UNIVERSIDADE FEDERAL DE SERGIPE

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

DEPARTAMENTO DE COMPUTAÇÃO

Classificação de linfócitos utilizando imagens microscópicas de células de pacientes com Leucemia Linfoide Aguda

Trabalho de Conclusão de Curso

José Elwyslan Maurício de Oliveira

São Cristóvão – Sergipe

2019

UNIVERSIDADE FEDERAL DE SERGIPE

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

DEPARTAMENTO DE COMPUTAÇÃO

José Elwyslan Maurício de Oliveira

Classificação de linfócitos utilizando imagens microscópicas de células de pacientes com Leucemia Linfoide Aguda

Trabalho de Conclusão de Curso submetido ao Departamento de Computação da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador(a): Daniel Oliveira Dantas

São Cristóvão – Sergipe

2019

Resumo

A contagem e classificação de células permite o diagnóstico de inúmeras doenças do tecido sanguíneo, dentre elas a Leucemia Linfoide Aguda (LLA), um câncer que afeta os linfócitos do sistema imunológico e que pode ser fatal se não tratada corretamente nas suas fases iniciais. A classificação dos linfócitos em saudáveis ou malignos geralmente é feita por patologistas treinados que, observando imagens microscópicas dos linfócitos do paciente, buscam por características citológicas e morfológicas que permitam classificá-los. Embora existam métodos avançados que podem realizar este diagnóstico, como a citometria de fluxo, eles ainda são caros e não estão disponíveis em todos os hospitais e clínicas de tratamento. Essa realidade faz com que a análise microscópica das células ainda seja um dos métodos utilizados. No entanto essa análise é demorada, tediosa e repetitiva para o profissional, além produzir resultados não uniformes e com baixa reprodutibilidade. Neste sentido, o emprego de técnicas de processamento de imagens em conjunto com algoritmos de classificação podem auxiliar os profissionais da área médica fornecendo informações sobre a morfologia celular dos linfócitos de forma mais padronizada e com resultados mais precisos, além de possibilitar a criação de sistemas de classificação automatizada capazes de diferenciar os linfócitos saudáveis dos malignos.

Este trabalho tem como objetivo aplicar técnicas de processamento de imagens e métodos de classificação em imagens de linfócitos extraídos de pacientes com LLA com a finalidade de classificá-los em saudáveis ou malignos. Para tal foram extraídos os seguintes descritores das imagens: estatísticas de 1ª ordem, características de textura, características morfológicas, função distância do centroide e coeficientes da Transformada Discreta do Cosseno. Cada conjunto de descritores foi utilizado com os algoritmos de classificação: Máquina de Vetores de Suporte, K-Vizinhos mais próximos, Floresta Aleatória e Redes Neurais. Também foram treinadas as redes neurais convolucionais *InceptionV3*, *Xception*, VGGNet e ResNet50 com as imagens brutas. Outro objetivo deste trabalho foi estudar o efeito que as variações celulares particulares dos linfócitos de cada paciente têm no desempenho dos classificadores. O conjunto de imagens utilizado é composto por 10661 imagens, das quais 7272 são linfócitos malignos e 3389 saudáveis, que foram extraídos de 47 com LLA e 26 pacientes saudáveis respectivamente. Foram avaliados os efeitos da técnica de *data augmentation* quando aplicada nas imagens.

Como resultado foi alcançada uma acurácia de 92,48% obtida pela rede convolucional VGG16. Entretanto, quando consideradas as variações celulares particulares de cada paciente, o maior resultado foi uma acurácia de 81,04%, obtida por uma rede neural treinada com as estatísticas de 1ª ordem. Ambos os classificadores foram treinados com imagens que passaram pelo procedimento de *data augmentation* que utilizou as técnicas de espelhamento, rotação, cisalhamento, ruído sal e pimenta e desfoque gaussiano.

Palavras-chave: Características de Textura, Características Morfológicas, *Data Augmentation*, Estatísticas de 1ª Ordem, Floresta Aleatória, Função Distância do Centroide, K-Vizinhos mais Próximos, Leucemia Linfoide Aguda, Máquina de Vetores de Suporte, Redes Neurais Convolucionais, Redes Neurais, Transformada Discreta do Cosseno.

Abstract

Counting and classifying cells allows the diagnosis of countless diseases of the blood tissue, among of them Acute Lymphoid Leukemia (ALL), a cancer that affects lymphocytes and can be fatal if not correctly treated in early stages. The classification of lymphocytes into healthy or malignant is usually performed by trained pathologists from observation of microscopic images of patient's lymphocytes. They look for for cytological and morphological features that allow a correct classification. Although there are advanced diagnostic methods, such as Flow Cytometry, they are still expensive and are not available in all hospitals and treatment clinics, this fact makes the microscopic analysis still one of most commonly used methods. However this analysis is time consuming, tedious and repetitive for the professional who perform it, besides producing non-uniform results and with low reproducibility. In this sense, the use of image processing techniques in combination with classification algorithms can help medical professionals by providing information on the cellular morphology of lymphocytes in a more standardized way and with more accurate results, besides allowing the creation of automated classification systems capable of differentiating between healthy and malignants lymphocytes.

This work aims to apply image processing techniques and classification methods to lymphocyte images extracted from ALL patients in order to classify them as healthy or malignant. For this purpose the following image descriptors were extracted: First Order Statistics, Texture Features, Morphological Featues, Centroid Distance Function and Discrete Transform of Cosine coefficients. Each set of descriptors was used with each of the following classification algorithm: Support Vector Machine, K-Nearest Neighbors, Random Forest and Neural Networks. Also was trained the convolutional neural networks: InceptionV3, Xception, VGGNet and ResNet50 using raw images. Another goal of this work was to study how patient's lymphocyte particularity affect the performance of classifiers. The dataset consists in 10,661 images, 7272 of which are healthy and 3389 malignant lymphocytes, which were extracted from 47 with ALL and 26 patients, respectively. The effects of the data augmentation technique when applied to the images were evaluated.

