UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS
FACULDADE DE MEDICINA DE RIBEIRÃO PRETO
INSTITUTO DE QUÍMICA DE SÃO CARLOS

MATHEUS CALIL FALEIROS

Avaliação Computadorizada de Sacroiliíte em Imagens de Ressonância
Magnética

São Carlos

2018

MATHEUS CALIL FALEIROS

Avaliação computadorizada de sacroiliíte em imagens de ressonância
magnética

Versão Corrigida

Dissertação apresentada ao Programa de Pós-
Graduação Interunidades em Bioengenharia da
Escola de Engenharia
de São Carlos – Faculdade de Medicina de
Ribeirão Preto e Instituto de Química de São
carlos da Universidade de
São Paulo, como requisito para a obtenção do
Título de Mestre em Ciências

Área de concentração: Bioengenharia

Orientador: Prof. Dr. Paulo Mazzoncini de
Azevedo Marques.

São Carlos

2018

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues
Fontes da EESC/USP

F187a Faleiros, Matheus Calil
Avaliação computadorizada de sacroiliíte em imagens de ressonância magnética / Matheus Calil Faleiros; orientador Paulo Mazzoncini de Azevedo Marques. -- São Carlos, 2018.

Dissertação (Mestrado)- Programa de Pós-Graduação Interunidades em Bioengenharia e Área de Concentração em Bioengenharia -- Escola de Engenharia de São Carlos; Faculdade de Medicina de Ribeirão Preto; Instituto de Química de São Carlos, da Universidade de São Paulo, 2018.

1. Sacroiliíte. 2. Inflamação. 3. Classificação.
4. Reconhecimento de padrões. I. Título.

Elaborado por Elena Luzia Palloni – CRB 4464

FOLHA DE JULGAMENTO

Candidato(a): Matheus Calil Faleiros

Título: “ Avaliação computadorizada de sacroiliíte em imagens de ressonância magnética”

Data da defesa: 08/10/2018

Comissão Julgadora	Assinatura	Resultado
Prof(a). Dr(a). Paulo Mazzoncini de Azevedo Marques FMRP/USP		NÃO VOTANTE
Prof(a). Dr(a). Marcello Henrique Nogueira Barbosa FMRP/USP		APROVADO
Prof(a). Dr(a). Marcelo Costa Oliveira UFAL		APROVADO.
Prof(a). Dr(a). Renato Tinós FFCLRP/USP		APROVADO

AGRADECIMENTOS

Ao Prof. Dr. Paulo Mazzoncini de Azevedo Marques e ao Prof. Dr. Marcello Henrique Nogueira Barbosa pela orientação ao longo deste projeto.

Ao Msc. José Raniery Ferreira Júnior e ao Dr. Vitor Faeda Dalto pelo apoio durante o projeto.

A toda equipe de trabalho do laboratório, Vitor Francisco, Eddy Jens, Ariane Priscilla, Natália Santana, Sathya Quatrina e Beatriz Norbiato que se envolveram e me auxiliaram no projeto em algum momento.

A Natália Baptista Cruz por ter me auxiliado psicologicamente durante o processo de pós-graduação e por ajudar a revisar meus textos.

Aos professores Dr. Renato Tinós e Dr. Luiz Otávio Murta Junior.

A Paula Santos pelas consultorias gratuitas de *deep learning*.

A todos os meus amigos e familiares que me apoiaram.

RESUMO

FALEIROS, M. C. **Avaliação computadorizada de sacroilíte em imagens de ressonância magnética**. 2018. 198 f. Dissertação (Mestrado) - Programa de Pós-Graduação Interunidades em Bioengenharia da Escola de Engenharia de São Carlos - Faculdade de Medicina de Ribeirão Preto e Instituto de Química de São Carlos da Universidade de São Paulo, São Carlos, 2018.

Espondiloartrites são um grupo de doenças com características clínicas, radiológicas e laboratoriais comuns, que afetam principalmente as articulações sacroilíacas e possuem um forte potencial de morbidade e impacto socioeconômico. O protocolo padrão para diagnóstico da sacroilíte utiliza seis cortes coronais de imagens de ressonância magnética. Contudo, a avaliação visual das articulações sacroilíacas pode apresentar desafios aos especialistas, além de uma variação intrapessoal e interpessoal. Nesse sentido, o objetivo desse projeto foi desenvolver e comparar métodos de extração de atributos e classificação computadorizada a fim de detectar padrões nas imagens que levem a sua classificação como positiva ou negativa para presença de sacroilíte ativa. O projeto obteve aprovação do Comitê de Ética do Hospital das Clínicas de Ribeirão Preto. Imagens de exames de ressonância magnética de 51 pacientes, obtidas com sequência *Spectral Attenuated Inversion Recovery* (SPAIR) e 46 com sequência *Short Tau Inversion Recovery* (STIR) foram utilizadas retrospectivamente, sendo que cada exame contém 6 imagens obtidas em corte coronal. Cada exame foi classificado por um especialista em positivo ou negativo para sacroilíte ativa, totalizando 22 exames positivos e 29 negativos para SPAIR e 20 exames positivos e 26 negativos para STIR. Para evitar desbalanceamento de classes, o método de síntese de dados *Syntetic Minority Over-sampling Technique* foi utilizado. Cada imagem foi segmentada manualmente e colada em um fundo escuro para separar a região de interesse. Para reduzir possíveis ruídos, um pré-processamento utilizando uma transformação de perspectiva *warp* foi aplicado em cada uma das imagens, ampliando o tamanho da região de interesse para que cobrisse o fundo escuro. Os atributos extraídos para caracterizar as imagens foram estatísticos de níveis de cinza, textura de Haralick baseados na matriz de coocorrência de níveis de cinza, textura de Tamura baseados na direcionalidade do histograma, atributos espectrais baseados na transformada de Fourier, energias da wavelet de Haar, Gabor e estimativa da dimensão fractal. Cada paciente foi caracterizado pela média e desvio padrão de cada um dos atributos para suas 6 imagens do exame, formando um vetor de 230 componentes. Para reduzir o impacto da dimensionalidade, métodos de seleção de

atributos ReliefF, *Wrapper* e teste Mann-Whitney U foram utilizados. Para classificação, foram utilizados os métodos de aprendizado de máquina *naive bayes*, árvore de decisão J48, floresta aleatória, perceptron multicamadas, k-vizinhos mais próximos e máquina de vetores de suporte. A classificação foi realizada utilizando validação cruzada 10-*fold* e foi avaliada pelas métricas sensibilidade, especificidade, área sob a curva ROC (*receiver operating characteristic*) (AUC), e pelas taxas de acerto e erro. Métodos de aprendizado profundo também foram utilizados. Foi efetuada uma comparação entre a concordância das sequências SPAIR e STIR utilizando coeficiente Kappa. A AUC máxima alcançada foi de 0,975 utilizando o perceptron multicamadas com seleção de atributos reliefF.

Palavras chave: Sacroilíte. Inflamação. Classificação. Reconhecimento de padrões.

ABSTRACT

FALEIROS, M. C. **Computer-aided diagnosis of sacroiliitis in resonance magnetic images**. 2018. 198 f. Dissertação (Mestrado) - Programa de Pós-Graduação Interunidades em Bioengenharia da Escola de Engenharia de São Carlos - Faculdade de Medicina de Ribeirão Preto e Instituto de Química de São Carlos da Universidade de São Paulo, São Carlos, 2018.

Spondyloarthritis are a group of diseases with common clinical, radiological and laboratory characteristics that mainly affect the sacroiliac joints and have a strong potential for morbidity and socioeconomic impact. The standard protocol for the diagnosis of sacroiliitis uses six coronal sections of magnetic resonance imaging. However, the visual evaluation of the sacroiliac joints may present challenges to the specialists, as well as an intrapersonal and interpersonal variation. In this sense, the objective of this project is to develop and compare methods of feature extraction and computerized classification in order to detect patterns in the images that lead to the classification between positive and negative for sacroiliitis. The project was approved by the Ethics Committee of the Hospital das Clínicas of Ribeirão Preto. Images of the exam of 51 patients for *Spectral Attenuated Inversion Recovery* and 46 for *Short Tau Inversion Recovery* were used retrospectively, and each exam contains 6 images. Each exam was classified by a specialist in positive and negative for sacroiliitis, totaling 22 positive and 29 negative exams for SPAIR and 20 positives and 26 negative for STIR. To avoid class imbalance, we used the data synthesis method *Syntetic Minority Over-sampling Technique*. Each image was manually segmented and pasted to a black background to separate the region of interest. To reduce possible high-frequency noise between the region of interest and the black background, a pre-processing using a warp perspective transformation was applied to each of the images, increasing the size of the region of interest to cover the entire black background. The extracted features to characterize were gray level statistics, Haralick texture based on gray level co-occurrence matrix, Tamura texture based on histogram directionality, spectral attributes based on Fourier transform, Haar wavelet energies, Gabor and fractal dimension estimation. Each patient was characterized by the mean and standard deviation of each of the features for their 6 images of the exam, forming a vector of 230 components. The feature selection methods ReliefF, Wrapper and Mann-Whitney U test was used. For classification, we used the conventional methods of machine learning naive bayes, decision tree J48, random forest, multilayer perceptron, k-nearest neighbors and support

vector machine. The classification was performed using 10-fold cross validation and was evaluated by the sensitivity, specificity, area under the receiver operating characteristic (AUC), and complemented by the values of the hit rate and error rate. Deep learning methods was used. The comparison between SPAIR and STIR sequences was performed using kappa statistics. The highest AUC obtained was 0,975 using multilayer perceptron and ReliefF feature selection method.

Keywords: Sacroiliitis. Inflammation. Classification. Pattern recognition.

