UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

JOÃO PEDRO GUBERT DE SOUZA

Detecção de apneia do sono em exames de polissonografia utilizando algoritmos de aprendizado de máquina

Monografia apresentada como requisito parcial para a obtenção do grau de Bacharel em Engenharia da Computação

Orientador: Prof. Dr. Leandro Krug Wives
Coorientador: MSc. Oscar Ortegon

Porto Alegre
2020

“A educação tem raízes amargas,
mas os seus frutos são doces.”
— ARISTÓTELES

AGRADECIMENTOS

Agradeço à minha família que sempre me incentivou e me possibilitou estudar em uma das melhores universidades do Brasil. Um agradecimento especial à minha avó Angelina, que saiu de casa ainda na adolescência para se formar professora, e sempre me mostrou o poder e a importância da educação.

Agradeço meu orientador Leandro Wives e ao coorientador Oscar Ortegon pelo apoio, conhecimento e paciência, tornando possível este trabalho. Agradeço também a todos os professores do curso que contribuíram para a minha formação, com eles aprendi muito mais do que somente o plano de ensino.

Agradeço à Clínica do Sono, na pessoa do Dr. Denis Martinez, pelo auxílio nas questões técnicas da área da saúde, tempo, e por ter disponibilizado os exames utilizados neste trabalho. Sem esta parceria este trabalho não seria possível.

Quero agradecer também ao meu colega Renan Bortoluzzi pelo apoio e contribuição nas soluções utilizadas neste trabalho.

Agradeço aos colegas da Universidade Federal de Santa Maria, onde fiz as etapas iniciais do curso de Engenharia da Computação, pelas horas de grupo de estudo, risadas e jogos de truco. Vocês tornaram a saudade de casa um pouquinho mais fácil de aguentar.

Um agradecimento especial aos colegas Rodrigo Morawski, Tiago Fróes e Rafael Calçada, pelos inúmeros cafés, conversas e almoços no restaurante universitário.

Agradeço também à Julia, minha namorada, que me acompanhou durante os 8 anos desta jornada. Obrigado pela companhia nas noites de estudo, nos momentos difíceis e nas comemorações de cada passo desta trajetória.

RESUMO

A apneia do sono é um distúrbio comum do sono, que afeta aproximadamente 200 milhões de pessoas. Porém a detecção de apneia em exames de polissonografia ainda é algo extremamente trabalhoso de se fazer de forma manual. O objetivo deste trabalho é realizar esta detecção de forma automatizada, reduzindo o tempo e o esforço necessários para o diagnóstico de cada paciente. Para tanto, utilizou-se algoritmos de aprendizado de máquina, do tipo redes neurais artificiais (RNA), e os sinais do exame de polissonografia que estão diretamente associados com a respiração do paciente. Após uma etapa de tratamento dos dados, que inclui filtragem, divisão em épocas e balanceamento, foram avaliadas 14 arquiteturas diferentes para encontrar o modelo de rede neural com melhor desempenho. O modelo com o melhor desempenho atingiu um F1-Score de 85%.

Palavras-chave: Detecção de apneia. Polissonografia. Aprendizado de máquina. RNA.

Obstructive Sleep Apnea Detection in polysomnography exams using machine learning algorithms

ABSTRACT

Sleep apnea is a common sleep disorder that affects approximately 200 million people. The detection of apnea in polysomnography exams is still extremely hard-working and time-consuming to do manually. The objective of this work is to perform this detection automatically, reducing the time and effort required for the diagnosis of each patient. For this purpose, machine learning algorithms were used, such as *Artificial Neural Networks* (ANN), and the polysomnography examination signals that are directly associated with the patient's breathing. After a stage of data treatment, which includes filtering, time division and balancing, 14 different architectures were evaluated to find the best performing neural network model. The model with the best performance reached an F1-Score of 85%.

Keywords: Apnea Detection, Polysomnography, Machine learning, ANN.

LISTA DE ABREVIATURAS E SIGLAS

RNA	Redes neurais artificiais
AOS	Apneia Obstrutiva do Sono
RQ	*Research question* (Questão de pesquisa)
IAH	Índice de apneia-hipopneia
ACS	Apneia central do sono
AMS	Apneia mista do sono
PSG	Polissonografia
ECG	Eletrocardiograma
EEG	Eletroencefalograma
EOG	Eletrooculograma
EMG	Eletromiograma
SpO2	Saturação periférica de oxigênio
EDF	*European data format*
EVT	*Microsoft Windows Event Viewer Log*
RNN	*Recurrent neural networks*
MLP	*Multilayer perceptron*
SVM	*Support vector machine*
HMM	*Hidden Markov Model*
LSTM	*Long short-term memory*
REPTree	*Reduced-error pruning tree*
SMOTE	*Synthetic Minority Over-Sampling Technique*
ENN	*Nearest neighbours*
SGD	*Stochastic gradient descent*
ROC	*Receiver Operating Characteristic*

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

A Apneia Obstrutiva do Sono (AOS) é um distúrbio comum do sono, caracterizado por pausas respiratórios repetitivas causadas pelo colapso das vias aéreas superiores durante o sono (RAVELO-GARCIA et al., 2015). Esse distúrbio leva a um sono fragmentado, gerando dores de cabeça, distúrbios cognitivos e fadiga durante o dia. Quando o paciente tem uma apneia, ocorre uma queda da oxigenação do sangue, que gera um aumento dos batimentos cardíacos e pressão arterial, e, por isso, a AOS também pode levar a doenças como hipertensão, acidentes cardiovasculares e doenças cardíacas (CAVALLARI et al., 2002). Segundo Zhang et al. (2013), estima-se que 200 milhões de pessoas sofram de AOS.

Para avaliar a intensidade do quadro do paciente, é considerada a quantidade de apneias ocorridas por hora, chamado de índice de apneia-hipopneia (IAH). Um paciente com IAH até 5 é considerado normal; de 5 a 15 é um grau leve; de 15 a 30 é um grau moderado; e um paciente com mais de 30 apneias por hora é considerado um grau grave. Segundo Pombo, Garcia e Bousson (2017), dentre os eventos da síndrome da apneia do sono estão a apneia obstrutiva do sono (AOS), apneia central do sono (ACS), apneia mista do sono (AMS) e hipopneia.

Diagnósticos de apneia quando realizados manualmente demandam muito tempo para análise visual dos sinais do exame de polissonografia (PSG), e é um processo sujeito a falhas humanas, devido ao cansaço ou falta de atenção do técnico que o realiza. Através de visitas à *Clínica do Sono*, clínica especializada em exames de polissonografia localizada em Porto Alegre, teve-se a oportunidade de acompanhar um técnico realizando o diagnóstico de apneia do sono de forma manual, e através desse acompanhamento foi possível compreender quais sinais do exame de polissonografia são relevantes para a realização do diagnóstico de apneia.

Após revisão na literatura, foi possível observar que existem trabalhos realizando diagnósticos de apneia utilizando métodos de aprendizado de máquina sobre o sinal de eletrocardiograma (ECG) do paciente e obtendo resultados satisfatórios, alguns ultrapassando a marca de 90% de acurácia conforme demonstrado no Capítulo 3. Porém, deve-se levar em consideração que nesses trabalhos foram utilizados dados que se aproximam de um mundo ideal, pois foram previamente tratados com filtros para eliminação de ruídos, e selecionados apenas exames que não apresentem inconsistência nos sinais. Nos exames de polissonografia que nos foram disponibilizados, é comum encontrar diversos tipos de

inconsistência nos sinais, devido às dificuldades de captação do sinal que podem surgir durante o exame de polissonografia, como ruídos causados pela movimentação do paciente, sensores com falhas ou que acabam caindo durante o exame, e até mesmo o suor excessivo nos pacientes.

Visando realizar diagnósticos de apneia do sono de forma mais rápida e eficiente, optou-se por explorar técnicas de aprendizado de máquina. Segundo Šter e Dobnikar (1996), a utilização de algoritmos de redes neurais para fazer o reconhecimento de padrões é extremamente eficiente, e vêm sendo cada vez mais utilizados para realizar diagnósticos médicos. Para o reconhecimento de padrões que indicam que o paciente está sofrendo de apneia do sono, foram utilizados modelos de redes neurais, e também o algoritmo de *Bagging* (detalhes na seção 2.3.5), utilizando múltiplos sinais para o treinamento dos modelos, de forma a garantir que a presença de ruído em um determinado canal não prejudique a performance do modelo. Para avaliação dos modelos gerados, são utilizadas as métricas de acurácia e *F1-Score*.

Neste trabalho, além da construção um modelo de rede neural que utiliza múltiplos sinais presentes no exame de polissonografia, foi explorado o treinamento com dados não tratados previamente, e avaliado se isto tornará o modelo robusto e eficiente para realizar o diagnóstico em clínicas que realizam exames de polissonografia. Para encontrar a melhor abordagem e método mais eficaz na resolução desse problema, foram propostas as seguintes perguntas de pesquisa (RQs):

RQ1: Quais os métodos computacionais mais adequados para realizar a detecção de AOS com a maior acurácia?

RQ2: Quais os sinais presentes no PSG contêm informação relevante para o diagnóstico da AOS?

RQ3: Os resultados obtidos através de um modelo de redes neurais são satisfatórios?

Nas próximas seções serão apresentados em mais detalhes o marco conceitual para a realização deste trabalho, uma visão geral dos trabalhos relacionados, e, por fim, será apresentado o modelo que apresentou melhor desempenho quanto às métricas de acurácia e *F1-Score* que foram analisadas. Finalmente, são apresentadas as conclusões, limitações e trabalhos futuros.

2 MARCO CONCEITUAL

Este capítulo descreve os conceitos utilizados para a realização e entendimento desta pesquisa.

2.1 Apneia obstrutiva do sono

Apneia obstrutiva do sono (AOS), é um distúrbio crônico no qual o fluxo de ar é interrompido por no mínimo 10 segundos, e é o tipo mais comum de apneia do sono (GUTTA; CHENG, 2016). Segundo Cavallari et al. (2002), a AOS está presente em 9% da população masculina entre 30 e 60 anos, e em 4% da população feminina nessa mesma faixa etária. Ainda segundo Cavallari et al. (2002), o quadro clínico da doença pode abranger até centenas de pausas respiratórias durante o sono, com dessaturações intensas da oxi-hemoglobina, arritmias cardíacas e sintomas diurnos e noturnos, como enurese noturna, cefaleia matinal, sonolência excessiva diurna, queda do rendimento intelectual, sintomas depressivos, impotência sexual e até alterações da personalidade.

De acordo com Gutta e Cheng (2016), pacientes com AOS também têm maior risco de hipertensão, doenças cardíacas e ataques cardiovasculares.

Para avaliar a intensidade do quadro do paciente, é utilizada a quantidade de apneias ocorridas por hora, chamado índice de apneia-hipopneia (*IAH*). De acordo com Daltro et al. (2006), o IAH é aplicado então da seguinte forma:

- Normal - até 5 eventos por hora;
- Leve - de 5 a 15 eventos por hora;
- Moderada - de 15 a 30 eventos por hora;
- Grave - mais de 30 eventos por hora.

Apesar da AOS estar presente na vida de milhares de pessoas, pacientes com apneia do sono raramente estão cientes de sua condição, e levam muito tempo para realizar o diagnóstico e iniciar o tratamento. Pelo menos 75% dos pacientes adultos com AOS ainda não foram diagnosticados (MAURER, 2008).