Was achieved an accuracy of 92,48% obtained by VGG16 Convolutional Network. However, when considering the particular cellular variations of each patient, the accuracy achived was 81,04% obtained by a Neural Network using first order statistics as feature vector. Both classifiers were trained with data augmentation images that used mirroring, rotation, shear, addition of salt and pepper noise, and Gaussian blur.

Keywords: Acute Lymphoblastic Leukemia, Centroid Distance Function, Convolutional Neural Networks, Data Augmentation, Discrete Cosine Transform, First Order Statistics, K-Nearest

Neighbors, Morphological Features, Neural Networks, Random Forest, Support Vector Machine, Texture Features.

Lista de ilustrações

Lista de quadros

Lista de tabelas

Lista de códigos

Lista de abreviaturas e siglas

ALL	Acute Lymphoblastic Leukemia
AML	Acute Myeloid Leukemia
BPNN	Back Propagation Neural Net
DCT	Discrete Cosine Transform
FAB	French-American-British
FN	Falso Negativo
FP	Falso Positivo
GLCM	Gray Level Co-occurence Matrix
GLDM	Grey Level Dependence Matrix
GLRLM	Gray Level Run Lenght Matrix
GLSZM	Gray Level Size Zone Matrix
HSV	Hue Saturation Value
IBSI	Image Biomarker Standardisation Initiative
ISBI	International Symposium on Biomedical Imaging
KNN	K Nearest Neighbors
L-SVM	Linear SVM
LLA	Leucemia Linfoide Aguda
LMA	Leucemia Mieloide Aguda
MLP	Multi Layer Perceptron
NGTDM	Neighbouring Gray Tone Difference Matrix
NN	Neural Network
P-SVM	Polinomial SVM
PCA	Principal Component Analysis
Q-SVM	Quadratic SVM

R-SVM	Radial Basis SVM
RBF	Radial Basis Function
RF	Random Forest
RGB	Red Green Blue
ReLU	Rectified Linear Unit
SBILab	Signal processing and Bio-medical Imaging Lab
SGLD	Spatially Gray Level Dependent
SVM	Support Vector Machine
Sig-SVM	Sigmoide SVM
VGG	Visual Geometry Group
VN	Verdadeiro Negativo
VP	Verdadeiro Positivo

Sumário

1

Introdução

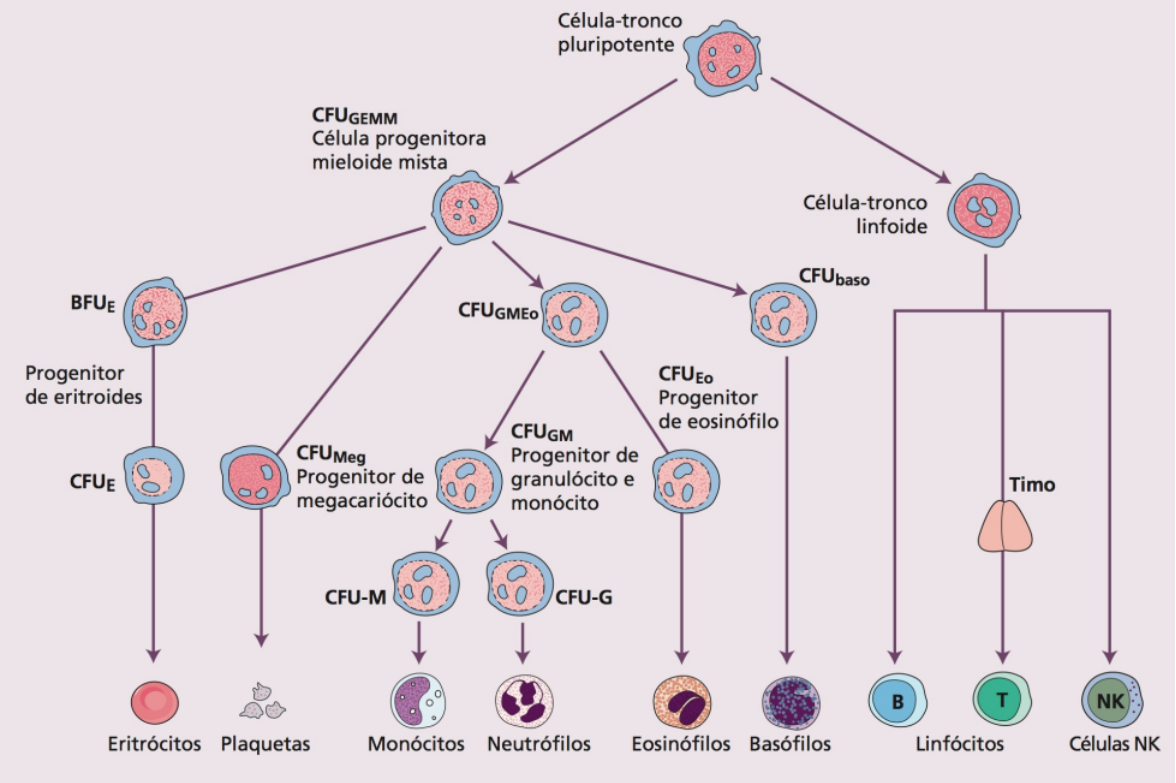
O sangue é um tecido biológico líquido que circula dentro do sistema vascular de animais que possuem sistemas circulatórios fechados. Dentre suas principais funções estão: o transporte de nutrientes, oxigênio, hormônios, anticorpos ou qualquer outra substância que seja essencial para o funcionamento dos órgãos irrigados por ele, bem como recolhe o gás carbônico ou qualquer outro resíduo produzido pelas atividades metabólicas do corpo e os entrega para os órgão responsáveis por sua excreção (LORENZI, 2006).

O tecido sanguíneo é composto por diversos tipos de células que juntas constituem a chamada "parte sólida do sangue". Cada tipo de célula possui anatomia e funções próprias e estão imersas no plasma, que é chamado de "parte líquida do sangue". O órgão central que produz as células do tecido sanguíneo é a medula óssea. Nela que estão localizadas as células-tronco pluripotentes, responsáveis por produzir células novas continuamente e liberá-las no sangue. O termo pluripotente surge da sua capacidade de se diferenciar para vários tipos de células. Diferenciação é o processo no qual células pluripotentes se especializam e se transformam num tipo diferente de célula. Na medula óssea uma célula-tronco pluripotente pode se diferenciar para uma célula progenitora mieloide mista ou para uma célula-tronco linfoide (LORENZI, 2006).

