 20 verifica-se o que foi dito na secção 2.3, pois o PCA assume que as direções mais informativas são aquelas com maior variância e dessa forma projeta os dados originais dessa forma, enquanto que no LDA, se comprova que a transformação tenta maximizar a distância entre as classes e, ao menos tempo, minimizar a variação dentro de cada classe.

De maneira a aferir qual dos classificadores utilizados produz melhores resultados, foram realizados vários testes. Para cada classificador utilizaram-se diferentes transformações, mencionadas no subseção 4.1.4, em que a métrica de avaliação foi a probabilidade de acertos. Os testes foram realizados para as duas base de dados, UrbanSound8k e ESC-50. A Tabela 3 evidencia os resultados obtidos.

Tabela 3 - Análise das probabilidades de acerto [%] para os dois conjuntos de dados, UrbanSound8k e ESC-50, variando os classificadores e as projeções

Probabilidade de acerto [%]		**Classificadores**			
		kNN	RF	SVM	RN
UrbanSound8k	PCA c/b	55,5	63,0	66,8	68,3
	PCA s/b	46,7	61,7	12,2	46,8
	LDA	63,1	63,8	64,3	64,7
	PCA+LDA	60,2	61,3	63,1	64,3
ESC-50	PCA c/b	39,5	50,9	57,3	59,4
	PCA s/b	20,4	50,7	7,8	24,1
	LDA	46,2	51,8	53,3	54,8
	PCA+LDA	42,2	52,8	56,1	56,6

Ao analisar a tabela acima referida, verifica-se rapidamente uma queda abrupta quando se utiliza PCA sem branqueamento, tanto no classificador SVM, como nas redes neuronais, visto que dos quatro classificadores testados, estes são os mais sensíveis ao escalonamento dos dados. Em contrapartida, como o classificador *random forest* não é sensível ao escalonamento dos dados, quando se usa PCA sem branqueamento neste classificador, não existe grande diferença perante as outras 3 projeções testadas.

Outro pormenor que sobressai é relativamente ao uso do LDA versus PCA com branqueamento, pois em ambas as bases de dados, se verifica que, quando se utiliza os classificadores kNN e *random forest*, estes apresentam melhores resultados para o LDA, mas em contrapartida quando se recorre aos classificadores SVM e redes neuronais, os melhores resultados pertencem ao PCA com branqueamento. Uma forma de explicar esta situação, deve-se ao facto dos classificadores kNN e *random forest* ao serem pouco eficazes para dados com altas dimensões, beneficiam quando se usa LDA, pois este reduz a dimensão dos dados para o número das classes menos um. Por sua vez, os classificadores SVM e redes neuronais como se adaptam melhor a dados com maiores dimensões, ao utilizar PCA com branqueamento os resultados são superiores ao LDA. Relativamente ao PCA com branqueamento, foi testado o seu desempenho para diferentes números de componentes principais, chegando à conclusão que usando 45 componentes principais servia para atingir o melhor resultado.

A respeito das probabilidades de acerto nota-se que são mais elevadas para o conjunto de dados do UrbanSound8k do que para o ESC-50, isto deve-se ao número de classes que cada base de dados possui, pois como o ESC-50 apresenta 50 classes e o melhor resultado para o UrbanSound8k é 68,3% com apenas 10 classes, era de prever que os valores fossem mais baixos para a base de dados ESC-50. Contudo, apesar de não serem muito elevados, são valores bastantes interessantes, pois superam alguns classificadores *baseline* [Piczak, 2015, Huzaifah, 2017, Aytar et al., 2016], que obtiveram probabilidades de acerto de 44%, 51% e 52%, respetivamente.

Por último, apesar da junção do PCA com o LDA também possuir bons resultados, foi escolhido apenas usar o PCA com branqueamento para os testes que se seguem. De seguida, irá analisar-se isoladamente cada classificador. Estes testes foram efetuados usando o conjunto de dados do UrbanSound8k.

Assim sendo, o primeiro classificador a ser analisado é o kNN. Neste, o parâmetro mais importante a ter em conta é o número de vizinhos que são escolhidos, a Figura 21 demonstra a probabilidade de acerto para um dado número de vizinhos.

Figura 22 - Variação da probabilidade de acerto para diferentes números de vizinhos, KNN

Ao analisar a Figura 21, conclui-se que o aumento do número de vizinhos nem sempre implica uma melhoria no desempenho deste classificador, contrariamente o seu aumento às vezes prejudica o resultado final. Desta forma, é comum utilizar um número baixo de vizinhos, pois apesar de não haver uma melhoria significativa, computacionalmente é mais eficaz.

De seguida, segue-se o *random forest*, em que o parâmetro a ter em conta é essencialmente o número de árvores de decisão. A Figura 22 demonstra a probabilidade de acerto para vários números de árvores criadas.