LISTA DE FIGURAS

Figura 1 - Exemplos de imagens de ressonância magnética apresentando articulações sacroilíacas (elipse azul) normal (Figura (a)) e com inflamação (Figura (b)). Pode-se observar o aumento do sinal (maior brilho) ao redor das superfícies articulares das articulações sacroilíacas apresentadas na Figura (b) em relação à Figura (a)	16
Figura 2 - Processo da aplicação da transformação Warp nas imagens	21
Figura 3 - Ilustração do funcionamento de um KNN com $K = 3$, 2 atributos e 2 classes (quadrados e triângulos)	24
Figura 4 - Perceptron multicamadas	26
Figura 5 - Estrutura de uma árvore de decisão. Cada nó interno representa uma regra, enquanto cada nó terminal representa um desfecho	26
Figura 6 - Floresta aleatória com N árvores. Cada árvore tem sua saída individual, a floresta tem como saída a votação do resultado das saídas individuais	27
Figura 7 - SVM com duas dimensões	28
Figura 8 - Aplicação do algoritmo Wrapper para seleção de atributos.	30
Figura 9 - Abstração em alto nível da arquitetura de uma CNN	32
Figura 10 - Intervalo de confiança de 95% para sensibilidade e especificidade dos classificadores SVM e MLP	40
Figura 11 - Valores de AUC utilizando o método de seleção de atributos ReliefF	41
Figura 12 - Valores de Sensibilidade utilizando o método de seleção de atributos ReliefF	42
Figura 13 - Valores de Especificidade utilizando o método de seleção de atributos ReliefF	43
Figura 14 - Intersecção dos subconjuntos de atributos agrupados por classificador utilizando o método Wrapper.	44
Figura 15 - Intersecção entre os 3 métodos de seleção de atributos	46
Figura 16 - Valores de AUC utilizando o método de seleção de atributos ReliefF	47
Figura 17 - Valores de Sensibilidade utilizando o método de seleção de atributos ReliefF	48
Figura 18 - Valores de Especificidade utilizando o método de seleção de atributos ReliefF	49
Figura 19 - Modelo de rede neural profunda implementado para o estudo	55

LISTA DE FIGURAS

Figura 20 - Código em Python do modelo de rede neural profunda

56

LISTA DE TABELAS

Tabela 1 - Exemplo genérico de uma tabela de concordância	36
Tabela 2 - Avaliação do coeficiente de concordância kappa	37
Tabela 3 - Métricas de avaliação para cada classificador com classes desbalanceadas. Os resultados em azul mostram os melhores valores para cada métrica, enquanto os vermelhos mostram os piores valores	38
Tabela 4 - Métricas de avaliação para cada classificador com classes balanceadas por SMOTE. Os resultados em azul mostram os melhores valores para cada métrica, enquanto os vermelhos mostram os piores valores	39
Tabela 5 - Métricas de cada classificador sem SMOTE utilizando somente o subconjunto selecionado por ele e utilizando o subconjunto de intersecção	45
Tabela 6 - Métricas de cada classificador com SMOTE utilizando somente o subconjunto selecionado por ele e utilizando o subconjunto de intersecção	45
Tabela 7 - Maiores valores de cada métrica para cada classificador	50
Tabela 8 - Métricas para o subconjunto de atributos selecionado pelo Wrapper	51
Tabela 9 - Tabela de concordância para todos os atributos	52
Tabela 10 - Atributos selecionados pelo Wrapper para as sequências SPAIR e STIR	53
Tabela 11 - Tabela de concordância para os atributos selecionados pelo método Wrapper	53
Tabela 12 - Valores de acurácia por teste e acurácia média calculada	56

LISTA DE EQUAÇÕES

Equação 1 - <i>Warp</i>	20
Equação 2 - Teorema de Bayes	25
Equação 3 - SMOTE	29
Equação 4 - Área sob a curva ROC estimada pelo teste estatístico Mann-Whitney U	31
Equação 5 - Cálculo do coeficiente Kappa	36

LISTA DE ABREVIATURAS

AUC – Área sob a curva.

J48 – Árvore de decisão.

KNN – K vizinhos próximos.

MLP – Perceptron Multicamadas.

NB – Naive Bayes.

RF – Floresta aleatória.

SMOTE – *Syntetic Minority Over-sampling Technique*.

SVM – Máquina de vetores de suporte.

SUMÁRIO

1 INTRODUÇÃO	21
1.1 SACROILÍTE: DIAGNÓSTICO, AVALIAÇÃO E DESAFIOS CLÍNICOS	21
1.2 APRENDIZADO DE MÁQUINA EM IMAGENS MÉDICAS	22
1.3 OBJETIVOS DO TRABALHO	23
2 MATERIAIS E MÉTODOS	25
2.1 AQUISIÇÃO, SEGMENTAÇÃO E PROCESSAMENTO DAS IMAGENS	25
2.2 EXTRAÇÃO DE ATRIBUTOS	26
2.3 APRENDIZADO DE MÁQUINA	27
2.4 SÍNTESE DE DADOS	33
2.5 SELEÇÃO DE ATRIBUTOS	33
2.6 APRENDIZADO PROFUNDO	35
2.7 COEFICIENTE DE CONCORDÂNCIA KAPPA	38
3 RESULTADOS E DISCUSSÃO	41
3.1 SPAIR	41
3.2 SELEÇÃO DE ATRIBUTOS	43
3.3 SELEÇÃO DE ATRIBUTOS E APRENDIZADO DE MÁQUINAS PARA SEQUÊNCIA STIR	49
3.4 COMPARAÇÃO ENTRE AS SEQUÊNCIAS SPAIR E STIR	54
3.5 APRENDIZADO PROFUNDO	57
4 CONCLUSÃO	61
5 REFERÊNCIAS	63
APÊNDICE A - PRODUÇÃO CIENTÍFICA ASSOCIADA À PESQUISA	67

1 INTRODUÇÃO

1.1 SACROILIÍTE: DIAGNÓSTICO, AVALIAÇÃO E DESAFIOS CLÍNICOS

O conceito de espondiloartrites (EpA) foi proposto pela primeira vez em 1974 pelos pesquisadores ingleses Moll e Wright que propuseram que se englobasse dentro do mesmo conjunto algumas doenças que até então eram consideradas completamente distintas. Essas doenças possuem aspectos clínicos, laboratoriais e radiológicos comuns (COSTA et al., 2015). As EpA acometem predominantemente jovens adultos e podem ter seu início aos 16 anos. Elas possuem um forte potencial de morbidade e impacto socioeconômico, causando comprometimento da coluna vertebral, esqueleto apendicular e das articulações sacroilíacas. O impacto socioeconômico se refere desde a impossibilidade de se manter atividades profissionais regulares até ao custo alto dos medicamentos utilizados no tratamento. Boonen (2002), em seu artigo de revisão, cita trabalhos onde foi identificado que a população alemã que sofria de EpA tinha uma taxa de desemprego 11% maior que o padrão de outras populações, além de outros que indicam que o custo do tratamento pode chegar a US\$ 6.270 (seis mil duzentos e setenta) por paciente por ano. O seu diagnóstico tardio pode ocasionar, no pior caso, a necessidade de prótese total do quadril (SAMPAIO-BARROS et al., 2007).

Os exames de imagem apresentam fundamental importância na avaliação desses pacientes, sendo a ressonância magnética (RM) considerada atualmente o método mais sensível para o diagnóstico precoce. A RM é capaz de detectar edema ósseo nas articulações sacroilíacas (Figura 1), sendo esse achado um dos critérios diagnósticos mais importantes dos protocolos clínicos atuais (RUDWALEIT et al., 2009).

Figura 1 - Exemplos de imagens de ressonância magnética apresentando articulações sacroilíacas (elipse azul) normal (Figura (a)) e com inflamação (Figura (b)). Pode-se observar o aumento do sinal (maior brilho) ao redor das superfícies articulares das articulações sacroilíacas apresentadas na Figura (b) em relação à Figura (a).

Fonte - Autor

Para o protocolo de RM estabelecido para diagnóstico da sacroiliíte proposto pela *Assessment of SpondyloArthritis International Society* (ASAS) (SIEPER et al., 2009) consta, entre outros fatores, a sequência ponderada em STIR (*Short Tau Inversion Recovery*) (MONTANDON et al., 2007). Porém, um estudo recente já confirmou que não existe diferença significativa entre o uso dessa sequência ou da sequência em SPAIR (*Spectral Attenuated Inversion Recovery*) para diagnóstico (DALTO et al., 2017). A avaliação proposta pela ASAS permite a classificação binária, por inspeção visual, entre positivo e negativo para sacroiliíte.

A avaliação radiológica somente qualitativa por inspeção visual das articulações sacroilíacas pode apresentar desafios aos especialistas, além de possíveis variações intra- e inter-pessoal na interpretação das imagens.

1.2 APRENDIZADO DE MÁQUINA EM IMAGENS MÉDICAS

Aprendizado de máquina é uma área da inteligência artificial cujo objetivo é o desenvolvimento de técnicas computacionais sobre o aprendizado, bem como a construção de sistemas capazes de adquirir conhecimento de forma automática (REZENDE, 2003). Quando aplicado em imagens médicas, aprendizado de máquinas segue um conjunto de etapas bem definidas: uma etapa de aquisição e processamento dos dados, uma etapa para extração de atributos relevantes para quantificar as imagens e a etapa de escolha e treinamento do modelo de

aprendizado de máquina no conjunto de atributos (ERICKSON, 2017).

Estudos recentes aplicaram essa metodologia em tipos variados de doenças. Kunimatsu (2018) obteve imagens de RM de 55 pacientes com glioblastomas e 21 pacientes com linfoma primário do sistema nervoso central, com o objetivo de treinar um classificador do tipo máquina de vetores de suporte que pudesse diferenciar essas doenças.

Ferreira Junior et al. (2018) treinou um classificador baseado em rede neural artificial de base radial utilizando 2465 atributos de imagens de tomografia computadorizada de 85 pacientes com câncer de pulmão para identificar o subtipo do tumor e o se houve ou não metástase.

Nasr-Esfahani et al. (2016) aplicou uma rede neural convolucional profunda em imagens capturadas por câmeras da superfície de melanomas e não melanomas a fim de reconhecer quando a doença está presente na imagem. Em contraste com os outros dois trabalhos apresentados, a utilização de uma rede neural convolucional profunda elimina a necessidade de uma prévia extração de atributos, deixando essa etapa implícita na etapa de treinamento do modelo.

Um estudo de revisão publicado em 2017 (LITJENS et al., 2017) apresenta a contribuição de mais de 300 trabalhos envolvendo aplicação de métodos de aprendizado profundo em imagens médicas, majoritariamente imagens de ressonância magnética (aproximadamente 80 trabalhos) e tendo como um dos principais objetivos a classificação do exame de um paciente.

Contudo, apesar do histórico recente de aprendizado de máquinas em imagens médicas apresentar diversos contextos e aplicações, até o presente momento nenhum trabalho aplicando aprendizado de máquinas para classificação entre positivo e negativo para sacroilíte foi encontrado.