2.2 Exame de polissonografia

Para realizar o diagnóstico de pacientes com síndrome da apneia do sono, geralmente é utilizado o exame de polissonografia (PSG). Nesse exame, é feito o monitoramento dos sinais fisiológicos do paciente durante todo o tempo de sono. Dentro os sinais monitorados estão o eletrocardiograma (ECG), eletroencefalograma (EEG), eletrooculograma (EOG), eletromiograma (EMG), fluxo aéreo (termistor nasal e bucal), frequência cardíaca, saturação periférica de oxigênio (SpO2), esforço respiratório (cintas torácica e abdominal), movimento do queixo e pernas, ronco (microfone no queixo), além registrar a posição que o paciente se encontra.

De acordo Gutta et al. (2017), o PSG é um exame com custo elevado, demorado e inconveniente para o paciente, que precisa dormir em média 8 horas em um ambiente diferente do seu habitual, e com diversos eletrodos conectados ao seu corpo. Ainda segundo Gutta et al. (2017), a baixa disponibilidade de laboratórios para realizar o PSG e a pouca quantidade de especialistas do sono leva a um aumento no tempo de diagnóstico e tratamento da síndrome da apneia do sono.

Os dados gerados pelo PSG usualmente são armazenados no formato *European data format* (EDF) (KEMP; OLIVAN, 2003), que é um padrão adotado para o armazenamento multicanal de sinais biológicos e físicos. Para realizar a visualização e diagnósticos do PSG, a Clínica do Sono de Porto Alegre, clínica que nos cedeu os exames, utiliza o software *Poliwin* da empresa EMSA. Neste software é possível realizar a visualização de todos os sinais capturados no exame, e realizar anotações com os diagnósticos. Estas anotações podem ser posteriormente exportadas no formato Microsoft Windows Event Viewer Log (EVT).

2.3 Aprendizado de máquina

Aprendizado de máquina é uma área dentro do campo da inteligência artificial, que segundo Luger (2005), envolve a generalização a partir da experiência. A partir de uma experiência limitada, o modelo gerado deve ser capaz de generalizar corretamente para ocorrências de domínio não vistas anteriormente.

De acordo com Mahmud et al. (2018), as técnicas convencionais de aprendizado de máquina podem categorizadas em dois grandes conjuntos: aprendizado supervisionado e não supervisionado. No método de aprendizado supervisionado, a classificação é feita

utilizando um conjunto de dados de entrada e saída já conhecidos. Em contrapartida, o método de aprendizado não supervisionado agrupa os dados de entrada identificando sua similaridade e utiliza isso para a classificação de amostras desconhecidas. Além disso, existe uma outra categoria, chamada aprendizado por reforço, onde o sistema aprende através de experiências obtidas da interação com o ambiente.

Segundo Foster, Koprowski e Skufca (2014), nos últimos anos houve um grande crescimento no uso de métodos computacionais para a análise de sinais biomédicos, em que a grande maioria utiliza métodos de inteligência artificial ou aprendizado de máquina.

2.3.1 Redes Neurais

De acordo com Lippmann (1987), redes neurais são modelos computacionais que foram inspirados pelo sistema nervoso biológico, e visam atingir uma boa performance computacional através de uma interconexão densa de elementos simples, que assemelham seu comportamento a um neurônio. Esses algoritmos são basicamente modelos matemáticos que "aprendem"através de exemplos (aprendizado supervisionado).

Segundo Lippmann (1987), modelos de redes neurais são especificados pela topologia da rede, características dos neurônios e regras de treinamento. Essas regras especificam um conjunto inicial de pesos e indicam como os pesos devem ser adaptados durante o uso para melhorar o desempenho.

As redes neurais podem ser do tipo *feed-forward* ou *recurrent*. Segundo Mahmud et al. (2018), redes neurais do tipo *feed-forward* realizam seu processamento de forma unidirecional, da entrada para a saída. Já nos modelos *recurrent neural networks* (RNN), a saída do estado atual depende também das saídas dos estados anteriores. Devido a essa propriedade do tipo "memória", RNN ganhou muita popularidade em muitos campos envolvendo dados de *streaming* (por exemplo, mineração de textos, séries temporais, genomas).

Ainda segundo Mahmud et al. (2018), o crescimento do poder computacional acompanhado por conjunto de dados maiores já permite que cientistas de várias áreas apliquem essas técnicas em conjuntos de dados que antes eram intratáveis por seu tamanho e complexidade.

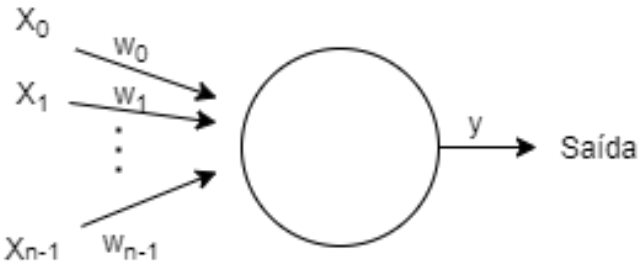
2.3.2 Perceptron

Desenvolvido por Frank Rosenblatt entre 1958 e 1962, o Perceptron (MALSBURG, 1986) foi um dos primeiros modelos de rede neural desenvolvido, e seu proposito era o reconhecimento de padrões. Esse modelo é visto como o tipo mais simples de rede neural, um classificador linear.

Como está ilustrado na Figura 2.1, o modelo Perceptron realiza uma soma ponderada de suas entradas, pelo peso correspondente, e gera uma saída, conforme Equação 2.1.

$$Saida = \sum_{i=0}^{n-1} (X_i w_i) \tag{2.1}$$

Figura 2.1: Modelo Perceptron



Fonte: Adaptado de Lippmann (1987)

2.3.3 Multilayer perceptron

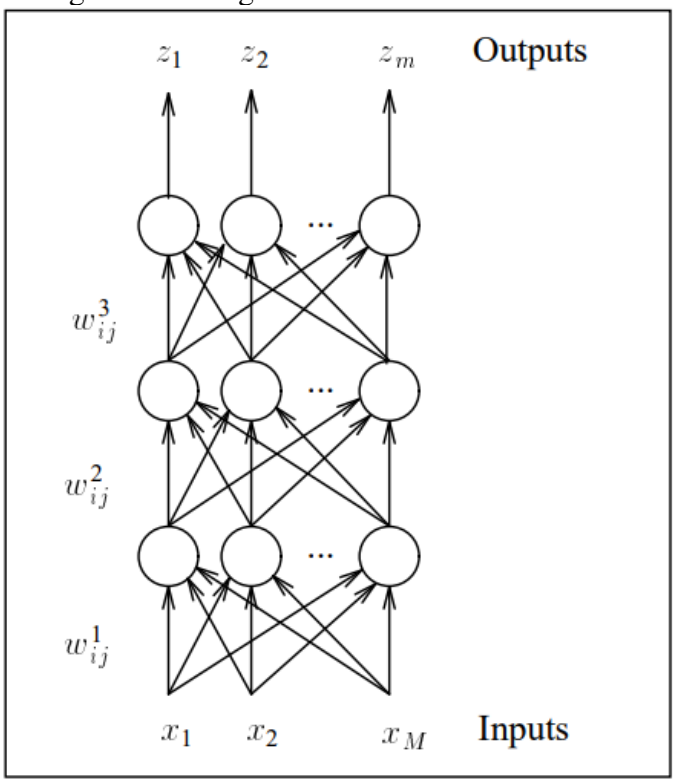
Segundo Gardner e Dorling (1998), *Multilayer perceptron* (MLP) consiste em um conjunto de neurônios simples interconectados, e desta forma representa o mapeamento não linear entre um conjunto de entrada e um conjunto de saída. Os neurônios são conectados por pesos, e a saída de cada neurônio é a soma das entradas aplicadas a uma função de ativação. Através da superposição de várias funções de ativação não lineares, o MLP consegue aproximar funções extremamente não lineares. A saída de um neurônio é utilizada como entrada para neurônios da camada seguinte da rede. Isso implica em uma direção de processamento de informações, característico das redes neurais do tipo *feed-forward*. A arquitetura de um MLP é variável, mas em geral consiste em várias camadas de neurônios.

Ainda segundo Gardner e Dorling (1998), MLP pode ser utilizada para resolver uma grande variedade de tarefas, que podem ser categorizadas como predição, aproxima-

ção de função e classificação de padrões.

Na Figura 2.2 está ilustrado o diagrama de uma MLP, onde x_i^j é a saída do neurônio *i* na camada *j*, e w_{ij}^k é o peso que conecta o neurônio *i* na camada *k-1* no neurônio *j* na camada *k*. A camada 1 é a primeira camada oculta, e a camada 0 é a camada de entrada.

Figura 2.2: Diagrama de uma rede neural MLP.



Fonte: Adaptado de (RUCK; ROGERS; KABRISKY, 1990)

2.3.4 Ensemble

Ensemble é uma técnica que surgiu no final da década de 80, que visa melhorar os resultados das predições feitas pela rede neural. Consiste no treinamento de múltiplos modelos, que combinados, atingem uma acurácia e diversidade maiores.

Segundo Hansen e Salamon (1990), o treinamento de redes neurais é um problema de otimização com muitos mínimos locais. Ou seja, dependendo dos pesos iniciais de treinamento, e do sequenciamento dos dados, podem chegar em diferentes resultados para os pesos da rede.

Ainda segundo Hansen e Salamon (1990), essa diferença nos pesos das redes correspondem a diferentes formas de generalização sobre diferentes padrões do conjunto de treinamento. Como cada rede comete erros de generalização em diferentes subconjuntos

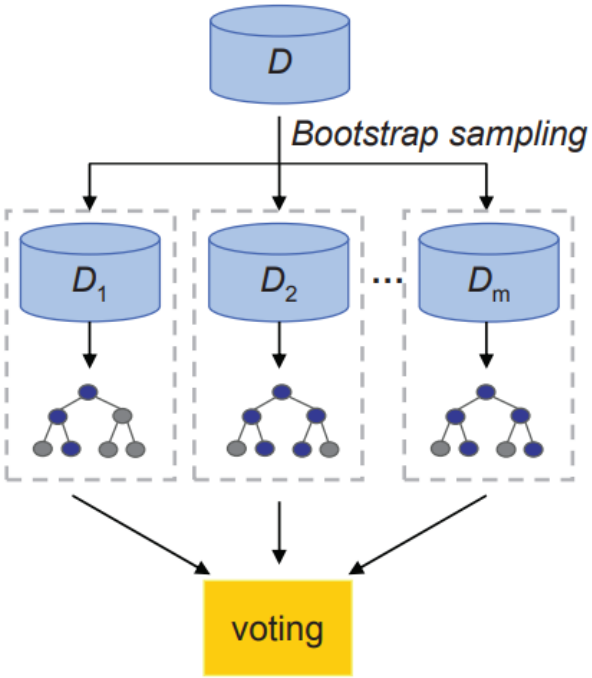
do espaço de entrada, a decisão coletiva produzida pelo *ensemble* tem menos probabilidade de ser um erro do que a decisão feita somente por qualquer uma das redes individuais. Alguns dos métodos de *ensemble* mais utilizados são *bagging*, *boosting* e *stacking*.

2.3.5 Bagging

Bagging (*bootstrap aggregating*) é uma técnica de *ensemble* utilizada para treinar múltiplos modelos a partir de réplicas do conjunto de dados de treinamento, que são geradas utilizando técnicas de amostragem com reposição. Segundo Breiman (1996), através desta técnica é possível gerar modelos individuais com diferenças significativas, aumentando a acurácia do conjunto. Ainda segundo Breiman (1996), as réplicas que são utilizadas para o treinamento dos modelos são geradas utilizando N amostras, escolhidas randomicamente e com reposição, do conjunto de dados original. Cada amostra pode aparecer diversas vezes, ou não, em qualquer réplica gerada (*bootstrap*).