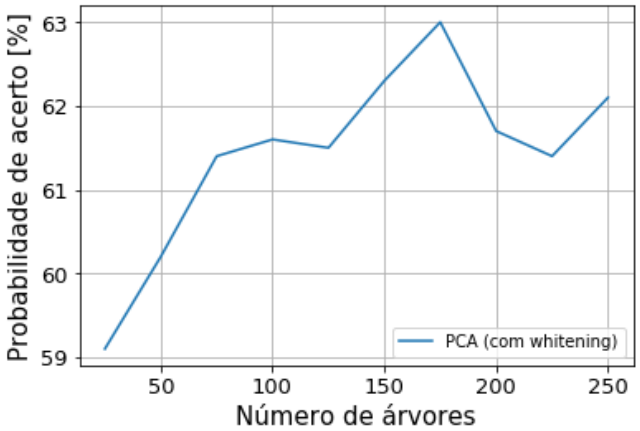

Figura 23 - Variação da probabilidade de acerto para diferentes números de árvores, random forest

Através da análise da Figura 22, verifica-se que há uma ligeira melhoria na precisão de classificação com o aumento do número das árvores de decisão. Contudo, não é rentável

utilizar um número muito elevado de árvores, pois a partir de um certo número de árvores os valores da probabilidade de acerto não apresentam uma melhora significativa.

O próximo classificador a ser analisado é o SVM. Por ser um classificador muito sensível aos parâmetros, como citado na subsecção 3.1.3, construiu-se uma grelha bidimensional, como um mapa de calor, variando os parâmetros C e gama. Nesta, as cores claras significam elevadas probabilidades de acerto, enquanto as cores escuras previsões mais reduzidas. A Figura 23 evidencia a grelha bidimensional.

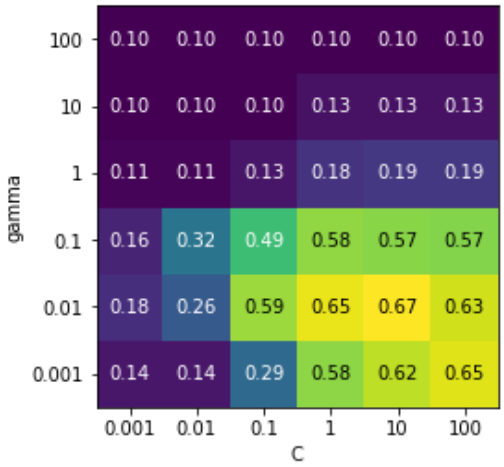

Figura 24 - Variação da probabilidade de acerto para diferentes valores de C e de gamma, SVM

Ao observar a Figura 23, podemos comprovar o quão sensível o SVM é relativamente à configuração dos parâmetros, pois ajustá-los pode alterar a precisão do classificador, passando de 10% a 67%. Desta maneira, ao utilizar este modelo os parâmetros que proporcionam melhor desempenho são C=10 e gama=0.01.

Por último, é avaliado o classificador redes neuronais. Como dito na subseção 3.1.4, há muitas alternativas para controlar a complexidade do mesmo, tendo como exemplos o número de camadas ocultas, o número de unidades em cada camada oculta, a função de ativação usada, entre outras. Desta forma, para verificar a influência dos parâmetros na eficiência do classificador, foram realizados testes com as duas funções de ativação, tanh e relu, para diferentes números de camadas ocultas e unidades em cada camada oculta. As Tabelas 4 e 5 mostram os resultados obtidos.

Tabela 4 - Variação das probabilidades de acerto, para diferentes números de camadas ocultas e unidades em cada camada oculta para tanh, redes neuronais

Função de ativação, ***tanh***		**Número de camadas escondidas**		
		1	2	3
Unidades em cada camada oculta	50	65,7%	66,3%	63,9%
	100	65,9%	66,4%	66,1%
	150	66,2%	67,2%	65,7%
	200	66,7%	66,7%	66,2%

Tabela 5 - Variação das probabilidades de acerto, para diferentes números de camadas ocultas e unidades em cada camada oculta para relu, redes neuronais

Função de ativação, ***relu***		**Número de camadas escondidas**		
		1	2	3
Unidades em cada camada oculta	50	65,8%	65,9%	64,5%
	100	66,8%	67,2%	65,1%
	150	67,3%	67,5%	65,2%
	200	67,6%	68,3%	65,5%

Ao analisar as tabelas acima, constata-se que a função de ativação relu produz melhores resultados perante a tanh, verificando também que aumentando o número de camadas ocultas nem sempre se traduz numa melhoria do resultado final. Consequentemente para o modelo proposto, a melhor hipótese será a função relu, usando duas camadas ocultas com duzentas unidades por camada oculta.

Em suma, comprova-se o que foi mencionado na seção 3.1, pois o kNN, considerado o mais trivial, apresenta-se com os piores resultados na probabilidade de acerto. De seguida, com melhorias nos resultados de classificação, como o expetável, surge o *random forest*.

Finalmente, evidenciam-se os dois classificadores considerados os mais poderosos e vantajosos, contudo as redes neuronais apresentam um melhor desempenho face ao SVM.

Para avaliar o desempenho de um classificador é necessário saber qual a probabilidade total de acertar independentemente das classes, mas também é necessário conhecer qual a probabilidade de errar e a distribuição dos erros em cada classe. Para representar a distribuição dos erros por classe, usa-se uma matriz de confusão, que permite realizar uma análise mais detalhada do desempenho do classificador. Assim sendo, a Figura 24 apresenta a matriz de confusão efetuada sobre o conjunto de dados do UrbanSound8k, recorrendo ao classificador redes neuronais e PCA com branqueamento.