1.3 OBJETIVOS DO TRABALHO

O objetivo geral deste estudo foi estabelecer um procedimento de classificação computadorizada das articulações sacroilíacas, baseado em atributos quantitativos extraídos de imagens de RM, para auxiliar o diagnóstico das EpA. Espera-se, dessa forma, contribuir no futuro para um diagnóstico mais preciso, melhorando a atenção à saúde de pacientes com espondiloartrites.

Para atingir o objetivo geral, alguns objetivos secundários específicos foram definidos:

- Desenvolver/organizar uma base de imagens de RM anonimizadas e manualmente segmentadas com exemplos de casos com e sem inflamação ativa das articulações sacroilíacas;
- Desenvolver algoritmos de extração de atributos das imagens;
- Desenvolver algoritmos de reconhecimento de padrão para classificação de casos de inflamação sacroilíaca;
- Avaliar a eficiência dos algoritmos de extração de atributos e classificação de imagens de casos de inflamação sacroilíaca;
- Comparar a classificação de RM de sequência STIR com sequência SPAIR
- Comparar métodos de classificação convencionais de aprendizado de máquina com métodos de aprendizado profundo (deep learning).

2 MATERIAIS E MÉTODOS

2.1 AQUISIÇÃO, SEGMENTAÇÃO E PROCESSAMENTO DAS IMAGENS

A pesquisa se baseia em um estudo retrospectivo com imagens de pacientes com EpA, aprovado pelo CEP-HCFMRP (processo: 1.600.012 e CAAE: 56887916.0.0000.5440). O conjunto de imagens é composto por exames de 51 pacientes com sequência SPAIR, sendo 22 pacientes positivos para sacroilíte ativa (articulação inflamada) e 29 negativos, e exames de 46 pacientes com sequência STIR, sendo 20 pacientes positivos para sacroilíte e 26 negativos, conforme avaliação feita por um radiologista utilizando os critérios de diagnóstico por RM da ASAS. As imagens foram obtidas em aparelho 1.5T (Achieva, Philips Medical Systems) usando uma bobina de coluna, com os pacientes em decúbito dorsal, em sequência coronal. De cada exame, o radiologista especialista selecionou 6 cortes (imagens) consecutivos e todas as imagens foram anonimizadas, garantindo a privacidade dos pacientes. As articulações esquerda e direita foram identificadas e manualmente segmentadas em cada corte por um radiologista e inseridas em um fundo preto.

Cada uma das imagens inseridas no fundo preto foi processada por uma transformação de perspectiva *Warp*, para aumentar a região de interesse de forma a cobrir toda a imagem (Figura 2). Essa transformação envolve polinômios não-lineares e pode ser definida de forma geral como apresentado na equação 1:

$$destino(x, y) = origem\left(f_x(x, y), f_y(x, y)\right) \quad (1)$$

onde $destino(x, y)$ é a imagem destino no ponto (x, y) , $origem$ é a imagem origem sendo mapeada, não alterando o conteúdo da imagem, apenas deformando a grade de pixels, fazendo seu mapeamento para a imagem resultante, f_x e f_y são funções de interpolação de pixel (GONZALES; WOODS, 1993). Esse processamento foi realizado para minimizar a presença de ruídos de alta frequência e otimizar o processo de classificação, conforme observado em um estudo anterior (FALEIROS et al., 2017).

Figura 2 - Processo da aplicação da transformação *Warp* nas imagens

Fonte – Autor

2.2 EXTRAÇÃO DE ATRIBUTOS

Sabendo que a inflamação é representada nas imagens de RM como variação em níveis de cinza em torno da articulação, foi realizada a extração de atributos baseados em níveis de cinza, textura e do espectro de frequências de cada imagem processada pelo método *Warp*.

Do nível de cinza foram obtidos 7 atributos estatísticos, sendo estes: média, desvio padrão, assimetria, kurtosis, variância, coeficiente de desvio e valor máximo de nível de cinza.

Referente a textura da imagem, foram extraídos 14 atributos derivados da matriz de coocorrência de níveis de cinza e propostos por Haralick, Shanmugan e Dinstein (1973) que são segundo momento angular, contraste, correlação, variância, momento da diferença, média da soma, entropia da soma, variância da soma, entropia, variância da diferença, entropia da diferença, duas medidas de correlação e coeficiente de correlação máxima. Ainda referentes à textura, 18 atributos baseados na direcionalidade do histograma propostos por Tamura, Mori e Yamawaki (1978) foram extraídos, sendo estes contraste, granularidade e direcionalidade obtidas em 16 direções. Os algoritmos de extração de atributos de níveis de cinza e textura foram implementados usando a biblioteca de código aberto JFeatureLib (JFEATURELIB OPEN SOURCE PROJECT, 2016).

Do módulo do espectro de frequências, da Transformada Rápida de Fourier (*Fast Fourier Transform - FFT*) de cada imagem, foram extraídos os mesmos 7

atributos já utilizados para a caracterização do histograma de níveis de cinza. Uma restrição da transformada rápida é que a imagem de entrada deve possuir dimensões em potência de 2.

Ainda em atributos de frequência, também foram extraídos a partir das 4 energias em segundo nível e 4 energias em terceiro nível da wavelet de Haar. A wavelet de Haar é uma wavelet quadrada binária, aplicada através de convoluções realizadas entre as imagens e os filtros da transformada (LIMA, 2003).

Wavelets de Gabor foram também utilizadas para a caracterização das imagens. Essas transformadas são um grupo de wavelets que calcula a energia em determinada banda de frequência e direção, sendo eficientes para captura de características locais da frequência estudada (ZHANG et al., 2000).

Como último atributo implementado, uma estimativa da Dimensão Fractal foi obtida através de uma média do método de contagem de caixas para os limiares 50, 100, 150 e 200. Um Fractal é, por definição, uma repetição de padrões em diferentes escalas, o que condiz com a definição de textura, que é uma repetição de determinado padrão na imagem (FRACTAL FOUNDATION, 2017).

Todos os atributos propostos foram extraídos para as 6 imagens do exame do paciente e, para a caracterização do vetor final, a média e o desvio padrão dos atributos foram calculados, considerando-se que a média poderá representar o padrão inflamatório quando este estiver presente em várias imagens do exame, enquanto o desvio padrão será representativo quando o padrão inflamatório estiver presente em poucas imagens do exame. O vetor de atributos total contém 230 componentes. Todos os valores dos vetores de atributos foram normalizados para o intervalo entre zero e um para evitar que a escala de um atributo influenciasse em sua importância em relação aos outros atributos e para ajustar os valores para um intervalo computável por todos os métodos de aprendizado de máquinas utilizados.

2.3 APRENDIZADO DE MÁQUINA

Aprendizado de máquina é uma área da inteligência artificial cujo objetivo é o desenvolvimento de técnicas computacionais sobre o aprendizado, bem como a construção de sistemas capazes de adquirir conhecimento de forma automática (REZENDE, 2003).

Dentre os vários métodos de aprendizado de máquina, aqueles utilizados nessa pesquisa são apresentados a seguir:

K-Vizinhos mais Próximos (*K-nearest neighbor* - KNN): também conhecido como algoritmo baseado em instância, o KNN escolhe a classe predominante entre os k -vizinhos mais próximos para classificar uma instância de testes com classe desconhecida. A variante do KNN utilizada neste trabalho é a proposta por David W. Aha, o *Instance-Based Learning* (IBL) (AHA; KIBLER; ALBERT, 1991). A figura 3 ilustra um KNN com o valor de $K = 3$, 2 atributos e 2 classes representadas por quadrados e triângulos, onde deseja-se classificar uma nova instância, X. Para tal, uma distância euclidiana é calculada entre a instância X e todas as outras instâncias, a classe majoritária entre as 3 instâncias mais próximas definem a classe de X.

Figura 3 - Ilustração do funcionamento de um KNN com $K = 3$, 2 atributos e 2 classes (quadrados e triângulos).

Fonte: Autor.

Naive-Bayes (NB): classificador probabilístico baseado no teorema de Bayes dado pela equação 2:

$$P(Ck|x) = \frac{P(Ck)P(x|Ck)}{P(x)} \quad (2)$$

onde Ck é uma classe e x um vetor de atributos. O lado esquerdo da equação contém as probabilidades à *posteriori*, que serão ajustadas de acordo com as probabilidades à *priori*, do lado direito da equação.

Esse classificador assume que os atributos numéricos são descritos por distribuições Gaussianas, já que esta provê uma boa aproximação para muitas

distribuições no mundo real. De forma geral, esse classificador assume que os atributos preditivos são independentes e cada valor numérico é modelado de acordo com uma distribuição de probabilidade contínua dentro dos valores do referido atributo.

Perceptron Multicamadas (*Multilayer Perceptron* – MLP): rede neural artificial multicamadas, das quais a primeira é a camada de entrada, ou seja, a camada de inserção dos valores dos atributos extraídos das imagens, as camadas intermediárias serão classificadores lineares e a última camada é a saída (classes). O treinamento de um Perceptron Multicamadas consiste na passagem dos dados ao longo da rede, em direção a camada de saída, executando operações lineares dadas por $Z = W * X + b$ em cada nó, onde W é o vetor de pesos, X é um vetor de entrada da camada e b uma constante, seguido da aplicação de uma função de ativação que pode ser descrita de forma geral como $ativação(Z)$. Após a passagem dos dados ao longo de toda a rede, uma etapa de propagação para trás calcula o erro e atualiza o vetor de pesos W de acordo com o erro multiplicado por uma constante, chama de taxa de aprendizado. A figura 4 ilustra um Perceptron Multicamadas.

Figura 4 - Perceptron multicamadas.

Fonte: Adaptado de Araújo et al. (2013).

Árvore de decisão (J48): implementação da árvore C4.5, proposta por Quinlan

(1993) em seu livro em Java. A J48 é uma árvore de decisão que seleciona os atributos mais descritivos para a classificação. No contexto de imagens médicas, além de indicar a classificação da amostra, ela pode ajudar também a entender quais critérios foram usados para chegar nessa conclusão. A figura 5 mostra a estrutura de uma árvore de decisão.

Figura 5 - Estrutura de uma árvore de decisão. Cada nó interno representa uma regra, enquanto cada nó terminal representa um desfecho.

Fonte: Autor.

Floresta Aleatória (*Random Forest* - RF): Constrói uma floresta de árvores de decisões geradas aleatoriamente. A utilização de atributos aleatórios para criar um número significativo de árvores diminui o erro, aumenta a robustez quando se trabalha com outliers e ruídos, possui alto desempenho computacional, gera uma estimativa mais precisa de fatores como erro, correlação e importância das variáveis e, por fim, é fácil de ser paralelizado (BREIMAN, 2001). Dado uma entrada, o modelo retorna uma saída de acordo com a votação de cada uma das árvores que compõem a floresta. A figura 6 ilustra uma floresta aleatória.