A célula progenitora mieloide mista tem a capacidade de se diferenciar para três linhagens diferentes, são elas: a Eritrocitária, que dará origem aos Eritrócitos (Hemácias), a Megacariocitária, que dará origem as plaquetas, e a Granulocítica, que dará origem ao Granulócitos, que podem se diferenciar mais uma vez em Basófilos, Eosinófilos, Mastócitos e Neutrófilos. Já a célula-tronco linfoide pode se diferenciar e dar origem a Linfócitos tipo T (timo-dependentes), Linfócitos tipo B (bursa-símile-dependentes) ou para as células NK (*Natural Killer*) (LORENZI, 2006; HOFFBRAND; MOSS, 2013).

Todo este processo de origem das células sanguíneas é conhecido na Hematologia como Hemopoese e pode ser visto na Figura 1

Figura 1 – Hemopoese: origem das células do sangue



Fonte: Hoffbrand e Moss (2013, p. 3)

A etimologia do termo Leucócito provém das palavras gregas *leuco*, que significa “branco” e *cito*, que significa “célula”, por este motivo é comum encontrar o termo "células brancas"dentro da literatura de Histologia. Representam uma linhagem de células que fazem parte do sistema imunológico do ser humano e são responsáveis por combater e proteger o corpo humano da ação de agentes infecciosos e contra a invasão de corpos estranhos no organismo. Sob essa denominação incluem-se vários tipos celulares que, morfologicamente e funcionalmente, diferenciam-se entre si (JUNQUEIRA; CARNEIRO, 2004; LORENZI, 2006).

Na hematologia denominam-se Anomalias Leucocitárias as alterações de forma e função dos leucócitos em geral, as Leucemias fazem parte destas anomalias. Definem-se pelo acúmulo de mieloblastos (células imaturas da linhagem granulocítica) ou de linfoblastos (linfócitos imaturos) na medula óssea e sangue periférico, elas são classificadas em quatro tipos: agudas e crônicas, que, por sua vez, se subdividem em linfoides ou mieloides. As Leucemias agudas são as mais agressivas pois evoluem rapidamente e apresentam seus sintomas de forma mais intensa em comparação com sua versão crônica (HOFFBRAND; MOSS, 2013).

A Leucemia Linfoide Aguda (LLA), objeto principal de estudo neste trabalho, é caracterizada por uma grande quantidade de linfoblastos na médula óssea e no sangue periférico que não se diferenciam até a suas formas mais maduras e normais. Uma LLA pode ser da linhagem B quando os linfócitos B são malignos, ou pode ser da linhagem T quando os linfócitos T são

malignos (LORENZI, 2006; VERRASTRO et al., 2006). É a Leucemia mais comum na infância. Dos casos registrados desta doença a maior ocorrência está nas crianças entre 3 e 7 anos de idade, com 75% dos diagnósticos ocorrendo antes dos 6 anos. Destes diagnósticos, 85% são da linhagem B. (HOFFBRAND; MOSS, 2013, p. 224).

Os linfócitos malignos são classificados de acordo com seus aspectos morfológicos, químicos, fenotípicos e genéticos. A contagem, o monitoramento e à análise destes linfócitos no paciente é feita durante todo o tratamento da Leucemia, estes dados auxiliam a equipe médica na escolha da terapia mais adequada para o paciente, bem como medem a eficácia do tratamento contra a doença (ONCOGUIA, 2018; LOURDES CHAUFFAILLE; MIHOKO, 2019).

Devido a este aspecto, em 1976 pesquisadores Franceses, Americanos e Britânicos realizaram uma ação conjunta para definir uma nomenclatura padrão para os leucócitos das Leucemias Agudas, este trabalho teve como resultado o artigo de Bennett et al. (1976) o qual define a chamada classificação FAB, abreviatura para *French–American–British*. Essa classificação separa os tipos diferentes de linfócitos baseado na categoria da Leucemia e nas suas características citológicas. Para a Leucemia Mieloide Aguda a FAB classifica os linfócitos em oito grupos: M0, M1, M2, M3, M4, M5, M6 e M7, já para Leucemia Linfoide Aguda só existem três grupos: L1, L2 e L3.

As células do tipo L1 têm todos os blastos de tamanho pequeno à médio e são bastante uniformes na sua aparência. Células maiores têm sua cromatina difusa e às vezes pequenos nucléolos, enquanto as blastos menores não têm nucléolo visível e mostram alguma condensação de cromatina. O citoplasma é escasso e fracamente a moderadamente basofílico. Pode haver alguns vacúolos citoplasmáticos. As células do tipo L2 apresentam os blastos maiores e mais pleomórficos (capacidade de variar sua forma de acordo com o ciclo de vida) com núcleos mais irregulares, nucléolos mais ressaltados e citoplasma mais abundante. As células do tipo L3 são caracterizadas por basofilia citoplasmática moderadamente intensa e vacuolação citoplasmática variável, mas geralmente intensa (VERRASTRO et al., 2006; BAIN, 2015).

Embora existam métodos avançados para o diagnóstico da LLA, como a Citometria de Fluxo, eles ainda são muito caros e não estão amplamente disponíveis em laboratórios de hematologia e é por este motivo que, muitas das vezes, a classificação dos linfoblastos entre normal e os tipos L1, L2 e L3 é realizada por patologistas, que fazem a análise microscópica de amostras de sangue extraídas da médula óssea do paciente na busca por características morfológicas e citológicas que possam permitir a classificação correta dos linfoblastos. Este tipo de análise tem como desvantagem ser tediosa, demorada e repetitiva para o patologista, e que produz resultados pouco uniformes, com baixa precisão e reprodutibilidade (VERRASTRO et al., 2006; BAIN, 2015; SBILAB, 2019).