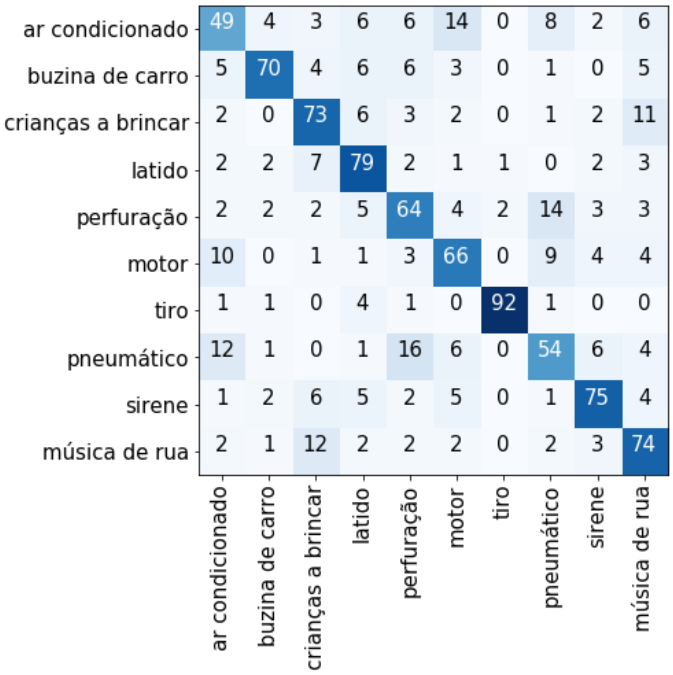

Figura 25 - Matriz de confusão sobre o conjunto de dados do UrbanSound8k, utilizando o classificador redes neuronais

Ao analisar a Figura 24, conclui-se que existe alguma confusão entre as classes com sons mecânicos repetitivos, ou seja, ar condicionado, perfuração, motor e pneumático. Há também algum erro entre as crianças a brincar e a música de rua, já que elas também têm padrões acústicos semelhantes. Aferindo também que as três classes que se evidenciam, são o tiro, a sirene e o latido. Contudo, como referido na seção 5.1, este conjunto de dados, UrbanSound8k, também fornece um nível de saliência para os excertos de som, indicando se o evento foi percebido em primeiro (foreground) ou segundo (background) plano da gravação. Os sons de primeiro plano são salientes e normalmente não distorcidos por outros sons, enquanto que os sons de segundo plano são frequentemente misturados com sons ambientes ou outros tipos de ruído. Desta forma e para o modelo redes neuronais, a classificação foi

feita para todo o conjunto de dados e isoladamente para os sons de primeiro e segundo plano. As probabilidades de acerto por classe são mostradas na Figura 25, enquanto que a Tabela 6 evidencia a probabilidade de acerto de cada conjunto de saliência.

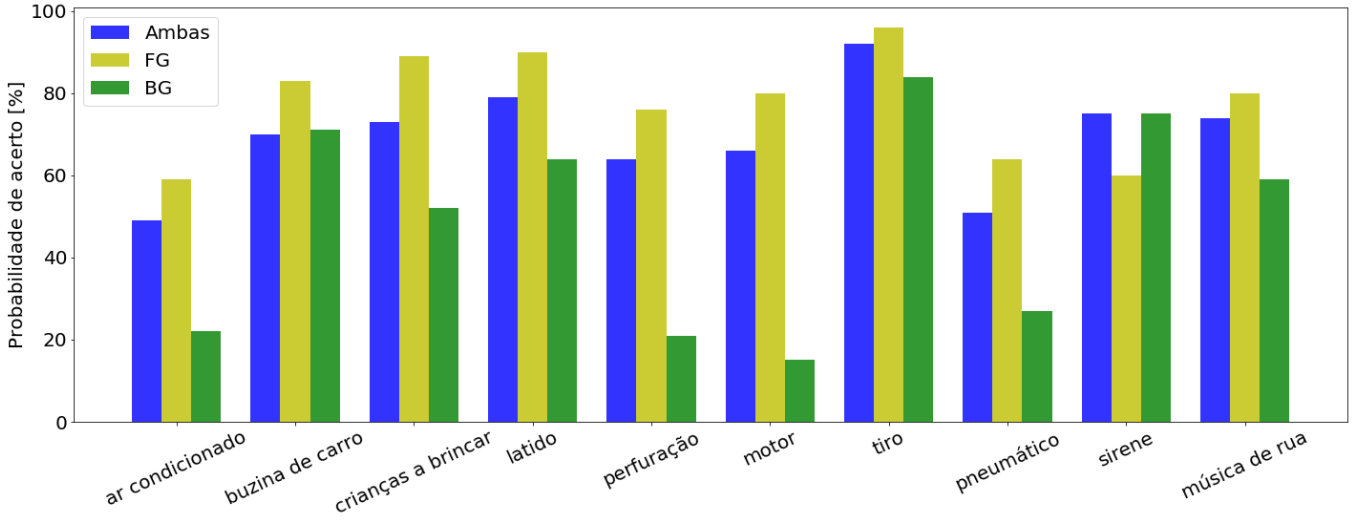

Figura 26 - Gráfico da probabilidade de acerto para todas as classes do conjunto de dados UrbanSound8k. No azul não foi considerada a saliência. Respetivamente ao restantes são de acordo com a saliência do som, em que o amarelo corresponde aos sons de primeiro plano (FG) e o verde aos sons de segundo plano (BG)