Cada réplica gerada é utilizada para o treinamento de um modelo de classificação individual. Para realizar predições utilizando *bagging*, o conjunto de dados é avaliado por todos os modelos individuais, e a saída de cada modelo é combinada através de um sistema de votação. A Figura 2.3 ilustra o funcionamento do método de *bagging*.

Figura 2.3: Ilustração do funcionamento do método de *bagging*.

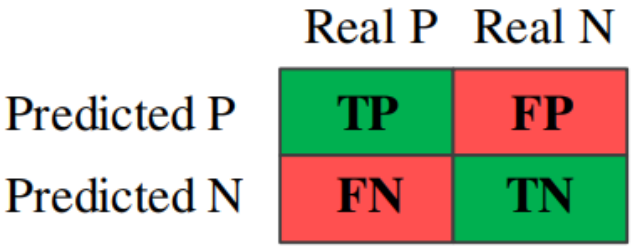


Fonte: (YANG et al., 2010)

2.3.6 Métricas de avaliação

Podemos avaliar a performance de modelos de aprendizado de máquina de diversas formas. Para avaliar modelos de classificação binária, segundo Gharib e Bondavalli (2019), as classificações realizadas pelo modelo podem ser divididas em 4 grupos: verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos. Nos casos de verdadeiro positivo e verdadeiro negativo, as predições foram feitas corretamente, enquanto nos casos de falso positivo e falso negativo, o modelo realizou a predição de forma incorreta. Organizando estes grupos na forma matricial, obtemos a matriz de confusão, conforme ilustrado na Figura 2.4.

Figura 2.4: Matriz de confusão



Fonte: Adaptado de (GHARIB; BONDAVALLI, 2019)

Utilizando estes grupos, podemos extrair algumas métricas, como *recall*, *precision* e F1-Score. O *recall* mostra a proporção entre predições verdadeiro positivo e a quantidade de casos positivos reais, conforme equação 2.3. Já a *precision*, mostra a proporção entre casos positivos previstos em relação aos casos positivos reais, conforme equação 2.4. A métrica F1-Score combina as medidas de *recall* e *precision* em uma única medida, a fim de medir a "eficácia"de pesquisa.

$$\textit{Acurácia} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.2}$$

$$\textit{Recall} = \frac{TP}{TP + FN} \tag{2.3}$$

$$\textit{Precision} = \frac{TP}{TP + FP} \tag{2.4}$$

$$\textit{F1-Score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2.5}$$

3 TRABALHOS RELACIONADOS

Foi feita uma revisão na literatura a fim de encontrar uma abordagem adequada para a resolução do problema apresentado, a metodologia feita para a revisão foi estruturada a partir dos métodos sistemáticos de mapeamento e revisão (KLOCK, 2018), onde foram definidas *strings* de busca, e atribuído aos artigos uma nota de relevância e contribuição para a nossa abordagem. Para atribuir as notas de cada trabalho, foi considerada a data de publicação (até 5 anos), a quantidade de citações, a acurácia que foi obtida pelo modelo proposto, e se o método de resolução do problema poderia ser utilizado para inspiração e base neste trabalho.

Na tabela 3.1 contém um resumo dos artigos mais relevantes encontrados nas pesquisas, por ordem de acurácia obtida. Vale ressaltar que a grande maioria os trabalhos estudados apresentaram como métrica de avaliação dos modelos apenas a acurácia, não apresentando outros indicadores, como o F1-Score.

Na grande maioria dos trabalhos relacionados, os autores realizam a extração de *features* do sinal do eletrocardiograma, como intervalo RR, amplitude e duração. Este sinal é muito utilizado nos modelos de detecção, pois contém evidências fisiológicas da ocorrência de AOS.

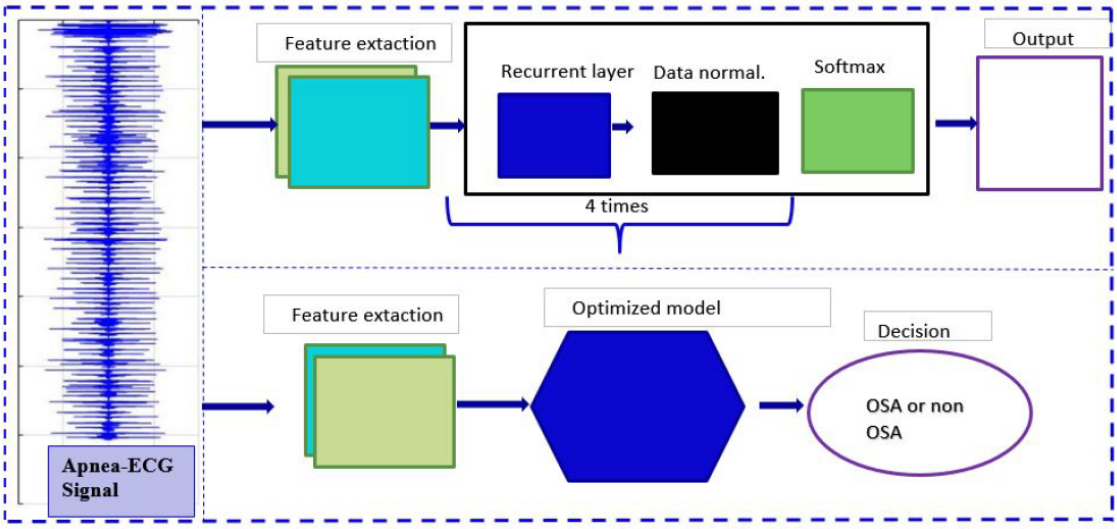
Chen, Zhang e Song (2015) propuseram uma abordagem utilizando o classificador *Support Vector Machine* (SVM) e utilizando a extração de *features* do sinal de ECG. Utilizando esta abordagem, obtiveram uma acurácia de 92.87%.

Por outro lado, os autores Song et al. (2016) exploram a dependência temporal existente na segmentação de sinais eletrocardiograma, onde é feita a detecção de QSR (movimentos que o coração faz a cada batida) e a correlação entre RR, através de uma etapa de pré-processamento dos exames. Após isto, são aplicados filtros neste sinal segmentado, e construídos frames de um minuto com a extração das *features*. Utilizando *Hidden Markov Model* (HMM), foi possível atingir uma classificação de 97.1% de acurácia.

Cheng et al. (2017) realizam uma extração de *features* do sinal de eletrocardiograma. Realizando a extração do intervalo RR, e utilizando essas *features* um modelo de *recurrent neural network* (RNN) e *batch normalization*, que consiste na normalização das camadas de input, visando um ganho de performance e estabilidade do modelo. A RNN utilizada foi a *Long short-term memory* (LSTM). Variando o número de épocas no treinamento e a taxa de aprendizado, os autores conseguiram atingir 97,80% de acurácia

nas detecções de apneia. Na Figura 3.1 está ilustrado o fluxo utilizado neste trabalho.

Figura 3.1: Visão do fluxo utilizado para modelo de detecção.



Fonte: Cheng et al. (2017)

Xie e Minn (2012) apresenta ótimos resultados com um modelo preditivo que combina a extração de *features* do sinal de saturação periférica de oxigênio (SpO2) e do eletrocardiograma (ECG). Utilizando 25 exames, e o método de *bagging* com *reduced-error pruning tree* (REPTree), obtiveram uma acurácia de 84,40%.

Tabela 3.1: Trabalhos relacionados

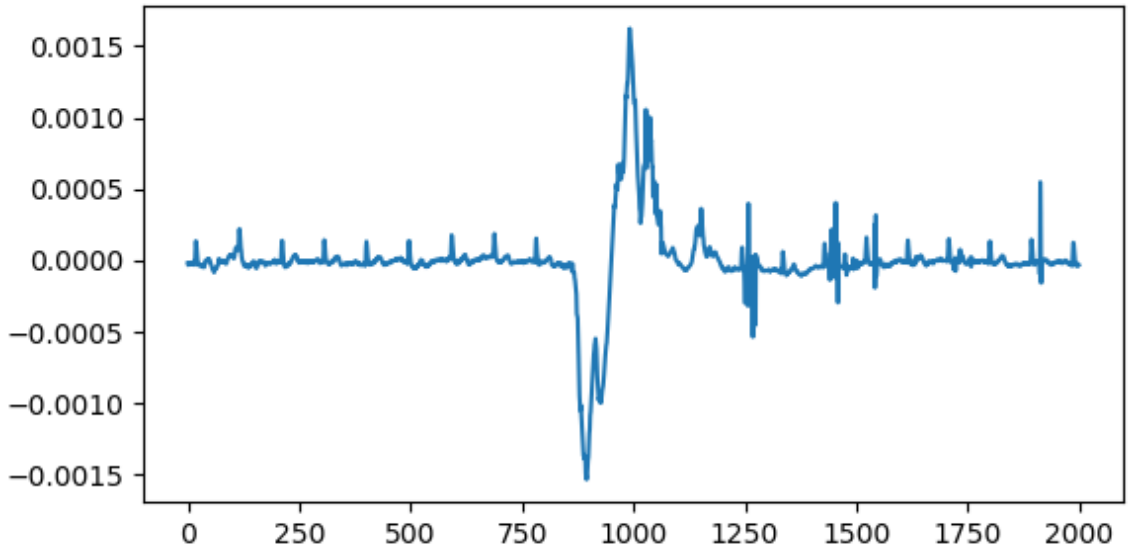
Título	Citações	Ano	Acurácia	Método	Relevância
Cheng et al.	34	2017	97,80%	Wavelet, RNN	4
Mostafa et al.	19	2017	97,64%	MLP	5
Chen; Zhang; Song	78	2015	97,41%	SVM	5
Sharma e Sharma	52	2016	97,14%	Hermite, KNN, MLPNN, SVM, LS-SVM	5
Song et al.	82	2015	97,10%	HMM	5
Almazaydeh, Elleithy e Faezipour	52	2012	96,50%	SVM	4
Wang, Lin e Wang	7	2019	94,40%	RNN	5
Rachim, Li e Chung	25	2014	94,30%	Wavelet, SVM	5
Acharya et al.	355	2017	94,03%	CNN	4
Hassan e Haque	68	2016	91,94%	RUSBoost	5
Kaguara, Nam e Reddy	12	2014	91,00%	DNN, SVM, KNN, NB	4
Sharma, Agarwal e Acharya	33	2018	90,11%	Wavelet, LS-SVM	4
Hassan e Haque	69	2015	85,97%	-	5
Biswal et al.	64	2017	85,76%	IRA	3
Li et al.	36	2018	85,00%	DNN, HMM	5
Varon et al.	145	2015	85,00%	SVM, RBF	4
Hassan	43	2015	83,77%	ELM	5
Jin e Dong	27	2015	83,66%	LCNN	3
Xie e Minn	185	2012	82,00%	REPTree	2
Zihlmann, Perekrestenko e Tschannen	97	2017	-	CNN, LSTM	3
Amiriparian et al.	126	2017	-	CNN	2
Hassan	33	2015	-	-	5
Rahhal et al.	359	2016	-	SDAEs, DNN	3

4 METODOLOGIA

Após verificação na literatura, foi possível observar a existência de muitos trabalhos explorando a utilização de redes neurais para realizar o diagnóstico de AOS, e obtendo resultados satisfatórios. Porém, a análise do sinal de eletrocardiograma é a abordagem mais comumente adotada devido à facilidade de captura deste sinal, e a existência de *datasets* gratuitos e bem documentados disponíveis online. Um dos repositórios mais utilizados pelos trabalhos da literatura é o PhysioNet (GOLDBERGER et al., 2000).