O emprego de técnicas de processamento de imagens pode auxiliar os profissionais da área médica fornecendo informações sobre a morfologia celular dos linfócitos de forma mais padronizada e com resultados mais precisos. Na área da Imagiologia Médica (*Medical Imaging*)

existem vários artigos publicados em periódicos que tratam da utilização dessas técnicas em conjunto com algoritmos de aprendizagem de máquina e Inteligência Artificial (SVM, *K-Nearest-Neighbors*, *Random Forest*, Redes Neurais e Redes Convolucionais) para propor metodologias de classificação automatizada para este problema.

Levando-se em consideração esses aspectos, o principal objetivo deste trabalho foi o de avaliar o desempenho dos métodos de classificação presentes na literatura de Imagiologia Médica que fazem uso de imagens microscópicas de linfócitos e técnicas de Processamento de Imagem para identificar linfócitos malignos e normais de pacientes acometidos com Leucemia Linfoide Aguda. Foi proposta uma metodologia para o desenvolvimento de classificadores para este problema que, posteriormente, foi aplicada e teve os seus resultados avaliados.

A estrutura deste trabalho consiste em (por ordem de capítulos): **Revisão Bibliográfica**, onde é analisado o estado na arte em processamento de imagens na tarefa de classificação de linfócitos LLA; **Fundamentação Teórica**, que aborda os principais conceitos teóricos utilizados no desenvolvimento deste trabalho; **Metodologia**, que apresenta os recursos utilizados, critérios e métodos utilizados para o desenvolvimento dos classificadores; **Resultados**, que apresenta os resultados obtidos com a metodologia aplicada; e **Conclusões** que comenta os resultados obtidos e orienta o desenvolvimento de trabalhos futuros.

2

Revisão Bibliográfica

Neste capítulo é apresentada uma revisão bibliográfica das técnicas e métodos que são empregados na classificação dos linfócitos da LLA e que foram publicados em Artigos e Jornais especializados na área de processamento de imagens médicas (*Medical Imaging*).

Uma primeira proposta de classificação automatizada para linfócitos malignos e normais foi feita por Suryani et al. (2014) que apresentaram um sistema capaz de classificar um linfócito entre normal e os 11 tipos diferentes de linfócitos malignos que ocorrem nas Leucemias agudas: L1 a L3 para ALL e M0 a M7 para AML. Seu sistema de classificação utiliza três características morfológicas extraídas das imagens das células, são elas: a área da célula (computada em *pixels* e convertida para micrômetros quadrados), a proporção do núcleo (computada dividindo área do núcleo pela área da célula) e a granularidade do núcleo (computada dividindo o número de *pixels* dos grãos, ver 2, pelo numero de *pixels* do núcleo).

Figura 2 – A seta vermelha aponta para os grãos

Fonte: Suryani et al. (2014, p. 42)

Após extraídas, essas características são avaliadas por um algoritmo baseado em lógica *Fuzzy* que é responsável por classificar o linfócito entre normal ou um dos seus 11 tipos malignos. Essa metodologia obteve uma acurácia de 83,65% utilizando um conjunto de teste com 104 imagens sendo: 29 ALL positivo, 50 AML positivo e 25 de linfócitos normais.

No mesmo ano Putzu et al. (2014) também desenvolveram um sistema de classificação automatizada, no entanto seu trabalho se limitou aos 3 tipos de linfócitos da ALL (L1 a L3). Seu artigo propõe a extração de características morfológicas relacionadas ao contorno e formato das células como: perímetro, área convexa, eixo maior, eixo menor, orientação, elongação, excentricidade entre outras. Também foram utilizadas características associadas a textura dos linfócitos, extraídas da Matriz de Co-ocorrência de Níveis de Cinza (GLCM - *Grey Level Co-occurrence matrix*), calculada usando a imagem do linfócito em escala de cinza, são exemplos destas características: autocorrelação, contraste, correlação, energia, entropia, homogeneidade entre outras. Como estratégias de classificação foram utilizados: SVM com diferentes tipos de Kernel (linear, quadrático, polinomial e Gaussiano), *K-Nearest Neighbors* (KNN) e *Random Forest* (RF). Em seus testes o classificador que obteve melhor resultado foi o que utilizou SVM com um *kernel* Gaussiano que apresentou 92% de acurácia e 98% de sensibilidade. O conjunto de teste foi composto por 267 imagens, destas, foi possível identificar corretamente a classe de 245.

Em 2016 MoradiAmin et al. (2016) propuseram uma metodologia para classificar os linfócitos da ALL em 4 grupos: não-cancerígenos, L1, L2, e L3. Análogo ao que foi proposto por Putzu et al. (2014), porém só foram utilizados classificadores baseados em SVM com tipos diferentes de Kernel. Outra diferença entre os dois trabalhos é que neste os resultados individuais de cada classificador são utilizados por um algoritmo de votação, responsável por dar a classificação definitiva para o linfócito.

Para realizar a classificação são extraídas as características morfológicas a partir das imagens de linfócitos de forma semelhante ao que foi feito por Suryani et al. (2014) e Putzu et al. (2014), porém são adicionas as estatísticas de 1ª e 2ª ordens ao conjunto de características que são utilizadas na classificação. As estatísticas de 1ª ordem são computadas dos histogramas dos canais vermelho, verde e azul no espaço de cores RGB e dos histogramas dos canais matiz, saturação e valor no espaço de cores HSV de cada imagem. Dentre as estatísticas de 1ª ordem calculadas de cada histograma estão: média, desvio padrão, energia, entropia, assimetria, curtose entre outras. As estatísticas de 2ª ordem dão informações sobre as posições dos diferentes níveis de cinza que compõem a imagem e são extraídas das matrizes SGLD (*Spatially Gray Level Dependent*) e, assim como as estatísticas de 1ª ordem, foram computadas dos histogramas de cada canal da imagem nos espaços de cores RGB e HSV. Dentre as estatísticas de 2ª ordem calculadas estão: energia, correlação, inércia, entropia, diferença média entre outras. Foram utilizadas 958 imagens e, para cada uma, foram extraídas 238 características, isto gerou um conjunto de dados com dimensão 238x958.