Tabela 6 - Probabilidades de acerto para os diferentes grupos de saliência

Conjuntos de Saliência	Probabilidade de acerto [%]
Ambas	68
Primeiro Plano (FG)	77
Segundo Plano (BG)	51

Como esperado, ao analisar a Figura 25 e a Tabela 6, comprova-se que os sons de primeiro plano são mais fáceis de reconhecer, já que a sua relação sinal-ruído (SNR) é alta. Particularmente, para os sons como a buzina de carro, crianças a brincar, latido e tiro, que apresentam um bom desempenho na sua identificação. Por outro lado, na identificação dos sons de segundo plano, o desempenho foi degradado porque as características foram corrompidas por ruído externos, o que leva a possuir valores baixos. Especialmente para os sons mecânicos repetitivos, como acima referidos.

Relativamente aos resultados finais da probabilidade de acerto, do modelo implementado, constata-se que o modelo é válido, pois apresenta probabilidades de acerto superiores ou similares a muitos classificadores encontrados na literatura [Puente, 2018, Antich, 2017, Huzaifah, 2017], que obtêm probabilidade de acerto de 66%, 68% e 64%,

respetivamente. Contudo, encontra-se num limiar abaixo de alguns classificadores mais rigorosos e exigentes. No [Ye et al., 2017] uma mistura de modelos combinados com características locais e globais atingiram uma probabilidade de acerto de 77%, enquanto que no [Piczak, 2015] e [Salamon and Bello, 2017] usam redes neuronais convolucionais e obtêm uma probabilidade de acerto de 73% e 79%, respetivamente.

O conjunto de exemplos usado para a avaliação do classificador, denominado de conjunto de teste, deve conter exemplos diferentes dos usados para treinar o classificador. É necessário usar novos exemplos no processo de avaliação para obter uma estimativa fiável do desempenho do classificador e assim medir a sua capacidade de generalização. Deste modo, para além da validação cruzada, técnica usada na obtenção dos resultados anteriores, foram escolhidas classes que coexistiam em ambas as base de dados, com o objetivo de umas servirem como conjunto de treino e as outras como conjunto de teste. Assim sendo, foram escolhidas 4 classes do conjunto de dados ESC-50 (buzina de carro, latido, motor e sirene), em que para cada classe retirou-se 30 excertos de sons. Posto isto, utilizou-se a base de dados UrbanSound8k, com os seus 8732 excertos de sons, como conjunto de treino e os exemplos retirados do ESC-50 como conjunto de teste. Desta maneira, é possível verificar a capacidade de generalização do modelo implementado, pois as amostras do conjunto de teste não existem no conjunto de treino. A Figura 26 evidencia a matriz de confusão efetuada sobre os dados.

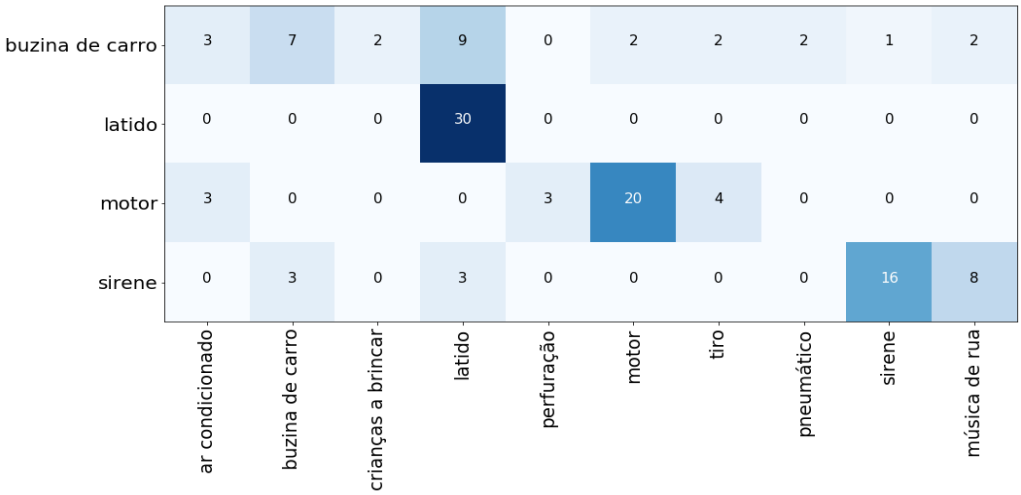

Figura 27 - Matriz de confusão treinada pela base de dados do UrbanSound8k e testada por alguns exemplos da base de dados ESC-50

Ao analisar a Figura 26, verifica-se que a classe do latido é totalmente identificada pela classe UrbanSound8k, enquanto que nas outras classes persistem alguns erros. A classe com mais confusão é a buzina de carro. No entanto, é obtida uma probabilidade de acerto entre os conjuntos de dados de 61%, percentagem esta bastante boa para bases de dados distintas.