Ao analisar os sinais dos exames que nos foram disponibilizados, é possível observar a grande presença de ruídos devido à dificuldade de captação desses sinais. Na Figura 4.1, podemos observar um trecho de 20 segundos do sinal de ECG sem tratamento, extraído dos exames disponibilizados pela Clínica do Sono, onde existe um ruído muito grande que impossibilita a leitura do estado do paciente naquele momento. É comum a presença de ruídos neste tipo de captura, devido aos movimentos que o paciente faz durante o sono, eletrodos que podem cair durante o exame, e até mesmo o suor dos pacientes.

Figura 4.1: Trecho de 20 segundos de um sinal ECG *raw data* com ruído.



Analisando o sinal ECG do *dataset* disponibilizado no PhysioNet, e utilizado pelos autores Cheng et al. (2017), observa-se uma ocorrência muito menor de ruídos e inconsistências no sinal de ECG. Na Figura 4.2, podemos observar 4 segundos do sinal ECG analisado. É possível notar que a presença de ruídos nos sinais é quase nula e temos um sinal muito bem definido.

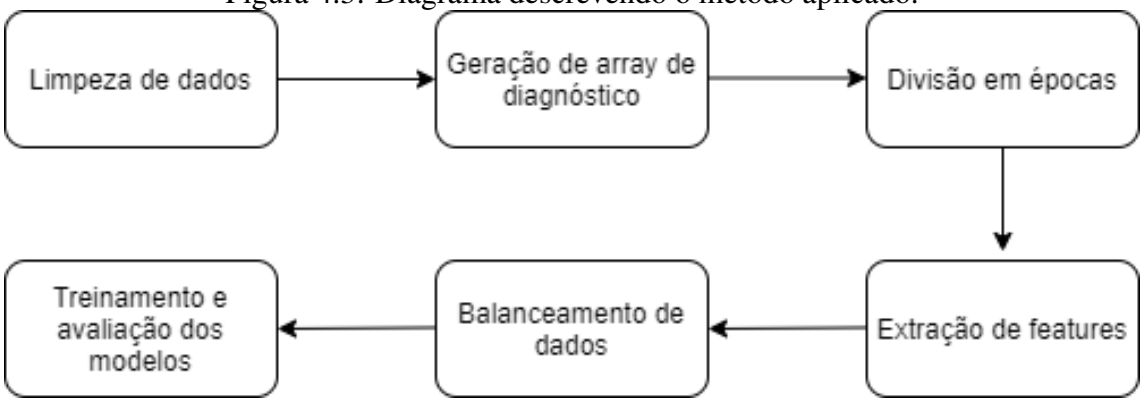
Figura 4.2: Trecho de 4 segundos de um sinal ECG extraído do PhysioNet.

Visto que nos exames de polissonografia que nos foram cedidos pela Clínica do Sono existe a presença de diversos canais, resolvemos explorar esses outros sinais que os demais trabalhos não estavam levando em consideração, e observar se haverá melhora ou não nos resultados obtidos. Utilizando múltiplos canais para realizar o diagnóstico, tentamos deixar os modelos utilizados robustos para que a presença de ruído em um canal não prejudique a performance do modelo.

Os sinais que serão utilizados como entrada para nosso modelo de rede neural estão associados a respiração do paciente e são os sinais analisados quando o diagnóstico é feito de forma manual, conforme instruções de Iber et al. (2007).

Conforme ilustrado na Figura 4.3, o fluxo construído é constituído pelas etapas de limpeza de dados, geração de *array* de diagnóstico, divisão em épocas, extração de *features*, balanceamento de dados e treinamento. As etapas serão descritas a seguir.

Figura 4.3: Diagrama descrevendo o método aplicado.



Fonte: O autor

4.1 Limpeza de dados

Nesta etapa é realizada a limpeza dos dados que nos foram disponibilizados, onde os sinais selecionados são devidamente preparados, deixando-os da melhor forma para se trabalhar nas etapas seguintes.

As informações dos sinais coletados no exame de polissonografia são armazenadas sem nenhum tratamento no formato EDF. Portanto foi necessário realizar a extração dos sinais selecionados do arquivo EDF, e os tratamentos necessários para obtenção de informação a partir dos dados. Os dados foram organizados de forma matricial, onde cada coluna representa sinal, e cada linha um instante de tempo, com intervalo de 10 ms.

Decidimos utilizar como entrada para os algoritmos preditivos os mesmos sinais utilizados no diagnóstico manual de AOS. Os sinais utilizados são fluxo aéreo, cinta abdominal, cinta torácica, saturação de oxigênio (SpO2) e frequência cardíaca.

Conforme orientação dos especialistas, foi removido do conjunto de dados as informações coletas antes do paciente dormir, e após o paciente acordar. Antes do paciente dormir, é feita uma etapa de calibragem dos equipamentos, que pode ser desconsiderada.

Como estamos utilizando exames reais, a presença de ruídos é um grande problema, que influencia diretamente nos resultados finais. Para reduzir o ruído presente nos dados, aplicamos o filtro Savitzky–Golay, que é como filtro de suavização de mínimos quadrados, o qual, segundo Uddin et al. (2016), é usado para suavizar funções do modo intrínseco de ruído dominante. Isto é realizado através do processo de convolução, ajustando subconjuntos sucessivos de pontos de dados adjacentes com um polinômio de baixo grau pelo método dos mínimos quadrados lineares.

Nos sinais do fluxo aéreo, cinta abdominal e cinta torácica também foi necessária uma etapa de normalização, tendo em vista que os sinais capturados estão na ordem de mV, e para utilizar estes dados nos algoritmos de redes neurais, precisamos de valores no intervalo de -1 a 1.

Nos sinais saturação de oxigênio (SpO2) e frequência cardíaca, realizamos um ajuste na magnitude do sinal para que os dados fiquem no intervalo entre 0 e 1.

4.2 Geração de array de diagnóstico

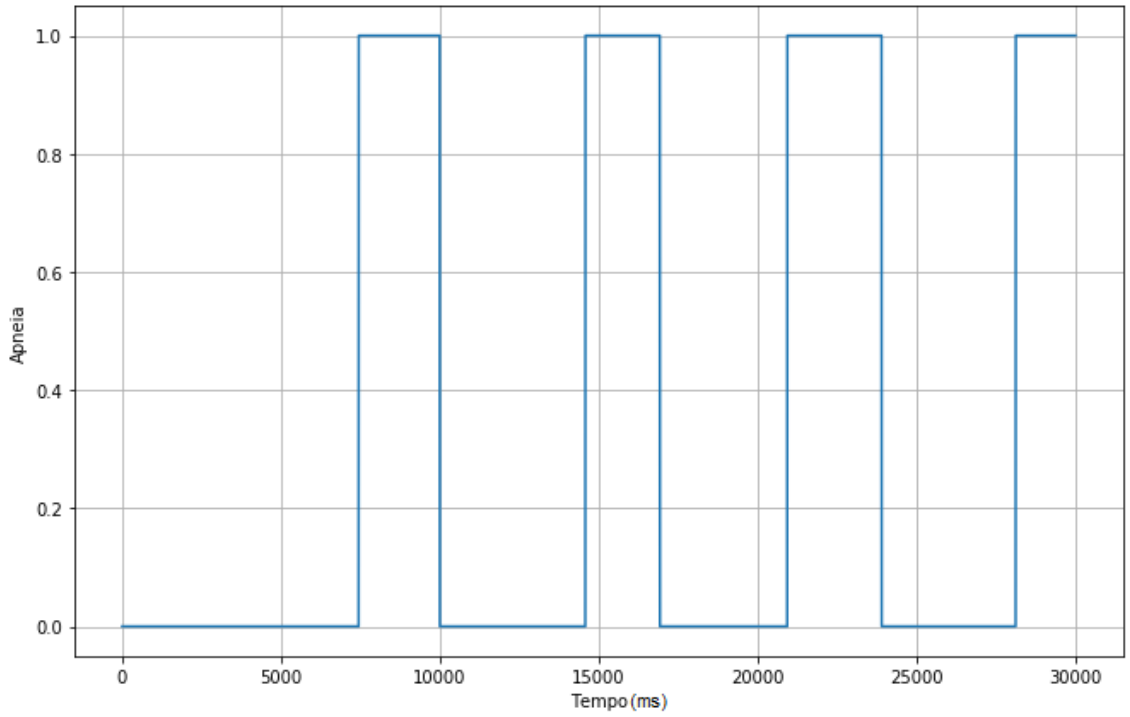
Em conjunto com os sinais dos exames, nos foi fornecido um arquivo de eventos, onde os técnicos da Clínica do Sono realizam uma série de anotações sobre o exame.

Dentre estas anotações, está a ocorrência de apneia, no qual utilizamos informações de tempo de início e duração dos eventos. Os eventos de apneia podem ser dos seguintes tipos:

- rera: despertar relacionado a esforço respiratório
- apnobst: apneia obstrutiva
- hipopn: hipoapneia
- apcent: apneia central

A partir deste arquivo de eventos, é criado um sinal binário de diagnóstico. Na Figura 4.4 está ilustrado 30 segundos do sinal de diagnóstico de apneia. Quando a onda está com valor 0 indica que o paciente não está sofrendo de apneia do sono, e com valor igual a 1, indica que está sofrendo. Este é o sinal que será usado como saída esperada para treinar a rede.

Figura 4.4: Sinal binário de diagnóstico



Fonte: O Autor

4.3 Divisão em épocas e extração de features

Após a limpeza dos sinais, e geração do *array* de diagnóstico, partimos para a etapa de divisão dos dados em intervalos menores, e extração de *features*.

Originalmente os sinais utilizados têm em média 8 horas de duração, que é o tempo que dura o exame de polissonografia na clínica que forneceu os dados. A partir da revisão na literatura, foi possível verificar que uma abordagem comum é a divisão dos sinais em intervalos menores, chamados de épocas, seguido de uma extração de *features*. O intervalo definido para as épocas varia nos trabalhos analisados. O autor Cheng et al. (2017), que utiliza o sinal de ECG, faz uma análise com base no intervalo RR. Um coração saudável tem entre 60 e 100 batidas por minuto.

Já Almazaydeh, Elleithy e Faezipour (2012) utilizou épocas de 10, 15 e 30 segundos, enquanto Xie e Minn (2012) analisou épocas de 60 segundos. Normalmente quando o diagnóstico é feito de forma manual, as janelas analisadas são intervalos de 10 segundos. Com base nas informações adquiridas nesses trabalhos, e nas sugestões do médico especialista da clínica do Sono, optou-se utilizar épocas de 2, 5 e 10 segundos.