No intuito de reduzir o número de dimensões Putzu et al. (2014) utilizaram o método de Análise de Componentes Principais (PCA). A escolha do número ótimo de componentes para a redução de dimensionalidade foi feita comparando o desempenho dos classificadores utilizando entre 3 e 15 componentes. Os resultados mostraram que o melhor desempenho ocorre com uso

de 13 componentes principais. Realizando a redução da dimensionalidade, o conjunto de dados passou de 238x958 para apenas 13x958.

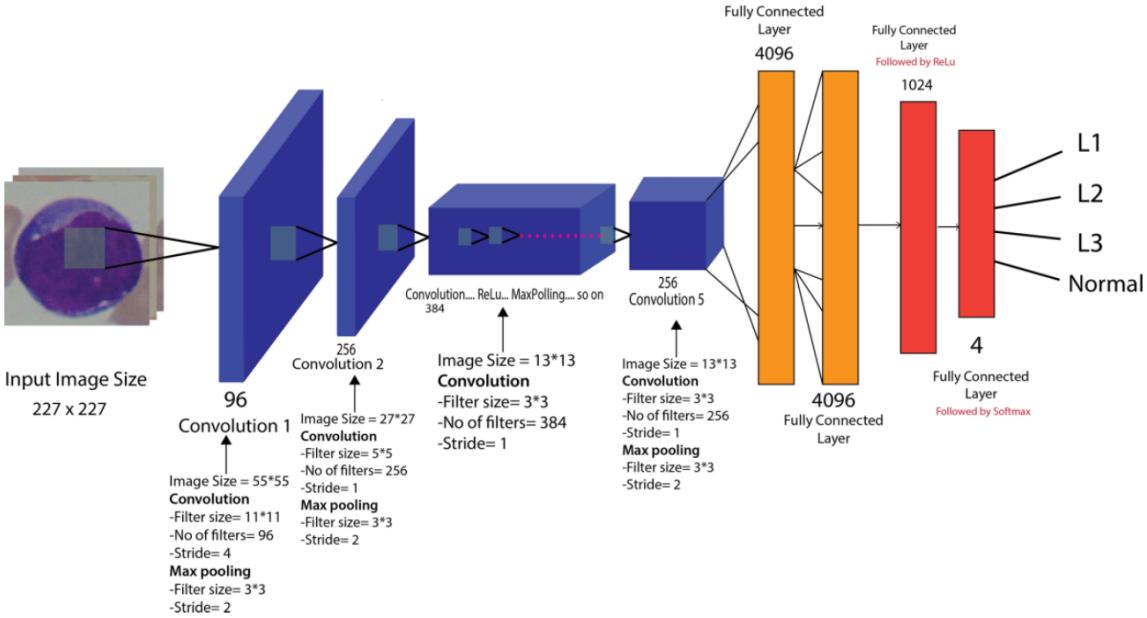
Com o conjunto de características reduzido foram desenvolvidos 5 classificadores baseados em SVM com *kernels*: linear, quadrático, polinomial, RBF (*Radial Basis Functions*) e MLP (*Multi-Layer Perceptron*). Os resultados parciais de cada classificador alimentam o algoritmo de votação por Maioria Absoluta, responsável por computar o grupo que o linfócito pertence. O desempenho médio deste método é: para células L1, 96,76% de acurácia e 94,23% de precisão; para células L2, 96,50% de acurácia e 92,67% de precisão; para células L3, 98,85% de acurácia e 96,73% de precisão; e para células não-cancerígenas, 98% de acurácia e 97,50% de precisão.

Ainda em 2016, com o objetivo de classificar os linfócitos em malignos ou benignos Mishra et al. (2016) utilizaram a Transformada Discreta do Cosseno - DCT (*Discrete Cosine Transform*) para extrair as características dos linfócitos e aplicá-las em quatro métodos de classificação diferentes: *Naive Bayes Classifier*, *K-Nearest Neighbor*, BPNN (*Back-Propagation Neural Network*) e SVM. O melhor resultado foi obtido utilizando SVM, o qual alcançou 89,76% de acurácia. No ano seguinte, Mishra et al. (2017) apresentaram outra metodologia para classificação dos linfócitos da LLA que utiliza somente as características extraídas da matriz GLCM, calculadas de suas imagens microscópicas. É usado o método de classificação *Random Forest* para construir o classificador que obtém como resultado uma acurácia de 96,29%.

Em 2018 encontramos o artigo de Shafique e Tehsin (2018) no qual eles fazem uso da arquitetura Alexnet (KRIZHEVSKY et al., 2012) pré-treinada com as imagens do desafio *ImageNet* em conjunto com a técnica de *transfer learning* para treinar a rede na tarefa de classificação dos linfócitos da LLA em L1, L2, L3 e Normal. Com essa metodologia foi alcançada uma sensibilidade de 100% e especificidade de 98,11% na classificação entre células normais e malignas. Nos subtipos de linfócitos foi alcançada uma sensibilidade de 96,74% e especificidade de 99,03%.

O principal ponto a ser notado neste trabalho é que o uso de Redes Neurais Convolucionais tornou dispensável a etapa de extração das características dos linfócitos, presente nas metodologias anteriores, a própria Rede Neural recebe a imagem bruta, realiza o seu processamento nas suas camadas internas e apresenta em sua saída a classificação final do linfócito, como pode ser visto na Figura 3.