Para além deste teste, foram classificados outros sons gravados pessoalmente, com o objetivo de testar sons que não existam nas bases de dados treinadas, em que nestes a maioria foi classificada com sucesso.

Em suma, podemos aferir que o modelo implementado possui capacidade de generalização, pois não só consegue classificar excertos de sons pertencentes à sua base de dados, como também o faz para sons que estejam fora desta.

5.3 Resultados: Multi-label

Neste subcapítulo apresenta-se os resultados para a classificação *multi-label*. Para esta tarefa, dez classificadores binários de redes neuronais foram treinados, um para cada classe, usando como exemplos positivos os membros da classe e como exemplos negativos os excertos de sons de todas as outras classes. Comparando com os resultados obtidos anteriormente, este é um cenário mais realista, já que um único excerto de som pode ser anotado com múltiplas classes, uma situação que é aplicável a vários eventos de áudio. De notar que a precisão neste tipo de classificação pode ser enganosa, pois atribuir um rótulo negativo a quase todos os exemplos de teste, resulta em probabilidades de acerto em torno de 90%. Desta forma, há medidas de desempenho mais fidedignas, que são obtidas através das curvas ROC, mencionadas na secção 3.2. A Figura 27 mostra a representação dessas curvas para o conjunto de dados UrbanSound8k. Foi tida em conta esta base de dados ao invés da ESC-50, pois esta última como possui 50 classes faria com que no gráfico houvesse 50 curvas, o que tornaria o gráfico confuso.

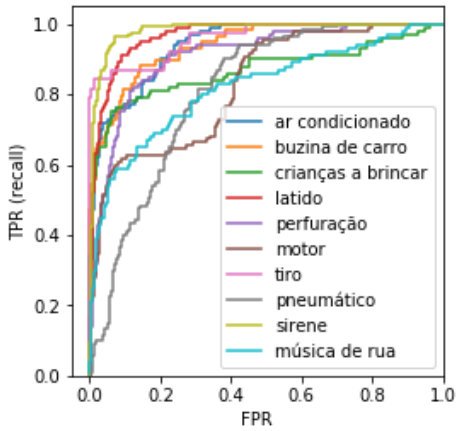

Figura 28 - Representação das curvas ROC para as 10 classes da base de dados do UrbanSound8k

As curvas ROC são gráficos bidimensionais que descrevem os *trade-offs* entre os benefícios (True Positives) e os custos (False Positives). As redes neuronais, como muitos outros modelos de classificação, produzem um pontuação que reflete o grau de certeza na decisão da classe prevista. As informações na curva ROC podem ser resumidas calculando a área sob a curva, denominado de AUC. A Figura 28 mostra os resultados obtidos do AUC.

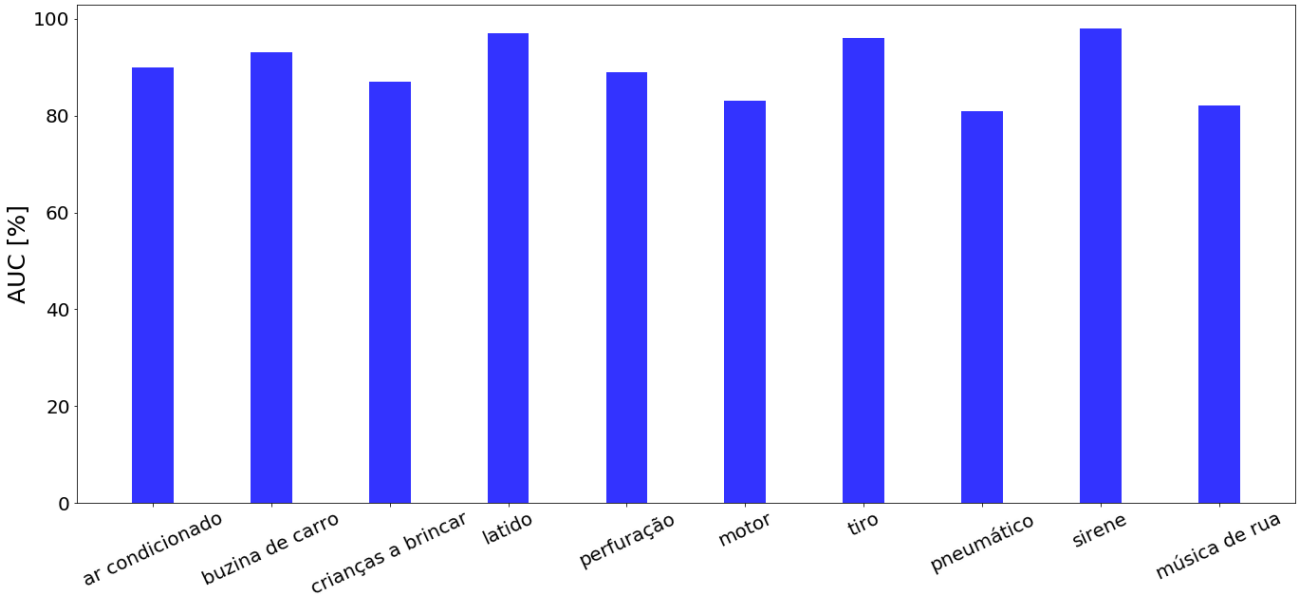

Figura 29 - Percentagem da área sob a curva ROC, denominada de AUC, para todas as classes da base de dados UrbanSound8k

Analisando os resultados, destacam-se 3 sons com as melhores previsões, sendo eles a sirene, o tiro e o latido. Por outro lado, as classes como o pneumático, o motor, a música de rua e as crianças a brincar, têm desempenhos piores.