Figura 6 - Floresta aleatória com N árvores. Cada árvore tem sua saída individual, a floresta tem como saída a votação do resultado das saídas individuais.

Fonte: Adaptado de Nguyen, Wang e Nguyen (2013).

Máquina de Vetores de Suporte (*Support Vector Machine* - SVM): É um classificador que utiliza um hiperplano, que pode ser de base radial ou linear, para separar o conjunto de dados e, em seguida, utiliza vetores de suporte, ou seja, vetores do conjunto de dados mais próximos deste hiperplano, para otimizar a distância, aumentando a capacidade de generalização do método (HEARST, 1998). No projeto em questão a base da SVM escolhida foi a linear, após a realização de testes iniciais também com kernels polinomiais e radiais. A figura 7 ilustra o classificador SVM.

Figura 7 - SVM com duas dimensões.

Fonte: Adaptado de Ivan (2014).

Os métodos de aprendizado de máquina utilizados neste projeto estão implementados na biblioteca de código aberto Weka (WITTEN et al., 2016, p.321-357).

Para a análise proposta na seção 3.1 e 3.2, foi realizado um ajuste fino dos parâmetros dos classificadores definindo-os como segue:

- KNN foram utilizados K de 1 a 5.
- MLP a taxa de aprendizado foi de 0.3 e o momento de 0.2, utilizando 500 épocas e uma camada oculta. Um número maior de épocas não alterou a performance do classificador, porém elevou o tempo de treino. Valores diferentes de taxa de aprendizado e momento geram erros maiores.
- SVM utilizou uma função de base linear após testes com bases radiais e polinomiais.
- RF gerou 100 árvores de decisão
- J48 com poda e utilizou o fator de confiança de 0.25.
- NB não possui parâmetros a serem otimizados.

Para a análise proposta na seção 3.3, os classificadores utilizados foram:

- SVM utilizou uma função de base linear após testes com bases radiais e polinomiais.
- KNN foram utilizados K = 1, 3 e 5.
- MLP a taxa de aprendizado foi de 0.3 e o momento de 0.2, utilizando 500 épocas e uma camada oculta.

2.4 SÍNTESE DE DADOS

A performance de métodos de aprendizado de máquina é tipicamente avaliada através de acurácia preditiva. Porém esse método não é apropriado quando o conjunto de dados não está balanceado, pois nesse caso uma simples estratégia de classificar todos os exemplos como a classe majoritária poderia resultar em uma acurácia alta.

Com o intuito de contornar este problema, sabendo que a base de imagens do problema em questão possui, majoritariamente, exemplos da classe negativa para sacroilíte ativa, foi proposto o uso de um método de síntese de dados baseado em vizinhança, *Syntetic Minority Over-sampling Technique* (SMOTE), com o intuito de avaliar o efeito do desbalanceamento dos dados na classificação e simular testes contendo uma base balanceada.

O método SMOTE gera dados sintéticos trabalhando no espaço dos atributos, e não no espaço dos dados brutos (níveis de cinza das imagens), utilizando uma técnica de k vizinhos mais próximos e um número aleatório para efetuar o cálculo da nova amostra, garantindo que ela sempre esteja entre amostras da mesma classe. (CHAWLA, 2002)

De forma geral, um exemplo da aplicação do método utilizando 2 vizinhos próximos é mostrado na equação 3:

$$(X, Y) = (x_1, y_1) + \text{rand}(0 - 1) * (x_2, y_2) \quad (3)$$

onde (X, Y) é o novo vetor gerado pelo SMOTE, (x_1, y_1) e (x_2, y_2) são os vetores vizinhos e $\text{rand}(0-1)$ é um número aleatório entre 0 e 1.

2.5 SELEÇÃO DE ATRIBUTOS

Como descrito na seção 2.2 - Extração de atributos, a extração de atributos resultou em um vetor com 230 variáveis. Um vetor dessa dimensionalidade tende a gerar um problema para medição da qualidade dos atributos, um problema chave para métodos de aprendizado de máquinas (KONONENKO, 1994).

Para solucionar esse problema, três métodos de seleção de atributos foram utilizados e comparados. Estes serão descritos a seguir.

ReliefF: Variante do algoritmo Relief para problemas multi classe, capaz de estimar a habilidade do atributo de separar cada par de classes independentemente de quão perto essas classes estão umas das outras. Para isso, o método atribui probabilidades *a-priori* para cada uma das classes e calcula como cada atributo altera essa probabilidade de predição (KONONENKO, 1994). O método ReliefF está implementado na biblioteca de código aberto Weka.

Wrapper: Esse método utiliza um classificador para efetuar a seleção dos atributos partindo do princípio de que o classificador deve “saber” quais os melhores atributos a serem usados, ou seja, saber qual atributo diminui ou aumenta sua performance. A Figura 8 ilustra o algoritmo *Wrapper* (KOHAVI; JOHN, 1997).

Figura 8 - Aplicação do algoritmo *Wrapper* para seleção de atributos.

Fonte: Adaptado de Kohavi e John (1997).

Teste de Mann-Whitney U: Utiliza um teste estatístico não paramétrico para avaliar a diferença de duas amostras. Esse teste permite realizar uma análise univariante, atribuindo significância estatística a cada variável através do valor p . Ele também possui uma relação com a área sob a curva ROC (*receiver operating characteristic* - AUC) que pode ser descrita conforme a Equação 4 (MASON; GRAHAM, 2002):

$$AUC = \frac{U}{n_1 \cdot n_2} \quad (4)$$

onde U é o valor do teste Mann-Whitney e n_1 e n_2 o número de amostras para cada um dos grupos.

2.6 APRENDIZADO PROFUNDO

Aprendizado Profundo (ou *Deep Learning*) é uma técnica de Aprendizado de Máquina que consiste na elaboração de redes neurais artificiais complexas a fim de extrair características relevantes para o aprendizado a partir de dados brutos. A maior vantagem da utilização dessa técnica é a aplicação direta do modelo nos dados, sem uma prévia preocupação com extração de atributos ou seleção de atributos, porém exige um custo computacional grande e cuidados no ajuste fino do modelo.

A aplicação dessa técnica em imagens consiste na elaboração de uma rede neural convolucional (*Convolutional Neural Network - CNN*). Essa arquitetura normalmente é composta por uma camada de entrada, uma série de camadas ocultas chamadas de extratores de características e por fim uma camada totalmente conectada para gerar a saída desejada. A Figura 9 ilustra uma arquitetura genérica de uma CNN (PATTERSON; GIBSON, 2017).

Figura 9 - Abstração em alto nível da arquitetura de uma CNN.

Fonte: Autor.

Quando aplicado em imagens, a camada de entrada de uma CNN consiste nos próprios pixels das imagens, que são vistos como tensores N -dimensionais. Um tensor é um vetor multidimensional. Sendo assim, a camada de entrada recebe os dados de acordo com as dimensões da imagem (altura e largura), número de canais de cor (1 para níveis de cinza e 3 para RGB) e número de imagens que compõem o *mini-batch*. O conceito de *mini-batch* será explicado mais à frente nessa mesma seção.

Uma camada convolucional aplica filtros de convolução através de janelas nos dados de entrada a fim de buscar abstrações em mais baixo nível na imagem. Cada filtro gera uma característica diferente. A saída de uma camada convolucional é um mapa de características composto de cada um desses filtros.

Após a passagem dos dados por uma ou mais camadas convolucionais, é recomendado a aplicação de uma camada de *Pooling*. Essa camada aplica janelas de operações de máximo (*MaxPooling*) ou média (*AvgPooling*) na saída da camada de convolução com o intuito de reduzir a dimensionalidade do problema. Por fim, após sucessivas camadas convolucionais e *Pooling*, uma ou mais camadas totalmente conectadas são usadas para computar a saída da rede.

Apesar de possuir uma arquitetura em alto nível simples e bem definida, uma CNN ainda necessita de cuidados para treinamento e ajuste fino de parâmetros em cada uma das camadas ou ajustes globais válidos para todas as camadas. Esses parâmetros serão descritos a seguir (PATTERSON; GIBSON, 2017):

Tamanho do *Batch* (ou *Mini-Batch*): É o número de exemplos a ser passado no modelo ao mesmo tempo. Tamanho ideal é que seja múltiplo de 32 ou alguma potência de 2 menor que 32, caso o conjunto de dados não tenha um tamanho adequado. Caso a base de dados seja muito pequena, um tamanho igual a 1 pode ser usado. Esse parâmetro tem influência direta na melhora da performance de tempo de treino da rede, porém aumenta o custo de memória. O *mini-batch* deve estar bem heterogêneo com informações do máximo de classes possíveis.

Épocas e Iterações: Época (*Epoch*) é o número de vezes que todos os dados farão uma passagem completa pela CNN, enquanto Iteração é o número de vezes que o mesmo dado irá passar pela rede consecutivamente. Por exemplo, sejam A e B dados de entrada em uma CNN, 3 épocas e 1 Iteração seria ABABAB enquanto 3 Iterações e 1 época seria AAABBB. A ordem de passagem dos dados possui influência no aprendizado, porém ao utilizar uma grande quantidade de dados essa diferença se torna não significativa.

Inicialização dos Pesos: O treinamento de qualquer rede neural nada mais é do que o ajuste de pesos em um grafo, porém o modo como esses pesos são inicializados pode influenciar no treinamento. Os pesos podem ser inicializados todos em zero, seguindo uma distribuição normal ou seguindo o método de Xavier (GLOROT; BENGIO, 2010), que leva em conta o tamanho da entrada, sendo menos arbitrário que os outros métodos.

Tamanho da camada convolucional: Uma camada convolucional possui 2 parâmetros específicos importantes, o tamanho da janela de convolução (*kernel*) e o tamanho do deslocamento (*stride*) dessa janela. É preferível o uso de janelas convolucionais pequenas (dimensões 3x3 pixels até 5x5 pixels) do que janelas muito grandes pois elas conseguem capturar abstrações em níveis mais baixos. O tamanho do deslocamento da janela nunca pode ser maior que o tamanho da janela, porém pode ser igual, e é recomendado que seja de apenas 1 pixel na vertical e na horizontal. O número de filtros, ou seja, o tamanho da saída, também é um parâmetro importante de uma camada convolucional e, apesar de não existir uma heurística bem definida para esse parâmetro, é de entendimento que ele deva aumentar à medida que a rede fique mais profunda.