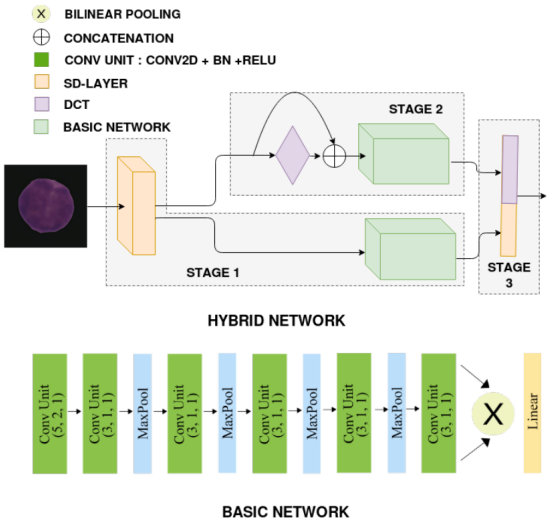
Figura 3 – Arquitetura Alexnet modificada por Shafique e Tehsin (2018) para classificação dos subtipos de linfócitos da LLA



Fonte: Shafique e Tehsin (2018, p. 4)

Outra metodologia que aplica a teoria de Redes Convolucionais para classificar os linfócitos da LLA foi feita por Mourya et al. (2018), que projetaram sua própria arquitetura, a *LeukoNet*, e fizeram o seu treinamento a partir do zero. Internamente a sua estrutura é uma rede híbrida que combina características extraídas das imagens com o uso da Transformada Discreta do Cosseno em conjunto com outras retiradas do espaço *Optical Density*, e que são usadas em conjunto com camadas convolucionais como pode ser visto em 4.

Figura 4 – Arquitetura da *LeukoNet*



Fonte: Mourya et al. (2018)

Neste mesmo estudo Mourya et al. (2018) aponta que as variações celulares que são particulares de cada paciente podem, de alguma forma, ser importantes no desempenho do classificador quando este for testado com células de paciente diferentes. Partindo desta premissa suas imagens de treino, validação e teste foram separadas de modo que a rede Convolucional fosse treinada com células de alguns pacientes e testados com a de outros.

Com essa metodologia a arquitetura *LeukoNet* obteve 89,7% de acurácia utilizando um conjunto com 13739 imagens sendo: 9211 ALL positivo e 4528 de linfócitos normais.

Ao colocar todos estes trabalhos lado a lado nota-se um padrão comum na abordagem do problema de classificação dos linfócitos da LLA que é, a extração de características dos linfócitos através de técnicas de processamento de imagens, seguida da sua utilização em algum método de aprendizagem de máquina. A exceção mostra-se somente em estudos mais recentes que, por utilizarem Redes Neurais Convolucionais, não necessitam da etapa de extração de características. No próximo capítulo serão apresentados os fundamentos teóricos que foram base para o desenvolvimento deste trabalho.

3

Fundamentação Teórica

Neste capítulo serão apresentados os principais conceitos teóricos abordados no decorrer deste trabalho. Primeiramente na seção 3.1 serão tratados os algoritmos de aprendizagem de máquina, na seção 3.2 serão apresentados os conceitos referentes as Redes Neurais Convolucionais e algumas de suas arquiteturas, na seção 3.3 será apresentado o método de redução de dimensionalidade baseado na Análise de Componentes Principais (PCA), na seção 3.4 serão apresentados os descritores de imagens padronizados pela *Image Biomaker Standardisation Initiative* (IBSI) que foram utilizados neste trabalho, nas seções 3.5, 3.6 e 3.7 serão descritas, respectivamente, as características morfológicas, de contorno e da transformada discreta do cosseno e, por fim, na seção 3.8 serão apresentadas as métricas utilizadas para avaliar o desempenho de classificadores binários.

3.1 Métodos de Aprendizagem de Máquina

Segundo Shalev-Shwartz e Ben-David (2014), de maneira formal, um cenário que envolve Aprendizagem de Máquina é composto pelas seguintes partes:

Domínio — Um conjunto arbitrário X, cujo os elementos pertencem a uma ou mais classes distintas. Geralmente os elementos deste domínio são representados por um vetor de características (*feature vector*) que contém os seus principais atributos.

Rótulos (*Labels*) — O conjunto Y que contém os rótulos de todas as classes possíveis dos elementos do Domínio. Geralmente em problemas de classificação binária utiliza-se $Y = \{0, 1\}$ ou $Y = \{-1, 1\}$.

Conjunto de Treinamento — O conjunto S, uma sequência finita de pares $X \times Y$ na forma $S = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$, subconjunto do Domínio

cujos elementos tem a sua classe conhecida e associada a seu respectivo Rótulo.

Modelo — O modelo é uma função $f : X \rightarrow Y$ chamada de Preditor ou Classificador. O modelo recebe como entrada os elementos do Domínio e como saída infere o Rótulo da classe a qual estes elementos pertencem.

Neste trabalho foram utilizados quatro métodos de Aprendizagem de Máquina para construção de modelos, são eles: Máquina de Vetores de Suporte (SVM) tratado na subseção 3.1.1, K-vizinhos mais próximos (KNN) tratado na subseção 3.1.2, Floresta Aleatória (RF) tratado na subseção 3.1.3 e Redes Neurais Multicamadas tratado na subseção 3.1.4.