Para melhor exemplificar o desempenho do *multi-label*, foi criado um ficheiro de som, com 22 segundos, de modo a simular uma paisagem sonora urbana. Este ficheiro possui cinco eventos sonoros, em que cada evento contém dois ou mais sons sobrepostos. Deste modo, o objetivo é assinalar quais as classes presentes em cada evento. Relativamente aos sons, as classes utilizadas para este teste foram: a buzina de carro, latido, tiro, avião, sirene e música de rua. A Figura 29 mostra a paisagem sonora utilizada e as separações por eventos sonoros.

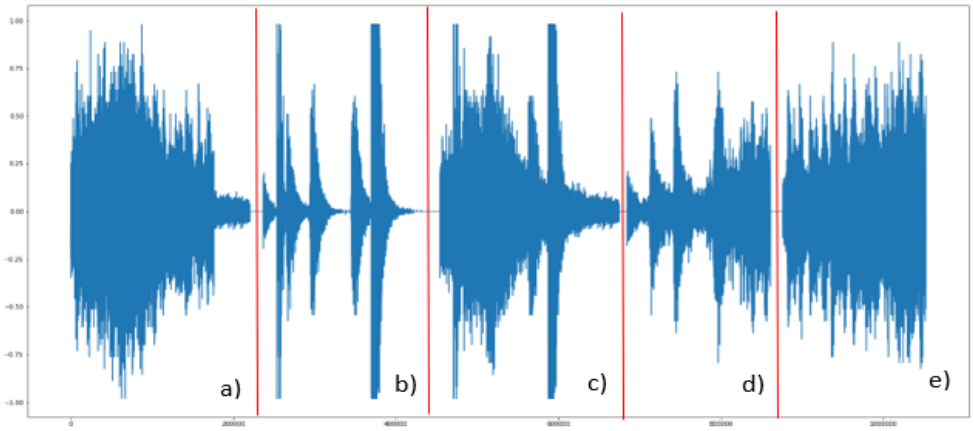

Figura 30 - Paisagem sonora utilizada e separada por eventos sonoros

Posto isto e com auxílio do classificador redes neuronais, a Tabela 7 demonstra a existência de diferentes classes em cada evento sonoro, evidenciando a diferença entre esta classificação e a de *multi-class*, em que nesta cada evento possui apenas uma das classes pré-definidas e mutuamente exclusivas.

Tabela 7 - Presença e Ausência da classe, em cada evento sonoro

"1" – Presença da classe	**Eventos Sonoros**				
"0" – Ausência da classe	a)	b)	c)	d)	e)
Classes de sons					
Buzina de carro	1	0	0	0	1
Latido	0	1	1	1	0
Tiro	0	1	1	0	0
Avião	1	0	1	0	0
Sirene	0	0	0	1	1
Música de rua	1	0	0	0	1

Na realização deste tipo de classificação, *multi-label*, recorreu-se a um limiar de probabilidade para definir que classes estão presentes num determinado evento sonoro. Assim sendo, para identificar 2 sons sobrepostos o limiar mínimo necessário é de 20%, querendo isto dizer que todas as probabilidades acima desta são assinaladas como presentes e as restantes como ausentes. No caso da identificação de 3 sons sobrepostos, é necessário baixar o limiar para 5%, de maneira a conseguir identificá-los corretamente. Consequentemente limitou o sistema para uma identificação máxima de 2 sons sobrepostos,

pois ao utilizar um limiar mínimo de 5% torna o sistema inviável, uma vez que acresce a possibilidade de mais sons ausentes se tornarem presentes.

Capítulo 6

Conclusão e trabalho futuro

6.1 Conclusão

Nesta dissertação, o sistema proposto avalia o campo de som e extrai as informações relevantes sobre os eventos sonoros ocorridos na cidade, por forma a criar um sistema capaz de “escutar” e “entender” a paisagem sonora urbana. Para esse efeito, é apresentada uma abordagem baseada em conteúdo, que depende exclusivamente das características extraídas do sinal de áudio.

Por forma a confirmar a credibilidade do sistema, este foi avaliado para dois conjuntos de dados e os resultados finais comparados com o estado da arte. Após comprovar a sua validade e generalização, conclui-se que é possível melhorar o desempenho do sistema, devido principalmente a dois fatores. O primeiro tem a ver com as características usadas no modelo, pois ao basear-se em descritores espetrais de curta duração, não é tido em consideração as relações temporais de médio ou longo prazo que diferenciam os sons urbanos. Isto acontece quando é usado o método BoF, pois ao converter a sequência de características num único vetor, descarta dependências temporais [Lagrange et al., 2015]. O segundo fator é devido à não utilização de algoritmos para redução de ruído de fundo, visto que para o caso do conjunto de dados do UrbanSound8k, ao ser considerada a situação em que é garantido uma melhor SNR, usando sons com a saliência de primeiro plano, foi alcançada uma probabilidade de acerto de 77%.