Função de ativação: É a função que vai decidir se determinado neurônio será ou não ativado. Existem diversas funções de ativação, as mais comuns para problemas de classificação com CNN são a sigmoideal e *softmax* usadas na camada de saída e *rectified linear unit* (ReLU) utilizada normalmente nas camadas ocultas.

Taxa de aprendizado: É considerado um dos parâmetros mais importantes pois regula a atualização dos pesos da rede neural. Seu valor influencia diretamente a velocidade de convergência do modelo. É recomendado iniciar seu valor entre 0,1 e 0,0001 e adicionar uma taxa de queda com o passar do tempo. Deve-se tomar cuidado para não utilizar valores que irão causar divergência no modelo.

Função de custo: Quantifica o quão perto um dado modelo está próximo ao ideal de acordo com o treino. Calcula-se uma métrica baseada no erro da predição do modelo, agrega-se esse erro a toda base de dados e calcula-se uma média. Para problemas de classificação as mais comuns são a *Hinge loss* e a *Negative Log Likelihood*.

Regularização: Método utilizado para ajustar modelos de forma que minimize a superposição com seus dados (o que levaria a um aprendizado não generalizável). Alguns dos métodos comuns para regularização são: L2, também chamado de queda de peso, adiciona um termo extra a função de custo, uma soma de quadrados de todos os pesos da rede, fazendo com que pesos maiores sejam penalizados pela rede, e *dropout*, que ignora de forma aleatória a ativação de determinados neurônios, a fim de inserir um ruído proposital no aprendizado.

As configurações de *hardware* utilizadas para os testes de aprendizado profundo foram: placa de vídeo NVidia GTX 920M 4GB, 8GB de memória RAM e

processador intel core i7 2.8GHz.

2.7 COEFICIENTE DE CONCORDÂNCIA KAPPA

Técnica proposta por Cohen (1960) que tem como objetivo avaliar a concordância entre dois juízes sobre uma determinada situação. O cálculo desse coeficiente está ilustrado na Equação 5

$$\kappa = \frac{p_o - p_e}{1 - p_e} (5)$$

onde p_o é a concordância observada e p_e é a concordância esperada em caso aleatório. Esses valores podem ser calculados através de uma tabela chamada de tabela de concordância, como ilustrado na tabela 1.

Tabela 1 - Exemplo genérico de uma tabela de concordância.

Juiz 1				
		P	N	Total
Juiz 2	P	p11	p12	p13
	N	p21	p22	p23
	Total	p31	p32	p33

Fonte: Autor

Onde p_{11} é a porção de valores que os dois juízes classificaram como positivos, p_{12} é a porção de valores que o juiz 1 classificou como negativo e o juiz 2 classificou como positiva, p_{21} é a porção de valores que o juiz 1 classificou como positivo e o juiz 2 classificou como negativo, p_{22} é a porção de valores que o juiz 1 e o juiz 2 classificaram como negativos, p_{31} e p_{32} é a soma das colunas, p_{13} e p_{23} é a soma das linhas e p_{33} é o valor total.

Da tabela 1, temos que $p_o = p_{11} + p_{22}$ e $p_e = \frac{(p_{31} \cdot p_{13}) + (p_{32} \cdot p_{23})}{p_{33}^2}$. A interpretação do valor resultante Kappa está ilustrado na tabela 2.

Tabela 2 - Avaliação do coeficiente de concordância kappa.

Valor Kappa	Nível de concordância
< 0	Sem concordância
0,0 to 0,2	Concordância ruim
0,2 to 0,4	Concordância razoável
0,4 to 0,6	Concordância Boa
0,6 to 0,8	Concordância muito boa
0,8 to 1,0	Concordância excelente

Fonte: Autor

Essa técnica já foi utilizada em contextos semelhantes para comparação visual de sequências de RM das articulações sacroilíacas (SUNG et al., 2017).

3 RESULTADOS E DISCUSSÃO

3.1 SPAIR

A avaliação de cada classificador foi realizada utilizando métricas de área AUC, sensibilidade e especificidade com validação cruzada 10-*fold*. Para complementar a avaliação, também foram medidos os valores brutos de taxa de acerto e taxa de erro.

As Tabelas 3 e 4 contém os valores dessas métricas para cada um dos classificadores, sendo a primeira tabela os valores para os dados desbalanceados, contendo 22 positivos e 29 negativos e a segunda tabela para os dados balanceados, contendo 29 positivos e 29 negativos.

Tabela 3 - Métricas de avaliação para cada classificador com classes desbalanceadas. Os resultados em azul mostram os melhores valores para cada métrica, enquanto os vermelhos mostram os piores valores.

	Sensibilidade	Especificidade	AUC	% acerto	% erro
MLP	0,818	0,862	0,914	84,3	15,7
J48	0,682	0,69	0,716	68,6	31,4
NB	0,5	0,759	0,701	64,7	35,3
K=1	0,364	0,724	0,595	56,9	43,1
K=2	0,182	0,966	0,661	62,7	37,3
K=3	0,545	0,724	0,702	64,7	35,3
K=4	0,318	0,931	0,708	66,7	33,3
K=5	0,455	0,724	0,687	60,8	39,2
RF	0,591	0,897	0,824	76,5	23,5
SVM Linear	0,727	0,862	0,795	80,4	19,6

Fonte: Autor

Tabela 4 - Métricas de avaliação para cada classificador com classes balanceadas por SMOTE. Os resultados em azul mostram os maiores valores para cada métrica, enquanto os vermelhos mostram os menores valores.

	Sensibilidade	Especificidade	AUC	% acerto	% erro
MLP	0,897	0,897	0,954	89,7	10,3
J48	0,828	0,793	0,911	81,0	19,0
NB	0,483	0,759	0,681	62,0	38,0
K=1	0,828	0,621	0,703	72,4	27,6
K=2	0,483	0,828	0,695	65,5	34,5
K=3	0,759	0,586	0,739	67,2	32,8
K=4	0,655	0,828	0,752	74,1	25,9
K=5	0,69	0,69	0,747	69,0	31,0
RF	0,724	0,724	0,757	72,4	27,6
SVM Linear	0,931	0,897	0,914	91,4	8,6

Fonte: Autor

Na primeira tabela, nota-se que a rede neural artificial MLP obteve a maior taxa de acerto (84,3%), bem como a maior AUC (0,914), classificando de uma forma satisfatória a classe de menor prevalência, as positivas (sensibilidade de 0,818). Por outro lado, o classificador que obteve a melhor classificação de verdadeiros negativos, ou seja, especificidade, foi o KNN quando K=2 (especificidade de 0,966). Porém os valores de suas outras métricas indicam que houve uma forte tendência a tentar acertar a classe majoritária, podendo dizer que sua classificação foi próxima ao aleatório.

Quanto a avaliação realizada sobre os dados balanceados (Tabela 4), tanto a rede neural artificial MLP quanto o classificador SVM Linear obtiveram resultados altos. MLP superou o SVM Linear em AUC (0,954 contra 0,914), porém obteve menor sensibilidade (0,897 do MLP contra 0,931 do SVM Linear) possuindo uma especificidade igual em ambos os casos (0,897). Porém o SVM Linear obteve uma taxa de acerto maior que a do MLP (91,4% contra 89,7%). Esse tipo de comportamento não era esperado pois o MLP obteve uma performance menor ou equivalente ao SVM em sensibilidade e especificidade e maior em AUC, portanto uma avaliação mais profunda foi realizada, a fim de buscar verificar se os dois

métodos apresentaram diferenças estatisticamente significativas para a base balanceada.

Para a análise estatística, foi definido um intervalo de confiança de 95% para a sensibilidade e especificidade, como mostrado na Figura 10.

Figura 10 - Intervalo de confiança de 95% para sensibilidade e especificidade dos classificadores SVM e MLP.

Fonte - Autor

O intervalo de confiança de 95% para a sensibilidade do SVM foi (0,723;0,973) e para o MLP (0,75;0,987), enquanto para a especificidade do SVM foi (0,715;0,972) e para o MLP (0,715;0,972).

Nota-se que os intervalos de confiança se sobrepõem quase que por completo, sendo assim podemos assumir que as performances dos dois classificadores não apresentam nenhuma diferença estatística e então, apesar das diferenças nas métricas, ambos possuem performances similares.

3.2 SELEÇÃO DE ATRIBUTOS

Como descrito na seção 2.5, 3 métodos de seleção de atributos foram

utilizados: ReliefF, *Wrapper* e análise univariante utilizando o teste estatístico Mann-Whitney U. A sequência de ressonância magnética utilizada nesta análise foi a SPAIR e a análise foi realizada utilizando dados balanceados pelo SMOTE.

O método ReliefF retorna uma lista dos atributos ordenada por relevância, sendo assim, a análise foi realizada de forma exaustiva para todos os N atributos, onde $0 < N < 231$ avaliando a sensibilidade, especificidade e AUC para cada teste e utilizando os classificadores MLP, SVM, NB, RF, J48.

As figuras 11, 12 e 13 ilustram, respectivamente, a AUC, sensibilidade e especificidade para cada classificador

Figura 11 - Valores de AUC utilizando o método de seleção de atributos ReliefF.

Fonte: Autor.

Figura 12 - Valores de sensibilidade utilizando o método de seleção de atributos ReliefF.

Fonte: Autor.

Figura 13 - Valores de especificidade utilizando o método de seleção de atributos ReliefF.

Fonte: Autor.

O maior valor obtido de AUC foi do MLP (0.973) utilizando 45 atributos, seguido pela RF, com AUC de 0.971 utilizando 64 atributos. Para a sensibilidade, o MLP (0.965 com 64 atributos) e o SVM (0.965 com 112 atributos) tiveram performances semelhantes, diferindo apenas no número de atributos utilizados. Para especificidade, o MLP (0.931 com 64 atributos), o SVM (0.931 com 32 atributos) e a RF (0.931 com 33 atributos) apresentaram performances semelhantes, diferindo apenas no número de atributos utilizados.

O método *Wrapper* realiza a seleção de atributos utilizando um classificador como esquema de aprendizado. Nesse contexto, foi realizada uma análise utilizando 4 classificadores: NB, MLP, J48 e RF a fim de avaliar a performance de cada classificador com o subconjunto de atributos definidos por eles. Também foi

realizada uma análise para definir quais atributos foram comuns para cada classificador e qual as suas performances utilizando esses subconjuntos de atributos. Essa avaliação foi realizada com e sem o balanceamento utilizando o método SMOTE e as métricas utilizadas foram AUC, sensibilidade, especificidade.