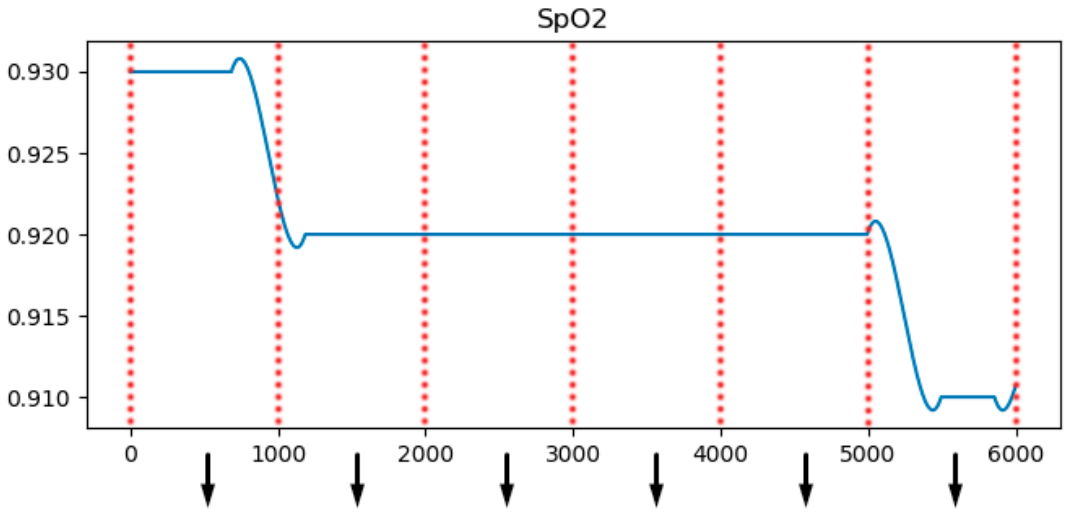
Após a divisão dos sinais em épocas menores, foi calculada a média, mediana e o desvio padrão desses intervalos, pois são *features* utilizadas na maioria dos trabalhos analisados.

Observando as *features* utilizadas por Xie e Minn (2012), resolvemos adicionar mais 5 *features* específicas do sinal de saturação do oxigênio (SpO2). O *Delta Index*, que consiste na soma do valor absoluto das diferenças entre as amostras do SpO2, dividido pela quantidade total de amostras dentro do intervalo delimitado. Outras *features* utilizadas foram as *ODIS2, ODIS3, ODIS4 e ODIS5*, que consiste no número total de amostras que estão 2%, 3%, 4% e 5% abaixo de um *baseline*. Esse *baseline* é calculado através da média dos primeiros 20% do exame. Com isso, totalizamos 20 *features*, que serão utilizadas como entrada para o treinamento da rede neural.

Para a saída (diagnostico), consideramos 1 caso exista a ocorrência de algum evento de apneia dentro da época analisada, e 0 se não houver nenhuma ocorrência.

Na Figura 4.5, está ilustrado a divisão de um trecho de 60 segundos, em épocas de 10 segundos, seguido da extração de *features*. No exemplo, estão sendo extraídos desvio padrão, média e mediana do sinal de SpO2.

Figura 4.5: Ilustração da divisão em épocas e extração de *features*.



SPO2_StdDev	0.0000	0.001807	0.000369	0.0000	0.0000	0.00000501
SPO2_Mean	0.9300	0.929354	0.920002	0.9200	0.9200	0.92000029
SPO2_Median	0.9300	0.9300	0.9200	0.9200	0.9200	0.920000

Fonte: O Autor

4.4 Balanceamento dos dados

Na Figura 4.6 pode ser observado que o número de amostras onde o paciente está sem AOS é muito maior que a quantidade de amostras onde o paciente está com AOS. A proporção é de 6,23 para 1, com os dados divididos em épocas de 5 segundos.

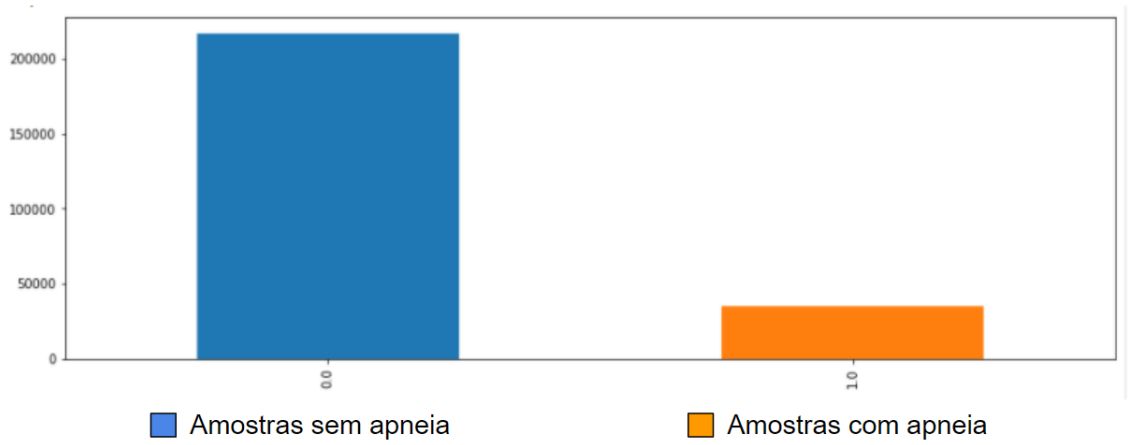
Segundo Tripathi, Batra e Pandey (2019), se treinarmos nossos modelos utilizando dados desbalanceados, podemos observar uma predição parcial da classe majoritária, causando uma acurácia enganosa. Dentre técnicas mais conhecidas para balanceamento de dados estão o *under-sampling* e o *over-sampling*. Utilizando *under-sampling*, as amostras da classe majoritária são reduzidas de forma aleatória para balancear o *dataset*. Já no *over-sampling*, aumentamos a quantidade de amostras da classe minoritária. Um dos algoritmos de *over-sampling* é o *Synthetic Minority Over-Sampling Technique* (SMOTE), que gera novas amostras da classe minoria, utilizando o algoritmo de *k-nearest neighbors*.

Também existem técnicas que combinam *over-sampling* e *under-sampling*, como o SMOTEENN. Nesse método, novas amostras da classe minoritária são geradas utilizando SMOTE, e o algoritmo *nearest neighbours* (ENN) é utilizado para limpeza de amostras da classe majoritária.

Visando uma melhora no desempenho dos algoritmos preditivos, foi realizada uma

etapa de balanceamento dos dados, onde a técnica SMOTEENN foi utilizada.

Figura 4.6: Proporção de amostras com épocas de 5 segundos



Fonte: O Autor

4.5 Treinamento e avaliação

Após a limpeza, filtragem e extração de *features*, chegamos finalmente à etapa de treinamento da rede neural. Para a construção do modelo preditivo, foi utilizada a linguagem Python 3 (ROSSUM; DRAKE, 2009), e a biblioteca Scikit-learn (PEDREGOSA et al., 2011). Para a construção do modelo preditivo, foi utilizado um *ensemble* de 10 redes neurais do tipo *Multilayer perceptron* (MLP). Para realizar o treinamento dos modelos, utilizamos o algoritmo de *Bagging*.

Em cada MLP que compõe o *ensemble*, a função de otimização utilizada foi a *Stochastic gradient descent* (SGD), devido a sua eficiência ao lidar com grande volume de dados. Quando comparamos a função de otimização SGD com a função de otimização *Adam*, que é uma versão otimizada da SGD, a *Adam* mostrou-se mais rápida, mas a SGD obteve uma capacidade de generalização maior. A função de ativação utilizada nas camadas ocultas foi a *Relu*, pois se mostra mais rápida e efetiva em grande volume de dados, quando comparada com a função sigmoide, por exemplo. Já a tolerância de otimização utilizada foi de $1x10^{-9}$. O número máximo de iterações definido foi de 2000. O parâmetro Alpha (termo de regularização) utilizado foi o padrão, de 0.0001.

Para o treinamento do protótipo, foram utilizados 52 exames de polissonografia, os quais nos foram cedidos pela Clínica do Sono. Desses 52 pacientes, 35 São homens com média de idade de 41,86 e desvio padrão de 14,17 anos; 17 são mulheres com média

de idade de 42,53 anos e desvio padrão de 18,97. Desses 52 exames, 3 foram removidos do *dataset* para uma futura etapa de validação. Os 49 exames restantes foram divididos em 2 grupos, onde 70% dos dados foram utilizados para o treinamento do modelo, e os 30% restantes foram utilizados posteriormente na validação do modelo treinado.

Quando o conjunto de dados foi particionado utilizando épocas de 2 segundos, são geradas 628.792 amostras (552.509 da classe sem apneia, e 76.283 da classe com apneia), ao utilizar épocas de 5 segundos são geradas 251.516 amostras (216.745 da classe sem apneia, e 34.771 da classe com apneia), e ao utilizar épocas de 10 segundos, são geradas 125.758 amostras (105.167 da classe sem apneia, e 20.591 da classe com apneia).

Para avaliação dos modelos gerados, foram utilizadas as métricas de *F1-score* e acurácia da classe positiva.

4.6 Validação

Após a conclusão do treinamento e teste, foi selecionada a rede neural que apresentou melhores resultados de acurácia e F1-Score para diagnosticar os três exames que são desconhecidos pela rede (deixados a parte, conforme descrito na seção anterior), para validar sua capacidade de generalização.

Nesses exames há um caso com apneia normal (até 5 ocorrências), um caso de apneia leve (de 5 à 15 ocorrências), e um caso de apneia grave (mais de 30 ocorrências).

5 EXPERIMENTOS

Nesta seção serão descritos os experimentos realizados para realizar o treinamento e avaliação dos modelos. Após encontrar o modelo com melhor desempenho, o mesmo foi utilizado para realizar a predição de 3 exames que não foram utilizados no conjunto de dados de teste e treino.

5.1 Treinamento e avaliação

O treinamento dos modelos foi feito de duas formas: com dados desbalanceados e balanceados. Foi feita uma comparação nos resultados obtidos para validar se a etapa de balanceamento dos dados resulta em modelos com predições mais acuradas.

5.1.1 Dados desbalanceados

Para remover um possível viés dos modelos utilizados nos experimentos, utilizou-se um método de *ensemble*. Dessa forma, aumentou-se a acurácia e a capacidade de generalização do modelo preditivo. O método de *ensemble* utilizado foi um *Bagging Classifier*, combinando 10 modelos de MLP. Como métrica para avaliação dos modelos gerados, foi utilizado a acurácia e o *F1-Score*. Para encontrar a melhor configuração das MLPs que compõe o *ensemble*, experimentamos 14 modelos diferentes, variando o número de camadas, e neurônios em cada camada. Para o número de neurônios em cada camada, foi utilizado números múltiplos da quantidade de *features* utilizadas. O número máximo de camadas que utilizamos foi 5, devido à limitação de processamento e tempo de treinamento das redes, que nos modelos mais complexos chegou a levar em torno de 8 horas.

O treinamento foi realizado utilizando os 3 conjunto de dados diferentes. O primeiro foi gerado dividindo os dados dos exames em épocas de 2 segundos na etapa de extração de *features*, o segundo utilizando intervalos de 5 segundos e o terceiro de 10 segundos. Avaliamos os resultados obtidos utilizando cada um desses conjuntos de dados para definir qual o melhor intervalo a ser utilizado.

Na tabela 5.1 podemos verificar os resultados encontrados nos treinamentos utilizando os 3 intervalos diferentes. Estão destacados em negrito a maior acurácia e F1-Score

obtidos em cada intervalo. O modelo que atingiu o melhor resultado foi um *ensemble* com 10 MLPs com 3 camadas ocultas, com 20, 80 e 120 neurônios, respectivamente, que atingiu um F1-Score de 34,27%.

Tabela 5.1: Tabela com resultados de acurácia e F1-Score avaliando os dados de teste desbalanceados.