Por forma a tornar a identificação mais realista, foi desenvolvida a *classificação multi-label*, a fim de entender quantas classes existem num evento sonoro. Uma vez que um ambiente urbano pode ser composto por diferentes classes em simultâneo, o que faz com seja um problema difícil de resolver usando apenas classificação *multi-class*.

6.2 Trabalho futuro

A extração de conteúdo sonoro urbano e a classificação/identificação de eventos sonoros são blocos de construção essenciais das s*mart cities*, no sentido de proporcionar conforto e segurança aos cidadãos. O projeto levou em consideração a intenção de operar em um conceito IoT, onde vários desses dispositivos poderiam ser implementados numa cidade para recolher informações acústicas urbanas, visto que esta dissertação não apresenta um fim do projeto, mas apenas uma parte do mesmo.

Contudo, há considerações que devem ser tidas em conta para possuir um sensor de áudio inteligente em pleno funcionamento. Uma delas tem a ver com as capacidades de processamento, pois é necessário garantir que os algoritmos possam operar em tempo real, sem consumir muitos recursos computacionais do sistema. Para essa finalidade, foram realizados alguns testes que mostraram que o sistema pode ser executado em tempo real no Raspberry Pi, pois quando se realizou o artigo, foram recolhidos os tempos de processamento, usando os classificadores kNN e SVM, obtendo mais ou menos 7 segundos a processar no Raspberry Pi, contra 2 segundos a correr num CPU Intel (R) Core (TM) i74710HQ. Na rotina testada, primeiramente foram carregados os dados de treino e feito o PCA com branqueamento sobre os mesmos, sabendo que eram 8732 eventos sonoros, retirados da base de dados do UrbanSound8k. De seguida, introduziu-se um áudio em bruto, extraíram-se as suas características, representaram-se os segmentos de áudio num vetor único de características e finalmente procedeu-se à classificação do mesmo. Contudo há melhorias no algoritmo que precisam de ser efetuadas e testadas com o classificador redes neuronais.

Bibliografia

J. Bello. Low-level features and timbre. MPATE-GE 2623 Music Information Retrieval, New York, 2017

United Nations. 2018 Revision of World Urbanization Prospects. Department of economic and social affairs. 2018

R. Higuti. Processamento Digital de Sinais – Análise Espectral Usando a DFT, 1989/1999

Burden of disease from environmental noise – quantification of healthy life years lost in europe. Tech. Rep., World Health Organization Regional Office for Europe; 2011

Y. Zheng, L. Capra, O. Wolfson and H. Yang. Urban computing: concepts, methodologies, and applications. ACM Trans Intell Syst Technol, 2014

Urban life: open-air computers. The Economist; Oct. 2012

D. Steele, D. Krijnders and C. Guastavino. The sensor city initiative: cognitive sensors for soundscape transformations. In: GIS Ostrava, 2013

T. Liu, Y. Zheng, L. Liu, Y. Liu and Y. Zhu. Methods for sensing urban noises. May 2014

H. Héritier, D. Vienneau, P. Frei, I. Eze, M. Brink, N. Probst-Hensch and M. Roosli. The association between road traffic noise exposure, annoyance and health-related quality of life, 2014

M. Crocco, M. Cristani, A. Trucco and V. Murino. Audio surveillance: A systematic review. ACM Comput. Surv. 2016

R. Radhakrishnan, A. Divakaran and A. Smaragdis. Audio analysis for surveillance applications. In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005

G. Nagy, R. Rodigast and D. Hollosi. Energy based traffic density estimation using embedded áudio processing unit. In Audio Engineering Society Convention, Apr. 2014

M. Rychtáriková and G. Vermeir. Soundscape categorization on the basis of objective acoustical parameters. Applied Acoustics, 2013

A. Torija, D. Ruiz and Á. Ramos-Ridao. A tool for urban soundscape evaluationapplying support vector machines for developing a soundscape classification model. Science of The Total Environment, 2014

A. Agha, R. Ranjan and W.-S. Gan. Noisy vehicle surveillance camera: A system to deter noisy vehicle in smart city. Applied Acoustics, 2017

I. Kivelä, C. Gao, J. Luomala and J. Ihalainen. Design of networked low-cost wireless noise measurment sensors. Sensors and Transducers, 2011

C. Mydlarz, J. Salamon and J. Bello. The implementation of low-cost urban acoustic monitoring devices. Applied Acoustics, 2017

H. Ising and B. Kruppa. Health effects caused by noise: Evidence in the literature from the past 25 years. Noise and Health, 2004

J. Salamon, C. Jacoby and J. P. Bello. A dataset and taxonomy for urban sound research. In Proceedings of the 22nd ACM international conference on Multimedia, 2014

J. Kindcaid. A brief history of ASR: Automatic Speech Recognition, 2018

A. Klapuri and M. Davy. Signal processing methods for music transcription. Springer Science & Business Media, 2007

D. Wang and G. Brown. Computational Auditory Scene Analysis: Principles, Algorithms, and Applications. Wiley-IEEE Press, 2006