A figura 14 ilustra a interseção da seleção de atributos de cada um dos classificadores. As tabelas 5 e 6 mostram as performances de cada classificador para cada subconjunto de atributos.

Figura 14 - Intersecção dos subconjuntos de atributos agrupados por classificador utilizando o método *Wrapper*.

Fonte: Autor.

Os 8 atributos selecionados em comum para os 4 classificadores são: Média da quarta direcionalidade de Tamura, média da sétima direcionalidade de Tamura, desvio padrão da décima primeira direcionalidade de Tamura, desvio padrão do coeficiente de correlação máximo, média da assimetria, média da energia de Wavelet LH2, média da energia de Wavelet HH2, desvio padrão da energia de Wavelet HH2.

Tabela 5 - Métricas de cada classificador sem SMOTE utilizando somente o subconjunto selecionado por ele e utilizando o subconjunto de intersecção.

	AUC		Sensibilidade		Especificidade	
	Wrapper do classificador	Intersecção	Wrapper do classificador	Intersecção	Wrapper do classificador	Intersecção
MLP	0,911	0,951	0,818	0,864	0,897	0,966
NB	0,892	0,879	0,864	0,818	0,897	0,897
J48	0,682	0,737	0,682	0,545	0,828	0,724
RF	0,893	0,884	0,727	0,818	0,966	0,897

Fonte: Autor

Tabela 6 - Métricas de cada classificador com SMOTE utilizando somente o subconjunto selecionado por ele e utilizando o subconjunto de intersecção.

	AUC		Sensibilidade		Especificidade	
	Wrapper do classificador	Intersecção	Wrapper do classificador	Intersecção	Wrapper do classificador	Intersecção
MLP	0,943	0,937	0,828	0,828	0,897	0,931
NB	0,919	0,904	0,862	0,862	0,897	0,828
J48	0,82	0,807	0,759	0,69	0,828	0,897
RF	0,81	0,95	0,793	0,828	0,828	0,862

Fonte: Autor

Por fim, para avaliar a confiabilidade de cada um dos métodos anteriormente descritos, o teste Mann-Whitney U foi realizado para atribuir significância estatística a cada um dos atributos ($p < 0,001$) e uma intersecção dos subconjuntos de atributos dos 3 métodos foi feita. Para essa análise, foram utilizados os 8 atributos selecionados pelos 4 classificadores utilizando *Wrapper*, os atributos cuja relevância atribuída pelo ReliefF foi maior que 0,05 (6 atributos) e os atributos estatisticamente significativos definidos pelo teste Mann-Whitney U. A figura 15 ilustra a intersecção de todos os atributos.

Figura 15 - Intersecção entre os 3 métodos de seleção de atributos.

Fonte: Autor.

A intersecção entre os 3 métodos resulta em 4 atributos, que são: Média da assimetria, média da energia de wavelet de Haar LH2, média da energia de wavelet de Haar HH2 e desvio padrão da wavelet de Haar HH2.

Os 4 atributos de intersecção indicam uma redundância nos subconjuntos de atributos selecionados pelos 3 métodos, sugerindo que os subconjuntos são, de fato, os mais relevantes para o contexto da classificação da sacroilíte.

3.3 SELEÇÃO DE ATRIBUTOS E APRENDIZADO DE MÁQUINAS PARA SEQUÊNCIA STIR

A avaliação de cada classificador foi realizada utilizando métricas de AUC, sensibilidade e especificidade com validação cruzada 10-*fold*.

O vetor de atributos utilizado foi composto por 230 componentes, representativos de características de níveis de cinza, textura, espectral e fractal.

Foi realizada a seleção de atributos utilizando o método ReliefF e um teste

exaustivo dos classificadores com os subconjuntos de atributos selecionados.

As figuras 16, 17 e 18 ilustram, respectivamente, a AUC, sensibilidade e especificidade.

Figura 16 - Valores de AUC utilizando o método de seleção de atributos ReliefF.

Fonte: Autor.

Figura 17 - Valores de Sensibilidade utilizando o método de seleção de atributos ReliefF.

Fonte: Autor.

Figura 18 - Valores de Especificidade utilizando o método de seleção de atributos ReliefF.

Fonte: Autor

A tabela 7 mostra o maior valor alcançado por cada classificador para cada uma das métricas e quantos atributos foram necessários para isso.

Tabela 7 - Maiores valores de cada métrica para cada classificador.

Classificador	Métrica	Valor	Número de atributos
SVM	AUC	0,867	2
	Sensibilidade	0,850	2
	Especificidade	0,960	1
KNN com K = 1	AUC	0,900	6
	Sensibilidade	0,850	6
	Especificidade	0,923	6
KNN com K = 3	AUC	0,932	5
	Sensibilidade	0,850	3
	Especificidade	1,000	24
KNN com K = 5	AUC	0,915	2
	Sensibilidade	0,800	5
	Especificidade	1,000	17
MLP	AUC	0,926	158
	Sensibilidade	0,85	150
	Especificidade	0,923	16

Fonte: Autor

O método de seleção de atributos *Wrapper* também foi utilizado. Para cada classificador um subconjunto de atributos foi obtido. A avaliação foi feita utilizando AUC, Sensibilidade e Especificidade e os resultados estão ilustrados na tabela 8.

Tabela 8 - Métricas para o subconjunto de atributos selecionado pelo Wrapper.

Classificador	AUC	Sensibilidade	Especificidade	Número de atributos
SVM	0,842	0,800	0,885	15
KNN com K = 1	0,798	0,800	0,769	13
KNN com K = 3	0,867	0,750	0,885	14
KNN com K = 5	0,969	0,750	0,962	9
MLP	0,965	1,000	0,923	6

Fonte: Autor

Em geral, os classificadores obtiveram bons valores com baixa dimensionalidade quando utilizado o método ReliefF. KNN com K = 3 obteve o maior valor de AUC (0,932) usando apenas 5 atributos, uma diferença significativa quando comparado com a segunda maior performance, alcançada pelo MLP (0,926) utilizando 158 atributos.

Utilizando o método *Wrapper*, KNN com K = 5 obteve a maior AUC (0,969) usando 9 atributos. Seguido pelo MLP, que obteve AUC de 0,956 utilizando 6 atributos. Porém, o classificador MLP obteve o maior valor de sensibilidade (1,000), uma grande diferença comparado com KNN com K = 5 (0,750).

Comparando os dois métodos de seleção, ReliefF alcançou o maior valor de AUC para o menor subconjunto de atributos com SVM (0,867 usando 2 atributos apenas), e KNN K = 5 (AUC de 0,915 utilizando 2 atributos). Por outro lado, o método *Wrapper* alcançou o maior valor de AUC com KNN K = 5 (0,969), sugerindo que, para uma melhor taxa de acerto, o *Wrapper* supera o ReliefF, porém para uma menor dimensionalidade o ReliefF supera o *Wrapper*.

3.4 COMPARAÇÃO ENTRE AS SEQUÊNCIAS SPAIR E STIR

A comparação das sequências SPAIR e STIR foi realizada utilizando o classificador perceptron multicamadas com todo o conjunto de atributos e também com o subconjunto de atributos selecionados pelo método de seleção *Wrapper*. Essas técnicas foram utilizadas pois o intuito era tornar a análise robusta utilizando

as informações obtidas com os resultados apresentados na seção 3.1 e 3.2, onde o MLP mostrou-se mais ou igualmente eficiente em relação aos outros classificadores.

As amostras em SPAIR e STIR diferem de tamanho, portanto um critério de inclusão para essa análise foi definido a fim de viabilizar a análise estatística, somente pacientes que possuam as amostras em SPAIR e STIR serão utilizados. Sendo assim, 46 pacientes, 20 positivos e 26 negativos, foram incluídos para essa análise.

Para avaliar a concordância entre a classificação utilizando a sequência SPAIR e STIR para os conjuntos de atributos, o coeficiente Kappa de concordância foi utilizado e uma tabela de concordância foi calculada para cada conjunto de atributos. A tabela 9 mostra a concordância para o conjunto de todos os atributos.

Tabela 9 - Tabela de concordância para todos os atributos.

		SPAIR		
		P	N	Total
STIR	P	13	7	20
	N	4	22	26
Total		17	29	46

Fonte: Autor

O coeficiente Kappa calculado utilizando a tabela 5 obteve valor de 0,505 com intervalo de confiança tendo limite inferior de 0,219 e superior de 0,791. O nível de concordância, segundo os valores de referências na tabela 2, dá evidências de que a concordância entre os métodos é boa.

Porém, é intuitivo pensar que a performance do classificador afeta diretamente o valor da concordância, pois se este tiver uma taxa de acerto de 100% para os dois métodos, a concordância seria perfeita.

Sendo assim, com o intuito de aumentar as evidências de que os métodos são concordantes, a mesma avaliação foi feita utilizando atributos selecionados pelo *Wrapper*. A lista de atributos se encontra na tabela 10 e os valores da concordância calculada são mostrados na tabela 11.

Tabela 10 - Atributos selecionados pelo *Wrapper* para as sequências SPAIR e STIR.

SPAIR	STIR
Média do contraste de Tamura	Média da primeira direcionalidade de Tamura
Média da oitava direcionalidade de Tamura	Desvio padrão para a trigésima direcionalidade de Tamura
Média da trigésima direcionalidade de Tamura	Média da soma da variância
Desvio Padrão do máximo coeficiente de desvio	Média do valor máximo dos pixels
Média da obliquidade	Média da segunda banda de frequência da wavelet de Gabor
Média da energia em segundo nível da wavelet de Haar (HL2)	Média da energia em segundo nível para a wavelet de Haar (LH2)
Média da energia em segundo nível para wavelet de Haar (LH2)	
Desvio Padrão da energia em terceiro nível para a wavelet de Haar (HH3)	
Fonte: Autor	

Tabela 11 - Tabela de concordância para os atributos selecionados pelo método *Wrapper*.

		SPAIR		
		P	N	Total
STIR	P	18	4	22
	N	2	22	24
	Total	20	26	46

Fonte: Autor

O coeficiente Kappa calculado utilizando a tabela 11 obteve valor de 0,738 com intervalo de confiança tendo limite inferior de 0,450 e superior de 1,000 . O nível de concordância, segundo os valores de referências na tabela 2, dá evidências de que a concordância entre os métodos aumentou de boa para muito boa, podendo alcançar até a concordância perfeita no melhor caso devido ao limite superior do intervalo de confiança. Porém, o fato de que somente um atributo selecionado foi

igual para as duas sequências (Média da energia em segundo nível para a wavelet de Haar (LH2)) sugere que as sequências, embora similares em relação ao resultado final da classificação, apresentam dados diferentes quando avaliadas de forma quantitativa, por um método computacional.