	Intervalos 2 segundos		Intervalos 5 segundos		Intervalos 10 segundos	
Camadas	Acuracia	F1-Score	Acuracia	F1-Score	Acuracia	F1-Score
20,40	84,79%	25,41%	84,81%	26,66%	83,39%	28,55%
40,20	85,18%	24,65%	84,13%	27,17%	83,27%	28,64%
60,20	84,98%	24,23%	84,93%	25,55%	82,82%	31,18%
80,20	**85,43%**	24,16%	**85,69%**	22,64%	82,74%	30,70%
100,20	84,57%	25,04%	83,49%	27,08%	81,45%	33,99%
100,40	83,50%	**26,58%**	84,17%	25,60%	83,06%	28,65%
20,40,100	84,60%	24,66%	83,77%	26,33%	**83,47%**	27,76%
20,40,120	84,94%	25,09%	83,27%	26,81%	82,97%	30,19%
20,40,140	84,21%	25,89%	83,39%	**28,03%**	83,38%	27,43%
20,80,120	84,93%	23,76%	82,99%	27,37%	81,10%	**34,27%**
20,40,20,15	84,93%	24,61%	82,31%	26,33%	83,24%	28,76%
20,60,20,15	84,46%	25,76%	81,61%	27,72%	81,44%	32,48%
20,40,100,40	84,16%	25,24%	81,70%	27,32%	82,56%	30,23%
20,20,20,20,20	84,25%	25,71%	82,68%	26,64%	82,65%	30,66%

5.1.2 Dados Balanceados

Para resolução do problema de desbalanceamento dos dados, utilizou-se a técnica de *over-sampling* somada com a técnica de *under-sampling*. Combinando essas duas técnicas, atingimos o objetivo de obter um conjunto de dados balanceados. Para tanto, foi utilizada a biblioteca imbalanced-learn (LEMAîTRE; NOGUEIRA; ARIDAS, 2017), onde utilizamos a função SMOTEENN, que combina o método SMOTE para fazer *over-sampling* das amostras, e posteriormente realiza uma limpeza utilizando o método *Edited Nearest Neighbours* (ENN).

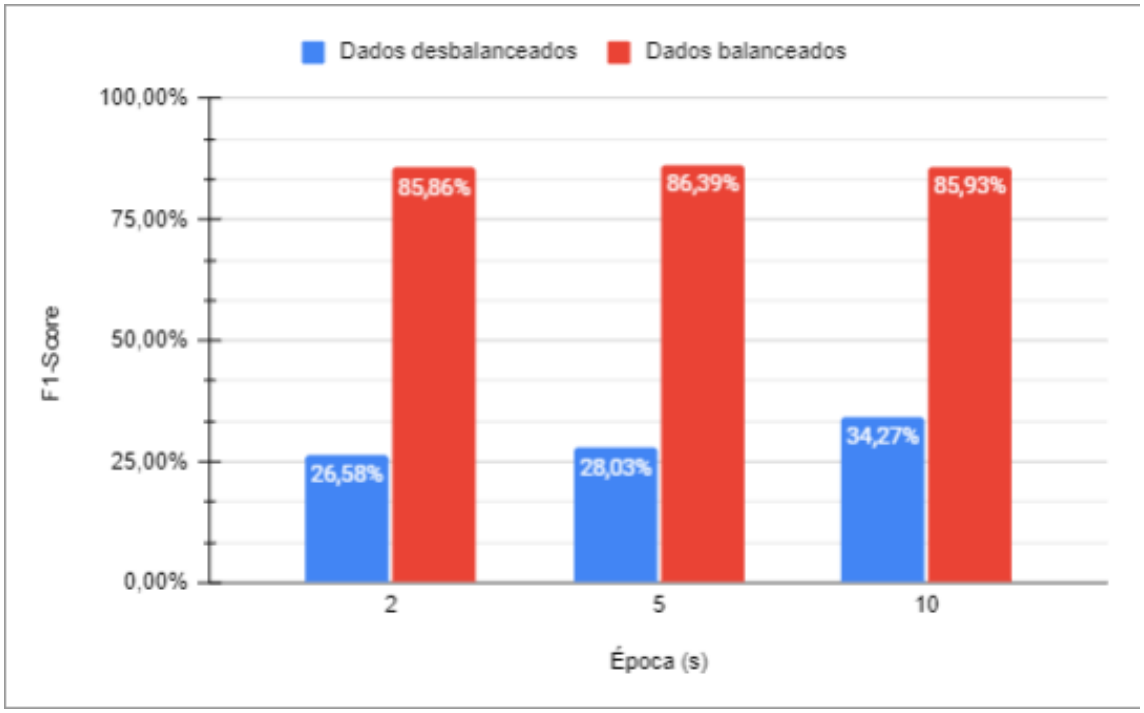
Após o balanceamento dos dados, realizamos novamente os treinamentos dos modelos, percebemos uma melhora notável nas métricas de avaliação. Na tabela 5.2 podemos verificar os resultados encontrados nos treinamentos utilizando os 3 intervalos diferentes. Estão destacados em negrito a maior acurácia e F1-Score obtidos em cada intervalo. O modelo que atingiu o melhor resultado foi um *ensemble* com 10 MLPs com 3 camadas ocultas de 20, 80 e 120 neurônios, que atingiu um F1-Score de 86.39%.

Tabela 5.2: Tabela com resultados de acurácia e F1-Score avaliando os dados balanceados.

	Intervalos 2 segundos		Intervalos 5 segundos		Intervalos 10 segundos	
Camadas	Acuracia	F1-Score	Acuracia	F1-Score	Acuracia	F1-Score
20,40	72,60%	84,12%	69,09%	81,72%	59,61%	74,70%
40,20	70,30%	82,56%	67,65%	80,70%	62,86%	77,19%
60,20	70,65%	82,80%	66,24%	79,69%	64,07%	78,10%
80,20	69,42%	81,95%	69,75%	82,18%	66,07%	79,57%
100,20	74,27%	85,24%	70,85%	82,94%	67,06%	80,29%
100,40	69,38%	81,92%	72,31%	83,93%	68,76%	81,49%
20,40,100	73,09%	84,45%	73,18%	84,51%	68,04%	80,98%
20,40,120	70,85%	82,94%	71,75%	83,55%	61,87%	76,45%
20,40,140	69,43%	81,96%	71,18%	83,16%	67,84%	80,84%
20,80,120	73,08%	84,45%	**76,05%**	**86,39%**	70,41%	82,63%
20,40,20,15	71,62%	83,46%	73,67%	84,84%	64,27%	78,25%
20,60,20,15	**75,23%**	**85,86%**	74,33%	85,27%	**75,33%**	**85,93%**
20,40,100,40	74,04%	85,09%	75,66%	86,15%	70,96%	83,02%
20,20,20,20,20	72,36%	83,97%	72,32%	83,94%	70,10%	82,42%

Na Figura 5.1 pode-se observar a grande diferença no F1-Score obtido após aplicar a etapa de balanceamento dos dados. Na imagem, foi comparado o F1-Score do melhor modelo gerado em cada intervalo, antes e depois do balanceamento dos dados ser realizado.

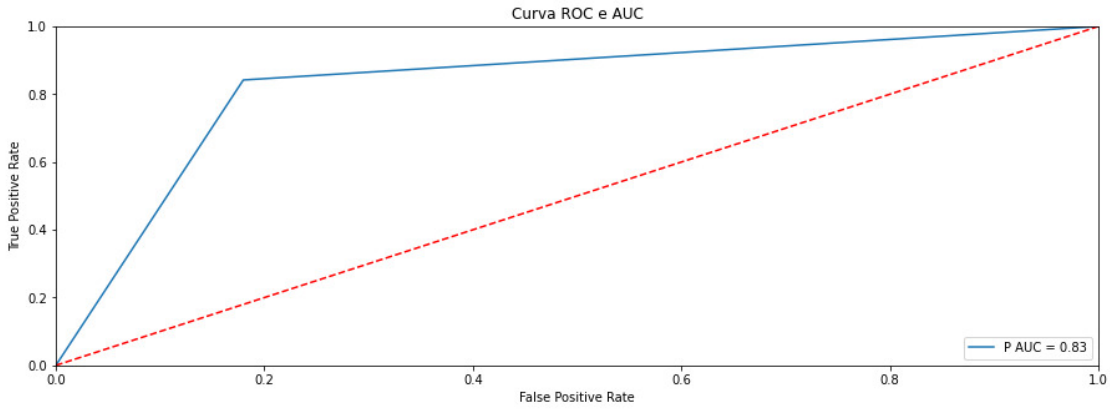
Figura 5.1: Comparação do F1-Score após o balanceamento dos dados



Fonte: O Autor

A curva *Receiver Operating Characteristic*(ROC) é uma ferramenta comum de análise de permite estudar a variação da sensibilidade e especificidade, para diferentes valores de corte. Na curva ROC, temos a representação gráfica da sensibilidade (taxa de verdadeiros positivos) versus 1-especificidade (taxa de falsos positivos). Na imagem 5.2 pode-se observar a curva ROC gerada pela rede que atingiu maior F1-Score. A área embaixo da curva (AOC) é de 0,83.

Figura 5.2: Curva ROC



5.2 Predições

Após identificar a arquitetura que atingiu o maior F1-Score, realizamos a predição de 3 exames que não foram utilizados nas etapas de treinamento e teste. Nestes exames, estão um caso de apneia leve, moderada e grave. Foram avaliados a acurácia, F1-Score, *recall* e *precision* de cada predição realizada.

Na Tabela 5.3, podemos observar que os valores atingidos ao realizar a predição dos exames. No exame com apneia leve, atingimos uma acurácia alta, de 71,63%, porém um F1-Score muito baixo, apenas 13,50%. Pode observar que a *precision* está muito baixo, aproximadamente 8%, e isso indica que a presença de falsos negativos é muito grande. Um *recall* de 45%, indica que praticamente metade dos casos de apneia foram diagnosticados incorretamente. Na tabela 5.4 podemos observar a matriz de confusão gerada.

Já na predição do exame com apneia moderada, a acurácia e o *recall* se mantiveram na mesma faixa do exame com apneia leve, enquanto o F1-Score e a *precision* praticamente dobraram, atingindo 26% e 18% respectivamente. Na tabela 5.5 podemos observar a matriz de confusão gerada.

No exame de apneia grave podemos notar uma queda na acurácia, e um ganho em todos os outros indicadores. O *precision* atingiu 43,83%, e isto indica que quase metade das vezes em que o modelo fez um diagnóstico de apneia, ele estava correto. O *recall* atingiu 78,63%, que indica que aproximadamente 4 a cada 5 eventos de apneia foram detectados. O F1-Score atingiu 56,28%, o melhor desempenho entre os 3 exames. Na tabela 5.6 podemos observar a matriz de confusão gerada.

Analisando os 3 diagnósticos, percebemos que as predições são mais corretas para os casos onde há maior ocorrências de apneia. O caso com menor F1-Score é o de apneia leve, e consequentemente o F1-Score mais alto é o exame com apneia grave. Esse comportamento pode ser resultado do balanceamento dos dados, onde geramos muitas amostras de ocorrência de apneia, e retiramos amostras onde não há ocorrência de apneia.

Tabela 5.3: Resultados das predições

	Acurácia	F1-Score	Precision	Recall
Apneia leve	72,79%	13,33%	7,94%	41,67%
Apneia moderada	75,39%	26,20%	19,76%	38,86%
Apneia grave	60,72%	55,67%	42,01%	82,51%

Tabela 5.4: Matriz de confusão da previsão do exame com apneia leve

Real	Previsto	
	Com apneia	Sem apneia
Com apneia	105	147
Sem apneia	1218	3546

Tabela 5.5: Matriz de confusão da previsão do exame com apneia moderado.