3.1.1 Máquina de Vetores de Suporte

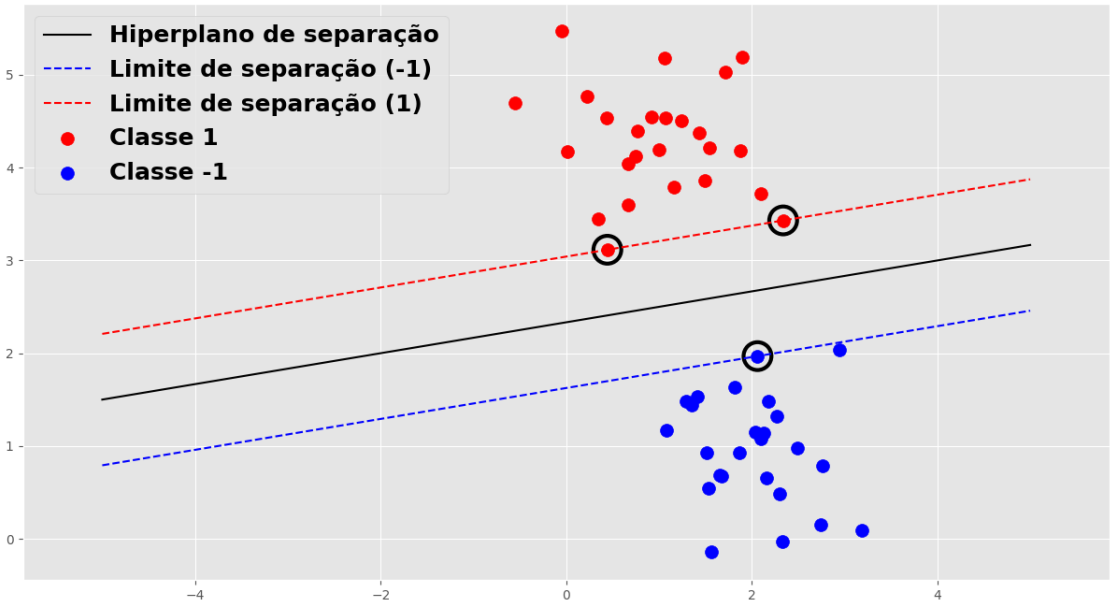
A Máquina de Vetores de Suporte (SVM) é um método de Aprendizagem de Máquina proposto por Cortes e Vapnik (1995) que pode ser usado em tarefas de classificação, regressão ou detecção de *outliers* (valores aberrantes ou atípicos dentro de um conjunto de dados) (SCIKIT-LEARN, 2019). Originalmente o SVM foi desenvolvido para resolver problemas de classificação binária, no entanto existem maneiras de utilizá-lo em problemas que envolvam mais de duas classes (DUAN; KEERTHI, 2005).

Dado um conjunto de treinamento com seus respectivos rótulos, o algoritmo SVM busca o seguinte mapeamento linear

$$y_i = f(x_i, W, b) = W \cdot x_i + b\,, \tag{3.1}$$

onde y_i e x_i são, respectivamente, o rótulo e o vetor de características do i-ésimo elemento do conjunto de treinamento, W e b são parâmetros da função f que serão otimizados para encontrar o melhor hiperplano que separe as classes do conjunto de treinamento. Um exemplo de hiperplano em $\mathbb{R}^2$ que separa duas classes distintas pode ser visto na Figura 5.

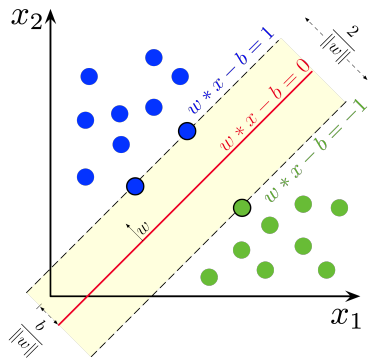
Figura 5 – Exemplo de hiperplano criado pelo algoritmo de SVM que separa 2 classes distintas. Os vetores de suporte das duas classes estão circulados.



Os vetores de suporte são definidos pelos elementos do conjunto de treinamento que estão na margem de separação entre as duas classes, é com relação a estes vetores que os parâmetros W e b são otimizados para que se obtenha a maior distância possível entre o hiperplano e os vetores de suporte das classes.

Considere o cenário ilustrado na Figura 6.

Figura 6 – Problema de Otimização do SVM



Fonte: Wikimedia Foundation, Inc. (2019)

Seja $w \cdot x - b = 1$ o hiperplano de separação da classe $y = 1$, e $w \cdot x - b = -1$ o hiperplano de separação da classe $y = -1$. A distância entre estes dois hiperplanos será dada por $\frac{2}{||w||}$. Para maximizar a distância entre estes dois hiperplanos será necessário minimizar $||w||$. Entretanto os

valores que w pode assumir possuem duas restrições:

$$w \cdot x - b \geq 1, \text{ se } y = 1$$
$$w \cdot x - b \leq -1, \text{ se } y = -1 \ .$$

Estas restrições podem ser reescritas como:

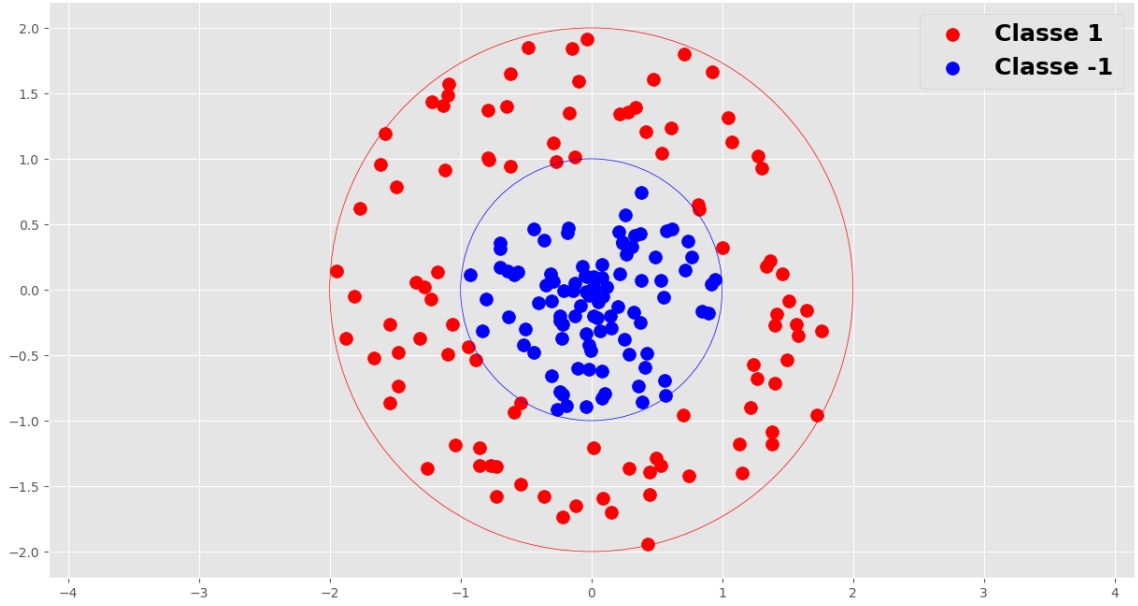
$$y(w \cdot x - b) \geq 1 \ . \tag{3.2}$$