3.5 APRENDIZADO PROFUNDO

Os resultados utilizando redes neurais profundas foram realizados através de testes visando adaptar os recursos computacionais exigidos e os disponíveis.

As imagens utilizadas para esses resultados foram de sequência SPAIR (51 pacientes) processadas pelo método Warp. Como os dados de entrada para os modelos de redes neurais profundas utilizam os valores de pixels das imagens, os pacientes foram tratados como um exame tridimensional (6 imagens por paciente). Os valores de pixel das imagens foram normalizados entre 0 e 1.

O modelo proposto foi inspirado no modelo LeNet (LECUNN; BENGIO, 1995), porém adaptada para entradas tridimensionais. O modelo é composto por uma camada convolucional com *kernel* 5x5x6, uma camada de *MaxPooling* com *kernel* 2x2 e *stride* 2x2, uma segunda camada convolucional 5x5x1 seguida de uma *MaxPooling* com *kernel* 2x2 e *stride* 2x2 e uma camada totalmente conectada de tamanho 500, seguida por outra totalmente conectada de tamanho 2 para predição da classe. Uma regularização de Dropout foi feita após a segunda convolução para controlar o *overfitting* que poderia ser causado pelo número de amostras. A figura 19 ilustra o modelo.

Figura 19 - Modelo de rede neural profunda implementado para o estudo.

Fonte: Autor.

O código que implementa esse modelo foi escrito em python utilizando o *framework* Keras e é mostrado na figura 20:

Figura 20 - Código em Python do modelo de rede neural profunda.

```

model = Sequential()
model.add(Conv3D(20, (5, 5, 6), activation='relu', input_shape=(1,64,64,6),
data_format='channels_first', name='conv3D_one'))
model.add(MaxPooling3D(pool_size=(2, 2, 1), strides=(2, 2, 1)))
model.add(Conv3D(50, (5,5,1),activation='relu'))
model.add(MaxPooling3D(pool_size=(2, 2, 1), strides=(2, 2, 1)))
model.add(Dropout(0.5))
model.add(Flatten())
model.add(Dense(500))
model.add(Activation('relu'))
model.add(Dense(nb_classes))
model.add(Activation('softmax'))
model.compile(loss='categorical_crossentropy', optimizer=keras.optimizers.SGD(lr=0.01,
momentum=0.0, decay=0.0, nesterov=False), metrics=['accuracy'])

```

Fonte: Autor

Os testes foram realizados com uma taxa de aprendizado fixa de 0.01 e tempo de treino de 100 épocas, avaliando a acurácia do modelo com um método de validação *hold-out*, dividindo 80% do conjunto de dados para treino e 20% para teste, totalizando 40 amostras para treino e 11 amostras para teste. Porém, para minimizar problemas causados pela falta de generalização devido a quantidade pequena de imagens, foram realizadas 5 avaliações alterando a divisão de treino e teste de forma aleatória. A tabela 12 mostra os valores de acurácia e a média dos testes.

Tabela 12 - Valores de acurácia por teste e acurácia média calculada.

Teste 1	Teste 2	Teste 3	Teste 4	Teste 5	Média
36.36%	81.82%	63.64%	81.82%	63.64%	64.85%

Fonte: Autor.

As limitações de utilizar modelos de aprendizado profundo, com uma pequena quantidade de exemplos e limitações de *hardware* que impedem um ajuste fino mais preciso dos hiperparâmetros, aparentemente estão refletidos nos valores de acurácia obtidos pelo modelo. A grande variabilidade entre os valores da acurácia indicam que o modelo é incapaz de generalizar dependendo do grupo de treino, ou seja, não converge para um ponto de ótimo local ou global.

A acurácia média indica, ainda, que apesar das limitações listadas, o modelo é capaz de atingir taxas de acerto acima do aleatório, sugerindo que, em condições

onde as limitações sejam reduzidas, o modelo pode ter desempenho melhor na caracterização da presença de sacroiliíte.

4 CONCLUSÃO

Nesse projeto foram implementados e utilizados algoritmos de extração de atributos quantitativos de imagens de ressonância magnética de sequência SPAIR e STIR. Os atributos extraídos são de níveis de cinza, textura, espectrais e uma dimensão fractal. Esses atributos deram origem a vetores que caracterizam cada um dos pacientes entre positivo e negativo para sacroiliíte.

Todas as imagens adquiridas dos exames de cada paciente foram segmentadas, processadas e anonimizadas.

Cada um dos vetores dos 51 pacientes para SPAIR e 46 para STIR foram utilizados como entrada em métodos de aprendizado de máquina com o intuito de classificar e comparar a performance desses métodos. Para tratar um problema comum do aprendizado de máquina foi utilizado o método SMOTE para balanceamento das classes, evitando que os classificadores apresentassem resultados contaminados por um viés da classe majoritária. Os métodos de seleção de atributos ReliefF, *Wrapper* e Teste Mann-Whitney U foram utilizados para reduzir a dimensionalidade e eliminar atributos ambíguos ou com ruídos, mostrando assim que são eficientes para incrementar a performance dos métodos de aprendizado de máquinas.

Um modelo tridimensional de rede neural convolucional profunda foi implementado para avaliar a viabilidade e performance e comparar com métodos tradicionais de aprendizado de máquinas.

Os resultados de melhor performance sem seleção de atributos foram obtidos com a rede neural artificial MLP quando se trata do vetor de atributos sem balanceamento e notou-se uma performance semelhante entre os classificadores MLP e SVM de kernel linear quando utilizado o balanceamento por SMOTE.

O método de aprendizado profundo não superou os métodos tradicionais de aprendizado de máquina. Isso pode ter sido causado pelas limitações de número de exemplos e *hardware*.

Os métodos de seleção de atributos mostraram que de fato o aumento da dimensionalidade insere ruídos na classificação. Os três métodos concordaram com os atributos selecionados e obtiveram as melhores performances, alcançando AUC de 0,973 com 45 atributos utilizando o MLP.

A análise de concordância dos diagnósticos feitos a partir das sequências SPAIR e STIR, utilizando o coeficiente kappa, indicaram que existe semelhança na capacidade de classificação baseada nas duas sequências para classificação computadorizada, que aumenta quando se realiza uma seleção de atributos, sugerindo que as duas sequências são equivalentes para avaliar a articulação sacroilíaca.

Esse estudo mostrou que é viável a utilização de métodos de aprendizado de máquina para criação de uma ferramenta de diagnóstico auxiliado por computador das articulações sacroilíacas.

REFERÊNCIAS

- AHA, D. W.; KIBLER, D.; ALBERT M, K. Instance-based learning algorithms. **Machine Learning**, v.6, n.1, p.37-66, Jan. 1991.
- ARAÚJO, J.M. et al. Assessment and certification of neonatal incubator sensors through an inferential neural network. **Sensors**, v.13, n.11, p.15613-15632, Nov. 2013. DOI:10.3390/s131115613.
- BOONEN, A. Socioeconomic consequences of ankylosing spondylitis. **Clinical and Experimental Rheumatology**, v.20, Suppl.28, p.s23-s26, 2002.
- BREIMAN, L. Random forests. **Machine Learning**, v.45, n.1, p.5-32, Oct. 2001.
- CHAWLA, N.V. et al. SMOTE: synthetic minority over-sampling technique. **Journal of Artificial Intelligence Research**, v.16, p.321-357, June 2002. DOI: <https://doi.org/10.1613/jair.953>.
- COHEN, J. A Coefficient of agreement for nominal scales. **Educational and Psychological Measurement**, v.20, n.1, p.20-37, 1960. DOI: <https://doi.org/10.1177/001316446002000104>.
- COSTA, I.P. et al. Avaliação do desempenho do BASDAI (Bath Ankylosing Spondylitis Disease Activity Index) numa coorte brasileira de 1.492 pacientes com espondiloartrites: dados do Registro Brasileiro de Espondiloartrites (RBE). **Revista Brasileira de Reumatologia**, v.55, n.1, p.48-54, jan./fev. 2015.
- DALTO, V. et al. MRI assessment of bone marrow oedema in the sacroiliac joints of patients with spondyloarthritis: is the SPAIR T2w technique comparable to STIR?. **European Radiology**, v.27, n.9, p.3669-3676, Feb. 2017. DOI: 10.1007/s00330-017-4746-7.
- ERICKSON, B.J. et al. Machine learning for medical imaging. **RadioGraphics**, v.37, n.2, p.505-513, Mar./Apr. 2017. DOI: 10.1148/rg.2017160130.
- FALEIROS, M.C. et al. Computer-aided classification of inflammatory sacroiliitis in resonance imaging. In: CARS 2017 - COMPUTER ASSISTED RADIOLOGY AND SURGERY INTERNATIONAL CONGRESS AND EXHIBITION, 31., 2017, Barcelona. CARS 2017 - Computer assisted radiology and surgery. **Proceedings...** Barcelona: NH Collection Constanza, 2017. p.s154.
- FERREIRA JUNIOR, J.R. et al. Radiomics association of quantitative CT features with lung cancer patterns. In: CARS 2018 - COMPUTER ASSISTED RADIOLOGY AND SURGERY INTERNATIONAL CONGRESS AND EXHIBITION, 32., 2018. **Proceedings...** Berlin: Elsevier, 2018.
- FRACTAL FOUNDATION. Disponível em: <cesso em: 30 ago. 2017.

GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND STATISTICS, 13., 2010, Sardinia. **Proceedings...** Sardinia: PMLR, 2010. v.9, p.249-256.

GONZALES, R.C.; WOODS, R.E. **Digital image processing**. Boston: Addison-Wesley, 1993.

HARALICK, R.M.; SHANMUGAM, K.; DINSTEN, I.H. Textural features for image classification. **IEEE Transactions on Systems man and Cybernetics**, v.SMC-3, n.6, p.610-621, 1973.

HEARST, M.A. Support vector machines. **IEEE Intelligent Systems**, v.13, n.4, p.18-28, July 1998.

IVAN, M. **How to simulate support vector machine (SVM) in R**. 2014. Disponível em: <<http://en.proft.me/2014/04/22/how-simulate-support-vector-machine-svm-r>>. Acesso em: 10/05/2018.