Real	Previsto	
	Com apneia	Sem apneia
Com apneia	232	365
Sem apneia	942	3771

Tabela 5.6: Matriz de confusão da previsão do exame com apneia grave.

Real	Previsto	
	Com apneia	Sem apneia
Com apneia	1382	293
Sem apneia	1908	2021

6 COMPARAÇÃO

Nesta seção vamos comparar os resultados obtidos neste trabalho, com o trabalho realizado pelo colega Silva (2020), que para a resolução deste problema, propôs métodos de aprendizado de máquina baseado em modelos estatísticos.

Foram utilizados o método de treinamento *ensemble AdaBoost*, e modelos do tipo *Support Vector Machines* (SVM). Também foi realizada o balanceamento dos dados, e analise se esta etapa gerou melhora na performance dos modelos. As 5 *features* geradas a partir do sinal de SpO2 (Delta Index, ODIS2, ODIS3, ODIS4 e ODIS5) não foram utilizadas para treinamento e avaliação dos modelos. Além disso foram 5 épocas diferentes: 2, 5, 10, 15 e 30 segundos.

O método de balanceamento de dados utilizado foi o *under-sampling*. Um ponto importante a ser destacado é a diferença de tempo de treinamento entre os métodos. Em alguns casos de arquiteturas mais complexas de *multilayer-perceptron*, o tempo de treinamento chegou a ser 20 vezes maior ao tempo de treinamento do algoritmo SVM.

Nas tabelas 6.1, 6.2, 6.3, 6.4, 6.5 podemos observar os resultados obtidos através dos experimentos realizados.

Tabela 6.1: Avaliação com *Train-Test* - Época de 2 segundos.

Modelo	Acurácia	*Recall*	*Precision*	*F1 Score*
SVM com dados desbalanceados	76%	20%	21%	20%
SVM com *Under-Sampling*	53%	63%	18%	28%
ADA Boost com dados desbalanceados	83%	25%	45%	32%
Ada Boost com *Under-Sampling*	65%	53%	71%	59%

Fonte: (SILVA, 2020)

Tabela 6.2: Avaliação com *Train-Test* - Época de 5 segundos.

Modelo	Acurácia	*Recall*	*Precision*	*F1 Score*
SVM com dados desbalanceados	75%	21%	22%	21%
SVM com *Under-Sampling*	41%	51%	13%	21%
Ada Boost com dados desbalanceados	84%	16%	49%	35%
Ada Boost com *Under-Sampling*	64%	62%	25%	35%

Fonte: (SILVA, 2020)

Tabela 6.3: Avaliação com *Train-Test* - Época de 10 segundos.

Modelo	Acurácia	*Recall*	*Precision*	*F1 Score*
SVM com dados desbalanceados	73%	24%	26%	25%
SVM com *Under-Sampling*	63%	47%	68%	55%
Ada Boost com dados desbalanceados	81%	25%	53%	34%
Ada Boost com *Under-Sampling*	63%	52%	69%	58%

Fonte: (SILVA, 2020)

Tabela 6.4: Avaliação com *Train-Test* - Época de 15 segundos.

Modelo	Acurácia	*Recall*	*Precision*	*F1 Score*
SVM com dados desbalanceados	73%	27%	31%	29%
SVM com *Under-Sampling*	63%	47%	70%	52%
Ada Boost com dados desbalanceados	80%	32%	54%	40%
Ada Boost com *Under-Sampling*	64%	51%	68%	57%

Fonte: (SILVA, 2020)

Tabela 6.5: Avaliação com *Train-Test* - Época de 30 segundos.

Modelo	Acurácia	*Recall*	*Precision*	*F1 Score*
SVM com dados desbalanceados	62%	19%	96%	32%
SVM com *Under-Sampling*	62%	41%	67%	49%
Ada Boost com dados desbalanceados	73%	47%	92%	62%
Ada Boost com *Under-Sampling*	62%	46%	64%	51%

Fonte: (SILVA, 2020)

Na Figura 6.1 é possível observar a comparação dos modelos desenvolvidos neste trabalho, com os modelos desenvolvidos por Silva (2020). Foi feita uma comparação das épocas de 2, 5 e 10 segundos. É possível observar que, com dados desbalanceados, os modelos de redes neurais e modelos estatísticos apresentam praticamente os mesmos resultados. A época de 10 segundos apresentou o melhor resultado, onde o F1-Score ficou aproximadamente 34% com os dois métodos.

Já na Figura 6.2 é possível observar que após a etapa de balanceamento dos dados, os modelos de redes neurais apresentaram um F1-Score superior. Enquanto SVM atinge

um máximo de 59%, utilizando MLP foi possível atingir um F1-Score de 86%.

Figura 6.1: Comparação de modelos com treinamento com dados desbalanceados.

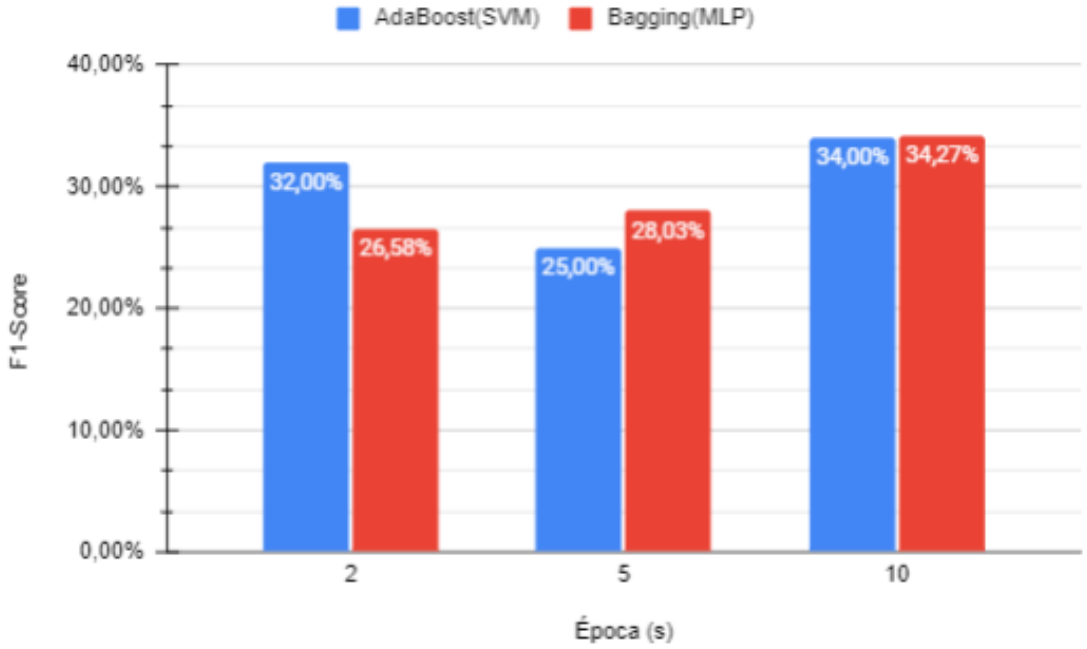
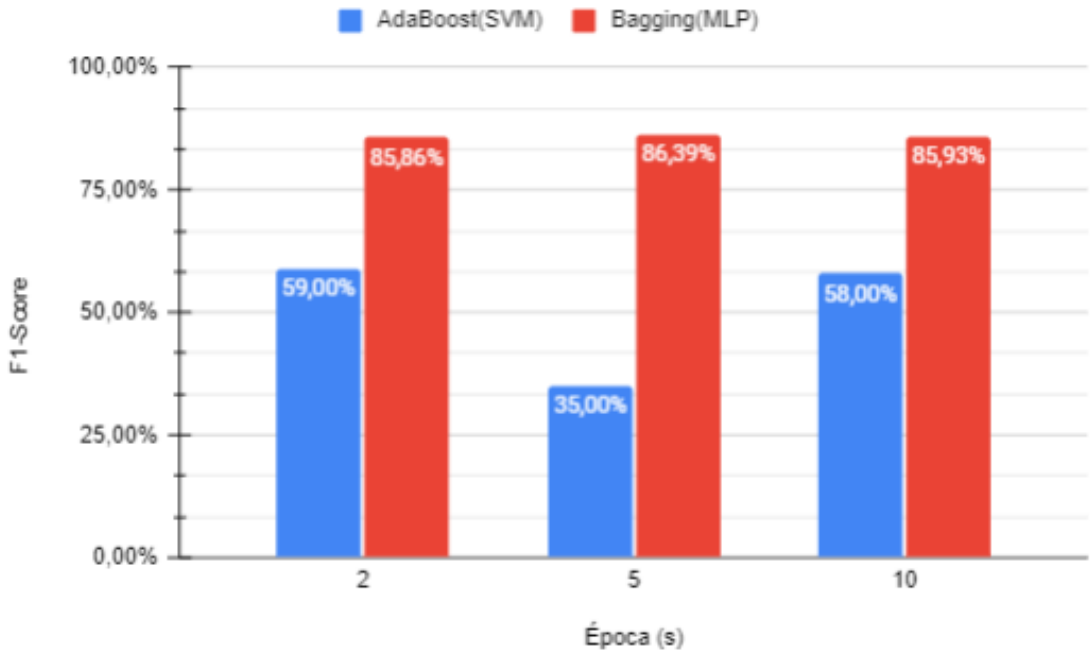


Figura 6.2: Comparação de modelos com treinamento com dados balanceados.



7 CONCLUSÃO

O diagnóstico de AOS é um processo que demanda muito tempo, é cansativo e está suscetível a erros humanos. Para realizar o diagnóstico mais rapidamente e de forma mais confiável, o uso de métodos computacionais de reconhecimento de padrões surge como uma alternativa. Neste trabalho, foi verificado se utilizar sinais ligados a respiração do paciente, que são os sinais utilizados quando o diagnóstico é feito manualmente, consegue-se reconhecer os padrões gerados pela AOS através de modelos de redes neurais. Os modelos foram avaliados através de métricas de acurácia e F1-Score.

Os resultados obtidos mostram que o tratamento correto dos dados, escolha adequada do intervalo a ser analisado e o balanceamento de amostras feito através da técnica SMOTEENN melhoraram os resultados dos modelos, atingindo um F1-Score de 86.39%.

Pode-se concluir também que a escolha dos sinais e *features* utilizados é de suma importância para construir um modelo com uma boa capacidade preditiva. Apesar da utilização de sinais que não foram explorados no trabalhos relacionados encontrados na literatura, como cinta abdominal, torácica e fluxo aéreo, não houve uma melhora no desempenho dos algoritmos.

Apesar da utilização de algumas técnicas extraídas dos trabalhos relacionados, o uso de uma fonte de dados diferente torna a comparação com os resultados obtidos na literatura difícil de ser feita. Seria interessante criar uma adaptação do método apresentado neste trabalho, para realizar uma comparação direta com os resultados da literatura, utilizando os mesmos datasets.

A utilização de outras métricas como o F1-Score utilizando a média macro e a média micro para uma comparação também é interessante para entender melhor os modelos gerados. Outro ponto interessante a ser explorado é o uso de técnicas para lidar com problemas de dados desbalanceados, como impor um custo adicional quando o modelo realizar classificações erradas da classe minoritária. Outra comparação interessante à ser feita é entre o modelo proposto neste trabalho, e modelos *AdaBoost* e *SVM* conforme proposto pelo Silva (2020), porém utilizando a técnica de SMOTEENN para balanceamento dos dados.