D. Botteldooren, T. Andringa, I. Aspuru, L. Brown, D. Dubois, C. Guastavino, C. Lavandier, M. Nilsson, A. Preis and et al. Soundscape for european cities and landscape: understanding and exchanging. In: COST TD0804 final conference: soundscape of European cities and landscapes, Soundscape-COST, 2013

C. Pham and P. Cousin. Streaming the sound of smart cities: experimentations on the smartsantander test-bed. In: Proceedings of the 2013 IEEE international conference on green computing and communications, GREENCOM-ITHINGS-CPSCOM´13. Washington, DC, USA: IEEE Computer Society, 2013

M. Kassler. Toward musical information retrieval. Perspectives of New Music, pág. 59–67, 1966

H. Dudley, R. Riesz, and S. Watkins, A Synthetic Speaker, J. Franklin Institute, Vol. 227, pp. 739-764, 1939.

A. Bregman. Auditory Scene Analysis. Cambridge, MA: MIT Press, 1990

B. Moore. Hearing – Handbook of Perception of Hearing. Academic Press, San Diego, California, 1995

C. Souza. Kernel Functions for Machine Learning Applications. 2010

M. Rossing. The Science of Sound. Addison Wesley, 1990

K. Pearson. On lines and planes of closest fit to systems of points in space. Phylosophical Magazine, 2(11):559–572, 1901

H. Hotelling. Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology, (24):417–441, 1933

R. Fisher. The use of multiple measurements in taxonomic problems. Annals of Eugenics, 7 (2):179–188, 1936

A. Mueller and S. Guido. Introduction to machine learning with python – a guide for data scientists, Gravenstein Highway North, Sebastopol, CA 95472, pág.71-81, 2016

Z. Zhang. Customizing kernels in Support Vector Machines. University of Waterloo in fulfilment of the requirement for the degree of Master of Mathematics, Canada, 2007

B. Vojt. Deep neural networks and their implementation, 2016

K. Seyerlehner. Content-Based Music Recommender Systems: Beyond simple Frame-Level Audio Similarity. PhD thesis, Johannes Kepler University, Linz, 2010

B. McFee, C. Raffel, D. Liang, D. Ellis, M. McVicar, E. Battenberg and O. Nieto. librosa: Audio and music signal analysis in python. In Proceedings of the 14th python in science conference, pág. 18–25, 2015

A. Coates and A. Ng. The importance of encoding versus training with sparse coding and vector quantization. In Proceedings of the 28th international conference on machine learning (ICML-11), pág. 921– 928, 2011

J. Ye, T. Kobayashi and M. Murakawa. Urban sound event classification based on local and global features aggregation. Applied Acoustics, 117:246 – 256, 2017

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011

J. Salamon and J. Bello. Deep convolution neural networks and data augmentation for environmental sound classification. In IEEE signal processing letters, 2016

K. Piczak. Environmental sound classification with convolutional neural networks. In 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP), Sept 2015.

E. Cakir. Multilabel Sound Event Classification with Neural Networks, 2014

J. Antich. Audio event classification using deep learning in an end-to-end approach, 2017

J. Salamon and J. Bello. Deep convolutional neural networks and data augmentation for environmental sound classification. IEEE Signal Processing Letters, 2017.

M. Lagrange, G. Lafay, B. Défréville, and J.-J. Aucouturier. The bag-of-frames approach: A not so sufficient model for urban soundscapes. The Journal of the Acoustical Society of America, 2015.

T. Virtanen, M. Plumbley and D. Ellis. Introduction to Sound Scene and Event Analysis. Springer International Publishing AG, 2018

A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen. DCASE 2017 challenge setup: Tasks, datasets and baseline system. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017), pages 85–92, November 2017.

K. Piczak. Esc: Dataset for environmental sound classification. In Proceedings of the 23rd ACM International Conference on Multimedia, MM ’15, pages 1015–1018, New York, NY, USA, 2015

J. Salamon, C. Jacoby, and J. Bello. A dataset and taxonomy for urban sound research. In ACM Multimedia, 2014.

J. Gemmeke, D. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. Moore, M. Plakal and M. Ritter. Audio Set: Na ontology and human-labeled dataset for audio events. Google, Inc., Mountain View, CA, and New York, NY, USA, 2017

M. Huzaifah. Comparison of Time-Frequency Representations for Environmental Sound Classification using Convolution Neural Networks, 2017

Y. Aytar, C. Vondrick and A. Torralba. SoundNet: Learning Sound Representations from Unlabeled Video. 2016

A. Ferreira. DFT, FFT and DCT. Sistemas de telecomunicações definidos por software & processamento de sinal em tempo real.

G. Marques. Machine Learning Techniques for Music Information Retrieval. Universidade de Lisboa Faculdade de Ciências Departamento de Informática. 2014

W. Burgos. Gammatone and MFCC features in speaker recognition. Melbourne, Florida, 2014

V. Lobo. Árvores de decisão. EN/ISEGI, 2010

S. Puente. Single and Multi-Label Environmental Sound Classification Using Convolutional Neural Networks. Gothenburg, Suécia, 2018