JFEATURELIB OPEN SOURCE PROJECT. Disponível em: <<https://github.com/locked-fg/JFeatureLib>>. Acesso em: 2 maio 2016.

KOHAVI, R.; JOHN, G.H. Wrappers for feature subset selection. **Artificial Intelligence**, v.97, n.1-2, p.273-324, Dec. 1997.

KONONENKO, I. Estimating attributes: Analysis and extensions of RELIEF. In: MACHINE LEARNING: ECML-94: EUROPEAN CONFERENCE OF MACHINE LEARNING, 1994, Catania. **Proceedings...** Berlin: Springer-Verlag, 1994. p.171-182.

KUNIMATSU, A. et al. Machine learning-based texture analysis of contrast-enhanced MR imaging to differentiate between glioblastoma and primary central nervous system lymphoma. **Magnetic Resonance in Medical Sciences**, 2018. DOI: 10.2463/mrms.mp.2017-0178.

LECUN, Y.; BENGIO, Y. Convolutional networks for images, speech, and time-series. In: ARBIB, M.A. (Ed.). *The Handbook of brain theory and neural networks*. Cambridge: MIT, 1995.

LIMA, P.C. **Wavelets: uma introdução**. Belo Horizonte: Departamento de Matemática; ICEX; UFMG, 2003.

LITJENS, G. et al. A Survey on deep learning in medical image analysis. **Medical Image Analysis**, v.42, p.60-88, 2017. DOI: 10.1016/j.media.2017.07.005.

MAKSYMOWYCH, W.P. et al. Spondyloarthritis research consortium of canada magnetic resonance imaging index for assessment of sacroiliac joint inflammation in ankylosing spondylitis. **Arthritis & Rheumatism**, v.53, n.5, p.703-709, Oct. 2005.

MASON, S.J.; GRAHAM, N.E. Areas beneath the relative operating characteristics

(ROC) and relative operating levels (ROL) curves: Statistical significance and interpretation. **Quarterly Journal of the Royal Meteorological Society**, v.128, p.2145-2166, 2002.

MONTANDON, C. et al. Sacroiliíte: avaliação por imagem. **Radiologia Brasileira**, v.40, n.1, p.53-60, jan./fev. 2007.

NASR-ESFAHANI, E. et al. Melanoma detection by analysis of clinical images using convolutional neural network. In: INTERNATIONAL CONFERENCE OF THE IEEE ENGINEERING IN MEDICINE AND BIOLOGY SOCIETY, 2016, Orlando. **Proceedings...** DOI: 10.1109/EMBC.2016.7590963.

NGUYEN, C.; WANG, Y.; NGUYEN, H.N. Random forest classifier combined with feature selection for breast cancer diagnosis and prognostic. **Journal of Biomedical Science and Engineering**, v.6, n.5, 2013. Article ID:31887. DOI: 10.4236/jbisc.2013.65070.

PATTERSON, J.; GIBSON, A. **Deep learning a practitioner's approach**. Newton: O'Reilly Media, 2017.

QUINLAN, J.R. **C4.5: programs for machine learning**. Burlington: Morgan Kaufmann, 1993.

REZENDE, S.O. **Sistemas inteligentes: fundamentos e aplicações**. Barueri: Manole, 2003.

RUDWALEIT, M. et al. Defining active sacroiliitis on magnetic resonance imaging (MRI) for classification of axial spondyloarthritis: a consensual approach by ASAS/OMERACT MRI group. **Annals of the Rheumatic Diseases**, v.68, n.10, p.1520-1527, Oct. 2009.

SAMPAIO-BARROS P. D. et al., Consenso brasileiro de espondiloartropatias: espondilite anquilosante e artrite psoriásica diagnóstico e tratamento - primeira revisão. **Revista Brasileira de Reumatologia**, v.47, n.4, p.233-242, jul./ago. 2007.

SIEPER, J. et al. The Assessment of SpondyloArthritis International Society (ASAS) handbook: a guide to assess spondyloarthritis. **Annals of the Rheumatic Diseases**, v.68, Suppl.2, p.ii1–ii44, June 2009.

SUNG, S. et al. MRI assessment of sacroiliitis for the diagnosis of axial spondyloarthropathy: comparison of fat-saturated T2, STIR and contrast enhanced sequences. **British Journal of Radiology**, v.90, n.1078, p.20170090, Oct. 2017.

TAMURA, H.; MORI, S.; YAMAWAKI, T. Textural features corresponding to visual perception. **IEEE Transactions on Systems, Man & Cybernetics**, v.8, n.6, p.460-473, 1978.

WITTEN, I. et al. **Data mining: practical machine learning tools and techniques**. 4thed. Burlington: Morgan Kaufmann, 2016.

ZHANG, D. et al. Content-based image retrieval using gabor texture features. In: IEEE PACIFIC-RIM CONFERENCE ON MULTIMEDIA: 2000 international symposium on multimedia information on processing, 2000, Sydney. **Proceedings...** Sydney: University of Sydney, 2000.

APÊNDICE A - PRODUÇÃO CIENTÍFICA ASSOCIADA À PESQUISA

Artigo de congresso apresentado no *Computed Assisted Radiology and Surgery (CARS)* 2017: **Computer-aided classification of inflammatory sacroiliitis in magnetic resonance imaging.**

Autores: Matheus Calil Faleiros, Eddy Jens Rivero Zavala, José Raniery Ferreira Junior, Vitor Faeda Dalto, Marcello Henrique Nogueira-Barbosa, Paulo Mazzoncini de Azevedo-Marques.

Artigo de congresso apresentado no *VI Eccomas Thematic Conference On Computational Vision And Medical Image Processing (VIPIMAGE)* 2017: **Pattern Recognition of inflammatory sacroiliitis in magnetic resonance imaging**

Autores: Matheus Calil Faleiros, Jose Raniery Ferreira Junior, Eddy Zavala Jens, Vitor Faeda Dalto, Marcello Henrique Nogueira-Barbosa, Paulo Mazzoncini de Azevedo-Marques.

Artigo de congresso apresentado no Congresso da Sociedade Brasileira de Computação (CSBC) 2017: **Reconhecimento computadorizado de padrões inflamatórios de sacroiliíte em imagens de ressonância magnética.**

Autores: Matheus Calil Faleiros, Jose Raniery Ferreira Junior, Eddy Zavala Jens, Vitor Faeda Dalto, Marcello Henrique Nogueira-Barbosa, Paulo Mazzoncini de Azevedo-Marques.

Artigo de congresso apresentado no *Computed Assisted Radiology and Surgery (CARS)* 2018: **Features selection analysis to quantify sacroiliitis in magnetic resonance imaging.**

Autores: Matheus Calil Faleiros, José Raniery Ferreira Junior, Priscilla Magalhães Tenório, Luppino Assad, Vitor Faeda Dalto, Marcello Henrique Nogueira-Barbosa, Paulo Mazzoncini de Azevedo-Marques.

APÊNDICE B - WEKA E EXEMPLOS DE CÓDIGO

Esta seção contém figuras do *software* de aprendizado de máquinas Weka e exemplos de códigos do processo de extração de atributos e do uso dos classificadores.

Figura B1 - Tela inicial do Weka contendo resumo dos dados, opções de filtros e aba de opções de tarefas, como classificação.

Fonte: Autor

Figura B2 - Aba de classificação do Weka mostrando as opções de classificadores.

Fonte: Autor.

Figura B3 - Exemplo de saída do Weka após a execução de um Perceptron Multicamadas e validação cruzada 10-fold.

Fonte: Autor

Figura B4 - Tela do Weka de seleção de atributos utilizando como saída o método ReliefF.

Fonte: Autor.

Dois processos importantes desse trabalho foram a extração de atributos a automação de testes de classificadores. Isso foi feito utilizando a biblioteca JFeatureLib e Weka, respectivamente. A seguir segue alguns trechos de códigos de exemplo para:

- Extração de atributos:

```
Tamura tamura;
image = new ColorProcessor(ImageIO.read(new File(PATH_TO_IMAGE)));
tamura = new Tamura();
tamura.run(image);
bw.write(Arrays2.join(tamura.getFeatures().get(0), ",")); //escreve no
arquivo como CSV
```

- Automação de testes utilizando Weka:

```

public void mlpMomentsLearningRate(){ //Teste automatizado variando momento e taxa
de aprendizado para MLP
    try{
        File output= new File(destiny);
        FileWriter fw = new FileWriter(output);
        BufferedWriter bw = new BufferedWriter(fw);
        BufferedReader reader = new BufferedReader(new FileReader(source));
        Instances data = new Instances(reader);
        reader.close();
        data.setClassIndex(data.numAttributes() - 1);

        //Normalizando
        Normalize norm = new Normalize();
        norm.setInputFormat(data);
        data = Filter.useFilter(data, norm);
        bw.write("Params, AUC, Sensibilidade, Especificidade");
        bw.newLine();
        for(double momentum = 0.1; momentum <= 1; momentum = momentum+0.1){
            for(double learningRate = 0.1; learningRate <= 1; learningRate =
learningRate+0.1){

                String[] options = weka.core.Utils.splitOptions("-M "+momentum+"
-L "+learningRate);
                MultilayerPerceptron mlp = new MultilayerPerceptron();
                mlp.setOptions(options);
                Evaluation eval = new Evaluation(data);
                eval.crossValidateModel(mlp, data, 10, new Random(1), new
Object[]{});

                double auc = eval.areaUnderROC(1);
                double sensitivity = eval.truePositiveRate(1);
                double especificity = eval.truePositiveRate(0);
                bw.write("-M "+momentum+" -L "+learningRate+",");
                bw.write(auc+", "+sensitivity+", "+especificity);
                bw.newLine();
                System.out.println("Executando..."+"-M "+momentum+" -L
"+learningRate);
            }
        }

        bw.close();
        reader.close();
    } catch (Exception ex) {
        ex.printStackTrace();
    }
}

```



```
}
```

- Código utilizado para listar e ordenar os exames de RM dos pacientes:

```
static List<String> images;
static void getImageFiles(File folder) {
    File folders[] = folder.listFiles();
    for (int i = folders.length - 1; i >= 0; i--) {
        if (folders[i].isDirectory())
            getImageFiles(folders[i]);
        else if (folders[i].isFile()) {
            String file = folders[i].getName();
            if (file.substring(file.length() - 4).equals(".png"))
                images.add(""+folders[i].getAbsolutePath());
        }
    }
}
```