Trabalhos de pesquisa desse tipo na área da saúde têm uma importância fundamental para tornar exames e tratamentos economicamente viáveis. Avanços na forma de coleta de sinais e diagnóstico de AOS abrem caminhos para a realização de um diagnóstico e tratamento precoce, melhorando a qualidade de vida e saúde da população.

REFERÊNCIAS

ACHARYA, U. R. et al. A deep convolutional neural network model to classify heartbeats. **Computers in Biology and Medicine**, v. 89, 08 2017.

ALMAZAYDEH, L.; ELLEITHY, K.; FAEZIPOUR, M. Obstructive sleep apnea detection using svm-based classification of ecg signal features. In: IEEE. **2012 annual international conference of the IEEE engineering in medicine and biology society**. [S.l.], 2012. p. 4938–4941.

AMIRIPARIAN, S. et al. Snore sound classification using image-based deep spectrum features. In: **INTERSPEECH**. [S.l.: s.n.], 2017. v. 434, p. 3512–3516.

BISWAL, S. et al. Sleepnet: automated sleep staging system via deep learning. **arXiv preprint arXiv:1707.08262**, 2017.

BREIMAN, L. Bagging predictors. **Machine Learning**, v. 24, p. 123–140, 1996.

CAVALLARI, F. E. et al. Relação entre hipertensão arterial sistêmica e síndrome da apnéia obstrutiva do sono. **Rev Bras Otorrinolaringol**, SciELO Brasil, v. 68, n. 5, p. 619–22, 2002.

Chen, L.; Zhang, X.; Song, C. An automatic screening approach for obstructive sleep apnea diagnosis based on single-lead electrocardiogram. **IEEE Transactions on Automation Science and Engineering**, v. 12, n. 1, p. 106–115, Jan 2015.

Cheng, M. et al. Recurrent neural network based classification of ecg signal features for obstruction of sleep apnea detection. In: **2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)**. [S.l.: s.n.], 2017. v. 2, p. 199–202. ISSN null.

DALTRO, C. H. et al. Síndrome da apnéia e hipopnéia obstrutiva do sono: associação com obesidade, gênero e idade. **Arquivos Brasileiros de Endocrinologia & Metabologia**, SciELO Brasil, v. 50, n. 1, p. 74–81, 2006.

FOSTER, K. R.; KOPROWSKI, R.; SKUFCA, J. D. Machine learning, medical diagnosis, and biomedical engineering research-commentary. **Biomedical engineering online**, Springer, v. 13, n. 1, p. 94, 2014.

GARDNER, M. W.; DORLING, S. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. In: . [S.l.]: Elsevier, 1998. v. 32, n. 14-15, p. 2627–2636.

GHARIB, M.; BONDAVALLI, A. On the evaluation measures for machine learning algorithms for safety-critical systems. In: IEEE. **2019 15th European Dependable Computing Conference (EDCC)**. [S.l.], 2019. p. 141–144.

GOLDBERGER, A. L. et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. **Circulation**, v. 101, n. 23, p. e215–e220, 2000. Circulation Electronic Pages: http://circ.ahajournals.org/content/101/23/e215.full PMID:1085218; doi: 10.1161/01.CIR.101.23.e215.

GUTTA, S.; CHENG, Q. Modeling of oxygen saturation and respiration for sleep apnea detection. In: IEEE. **2016 50th Asilomar Conference on Signals, Systems and Computers**. [S.l.], 2016. p. 1636–1640.

GUTTA, S. et al. Cardiorespiratory model-based data-driven approach for sleep apnea detection. **IEEE journal of biomedical and health informatics**, IEEE, v. 22, n. 4, p. 1036–1045, 2017.

HANSEN, L.; SALAMON, P. Neural network ensembles. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, v. 12, p. 993 – 1001, 11 1990.

HASSAN, A. R. Automatic screening of obstructive sleep apnea from single-lead electrocardiogram. In: IEEE. **2015 international conference on electrical engineering and information communication technology (ICEEICT)**. [S.l.], 2015. p. 1–6.

HASSAN, A. R. A comparative study of various classifiers for automated sleep apnea screening based on single-lead electrocardiogram. In: IEEE. **2015 International Conference on Electrical & Electronic Engineering (ICEEE)**. [S.l.], 2015. p. 45–48.

HASSAN, A. R.; HAQUE, M. A. Computer-aided obstructive sleep apnea screening from single-lead electrocardiogram using statistical and spectral features and bootstrap aggregating. **Biocybernetics and Biomedical Engineering**, Elsevier, v. 36, n. 1, p. 256–266, 2016.

HASSAN, A. R.; HAQUE, M. A. An expert system for automated identification of obstructive sleep apnea from single-lead ecg using random under sampling boosting. **Neurocomputing**, Elsevier, v. 235, p. 122–130, 2017.

IBER, C. et al. The aasm manual for the scoring of sleep and associated events: Rules, terminology and technical specifications. **Westchester, IL: American Academy of Sleep Medicine**, 01 2007.

JIN, L.; DONG, J. Deep learning research on clinical electrocardiogram analysis. **Scientia Sinica Informationis**, Science China Press, v. 45, n. 3, p. 398–416, 2015.

KAGUARA, A.; NAM, K. M.; REDDY, S. A deep neural network classifier for diagnosing sleep apnea from ecg data on smartphones and small embedded systems. **BA Computer Science**, 2014.

KEMP, B.; OLIVAN, J. European data format 'plus'(edf+), an edf alike standard format for the exchange of physiological data. **Clinical neurophysiology**, Elsevier, v. 114, n. 9, p. 1755–1761, 2003.

KLOCK, A. C. T. Mapeamentos e revisões sistemáticos da literatura: um guia teórico e prático. **Cadernos de Informática**, v. 10, n. 1, p. 01–09, 2018.

LEMAîTRE, G.; NOGUEIRA, F.; ARIDAS, C. K. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. **Journal of Machine Learning Research**, v. 18, n. 17, p. 1–5, 2017. Available from Internet: <http://jmlr.org/papers/v18/16-365.html>.

LI, K. et al. A method to detect sleep apnea based on deep neural network and hidden markov model using single-lead ecg signal. **Neurocomputing**, Elsevier, v. 294, p. 94–101, 2018.

Lippmann, R. An introduction to computing with neural nets. **IEEE ASSP Magazine**, v. 4, n. 2, p. 4–22, Apr 1987.

LUGER, G. Artificial intelligence: Structures and strategies for complex problem-solving 4rd edition. 01 2005.

MAHMUD, M. et al. Applications of deep learning and reinforcement learning to biological data. **IEEE transactions on neural networks and learning systems**, IEEE, v. 29, n. 6, p. 2063–2079, 2018.

MALSBURG, C. V. D. Frank rosenblatt: principles of neurodynamics: perceptrons and the theory of brain mechanisms. In: **Brain theory**. [S.l.]: Springer, 1986. p. 245–248.

MAURER, J. T. Early diagnosis of sleep related breathing disorders. **GMS current topics in otorhinolaryngology, head and neck surgery**, German Medical Science, v. 7, 2008.

MOSTAFA, S. S. et al. Spo2 based sleep apnea detection using deep learning. In: IEEE. **2017 IEEE 21st international conference on intelligent engineering systems (INES)**. [S.l.], 2017. p. 000091–000096.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

POMBO, N.; GARCIA, N.; BOUSSON, K. Classification techniques on computerized systems to predict and/or to detect apnea: A systematic review. **Computer methods and programs in biomedicine**, Elsevier, v. 140, p. 265–274, 2017.

RACHIM, V. P.; LI, G.; CHUNG, W.-Y. Sleep apnea classification using ecg-signal wavelet-pca features. **Bio-medical materials and engineering**, v. 24, p. 2875–82, 09 2014.

RAHHAL, M. M. A. et al. Deep learning approach for active classification of electrocardiogram signals. **Information Sciences**, Elsevier, v. 345, p. 340–354, 2016.

RAVELO-GARCIA, A. et al. Application of the permutation entropy over the heart rate variability for the improvement of electrocardiogram-based sleep breathing pause detection. **Entropy**, v. 17, p. 914–927, 03 2015.

ROSSUM, G. V.; DRAKE, F. L. **Python 3 Reference Manual**. Scotts Valley, CA: CreateSpace, 2009. ISBN 1441412697.

RUCK, D. W.; ROGERS, S. K.; KABRISKY, M. Feature selection using a multilayer perceptron. **Journal of Neural Network Computing**, v. 2, n. 2, p. 40–48, 1990.

SHARMA, H.; SHARMA, K. An algorithm for sleep apnea detection from single-lead ecg using hermite basis functions. **Computers in Biology and Medicine**, v. 77, 08 2016.

SHARMA, M.; AGARWAL, S.; ACHARYA, U. R. Application of an optimal class of antisymmetric wavelet filter banks for obstructive sleep apnea diagnosis using ecg signals. **Computers in biology and medicine**, Elsevier, v. 100, p. 100–113, 2018.

SILVA, R. Bortoluzzi da. **Detecção de Apneia do Sono Utilizando Machine Learning Baseado em Modelos Estatísticos**. 46 p. Monografia (Bacharel em Engenharia da Computação) — Universidade Federal do Rio Grande do Sul, Porto Alegre, 2020.

Song, C. et al. An obstructive sleep apnea detection approach using a discriminative hidden markov model from ecg signals. **IEEE Transactions on Biomedical Engineering**, v. 63, n. 7, p. 1532–1542, July 2016.

ŠTER, B.; DOBNIKAR, A. Neural networks in medical diagnosis: Comparison with other methods. In: **International conference on engineering applications of neural networks**. [S.l.: s.n.], 1996. p. 427–30.

TRIPATHI, S.; BATRA, S.; PANDEY, S. Unbiased mortality prediction for unbalanced data using machine learning. In: IEEE. **2019 International Conference on Electrical, Electronics and Computer Engineering (UPCON)**. [S.l.], 2019. p. 1–5.

UDDIN, M. B. et al. A new machine learning approach to select adaptive imfs of emd. In: IEEE. **2016 2nd International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE)**. [S.l.], 2016. p. 1–4.

VARON, C. et al. A novel algorithm for the automatic detection of sleep apnea from single-lead ecg. **IEEE Transactions on Biomedical Engineering**, IEEE, v. 62, n. 9, p. 2269–2278, 2015.

WANG, L.; LIN, Y.; WANG, J. A rr interval based automated apnea detection approach using residual network. **Computer methods and programs in biomedicine**, Elsevier, v. 176, p. 93–104, 2019.

Xie, B.; Minn, H. Real-time sleep apnea detection by classifier combination. **IEEE Transactions on Information Technology in Biomedicine**, v. 16, n. 3, p. 469–477, 2012.

YANG, P. et al. A review of ensemble methods in bioinformatics. **Current Bioinformatics**, Bentham Science Publishers, v. 5, n. 4, p. 296–308, 2010.

Zhang, J. et al. A real-time auto-adjustable smart pillow system for sleep apnea detection and treatment. In: **2013 ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)**. [S.l.: s.n.], 2013. p. 179–190.

ZIHLMANN, M.; PEREKRESTENKO, D.; TSCHANNEN, M. Convolutional recurrent neural networks for electrocardiogram classification. In: IEEE. **2017 Computing in Cardiology (CinC)**. [S.l.], 2017. p. 1–4.