Desta forma os parâmetros w e b do hiperplano de separação serão obtidos a partir da solução do problema de otimização que busca maximizar a distância entre os hiperplanos de separação ($||w||$) definidos pelos vetores de suporte das classes como mostrado na equação 3.3.

$$\text{argmin}_{w,b} \, ||w||, \text{ sujeita a restrição } y(w \cdot x - b) \geq 1. \tag{3.3}$$

Devido ao fato do SVM ser um classificador linear, ele não apresenta bons resultados quando o conjunto de treinamento não é linearmente separável (SHALEV-SHWARTZ; BEN-DAVID, 2014). Um exemplo é quando os elementos de uma das classes estão contidos dentro de um círculo de raio 1, e os elementos da outra classe estão contidos na região compreendida entre um círculo de raio 1 e outro de raio 2, neste caso não há um hiperplano em $\mathbb{R}^2$ que consiga separar as duas classes de maneira satisfatória. Este cenário pode ser visto na Figura 7.

Figura 7 – Exemplo de duas classes não linearmente separáveis

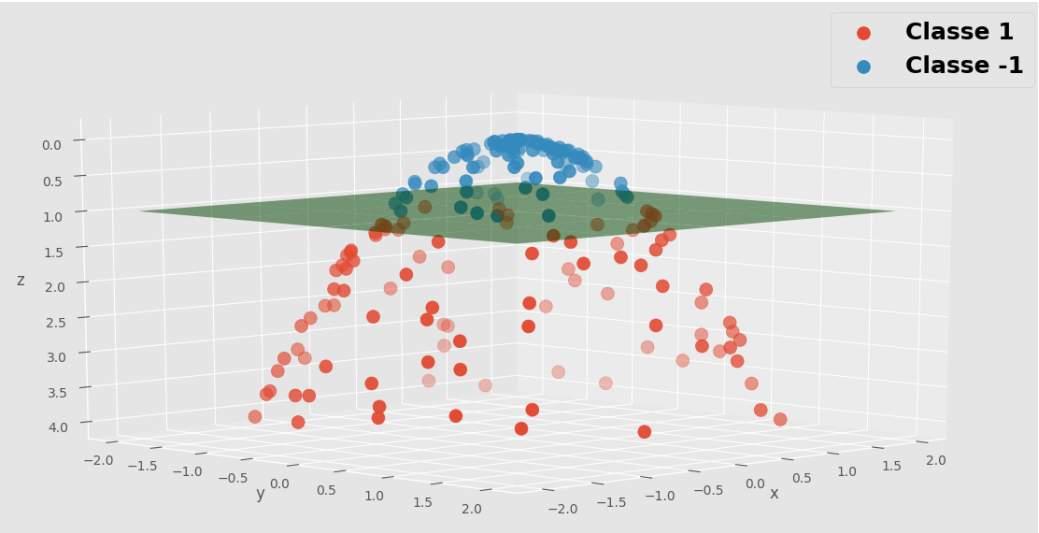


Este problema pode ser contornado com o mapeamento destes elementos para um espaço no qual eles sejam linearmente separáveis. Para o exemplo mostrado na Figura 7, considere a função $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ que faz o seguinte mapeamento

$$\phi(\,(x_1,\ x_2)\,) = (x,\ y,\ z) = (x_1,\ x_2,\ x_1^2 + x_2^2)\,. \tag{3.4}$$

Aplicando a equação 3.4 nos elementos da Figura 7 os elementos tornam-se linearmente separáveis em $\mathbb{R}^3$ pelo hiperplano $z = 1$ como pode ser visto na Figura 8.

Figura 8 – Exemplo de dados linearmente separáveis em $\mathbb{R}^3$ após a aplicação do mapeamento da equação 3.4



Na literatura de Aprendizagem de Máquina o processo de mapear os elementos do conjunto de treinamento de um espaço em que não sejam não linearmente separáveis para um espaço em que sejam é feito com o uso de funções *kernel* pelo método do Truque do *Kernel* (*kernel trick*). Os *kernels* mais utilizados com SVM são: Linear, Polinomial, *Radial Basis Function* (RBF) e Sigmoide (SHALEV-SHWARTZ; BEN-DAVID, 2014).

3.1.2 K-vizinhos mais próximos

O K-Vizinhos Mais Próximos (KNN) é considerado um dos algoritmos mais simples em Aprendizagem de Máquina (SHALEV-SHWARTZ; BEN-DAVID, 2014). A estratégia deste algoritmo é armazenar em memória todos os elementos do conjunto de treinamento para então classificar um elemento do Domínio baseado na sua proximidade ou semelhança em relação aos elementos do conjunto de treinamento (os "vizinhos"mais próximos do elemento a ser classificado). O único parâmetro deste classificador é a quantidade de "vizinhos"que devem ser computados na classificação.

Seja X um Domínio no qual cada um de seus elementos está associado a uma dada classe, seja $Y = \{y_1, y_2, \cdots, y_m\}$ o conjunto que contém os Rótulos das classes possíveis dos elementos de X e seja $\rho : X \times X \rightarrow \mathbb{R}$ a função que calcula a distância entre dois elementos de X. Se $X \in \mathbb{R}^d$ e ρ é a função que calcula a distância euclidiana entre dois elementos, x e x', então

$$\rho(x, x') = ||x - x'|| = \sqrt{\sum_{i=1}^{d} (x_i - x'_i)^2}\,. \tag{3.5}$$

Seja $S = \{(x_1,\ y_1),\ (x_2,\ y_2),\ \cdots,\ (x_m,\ y_m)\}$ o conjunto de treinamento, e x um elemento do Domínio sem classificação. A sequência das distâncias entre x e cada um dos elementos de S será definida como

$$\left(\pi_i\right)_{i=0}^{m} = \left(\pi_1(x),\ \pi_2(x),\ \cdots,\ \pi_m(x)\right) = \left(\rho(x,\ x_1),\ \rho(x,\ x_2),\ \cdots,\ \rho(x,\ x_m)\right). \tag{3.6}$$

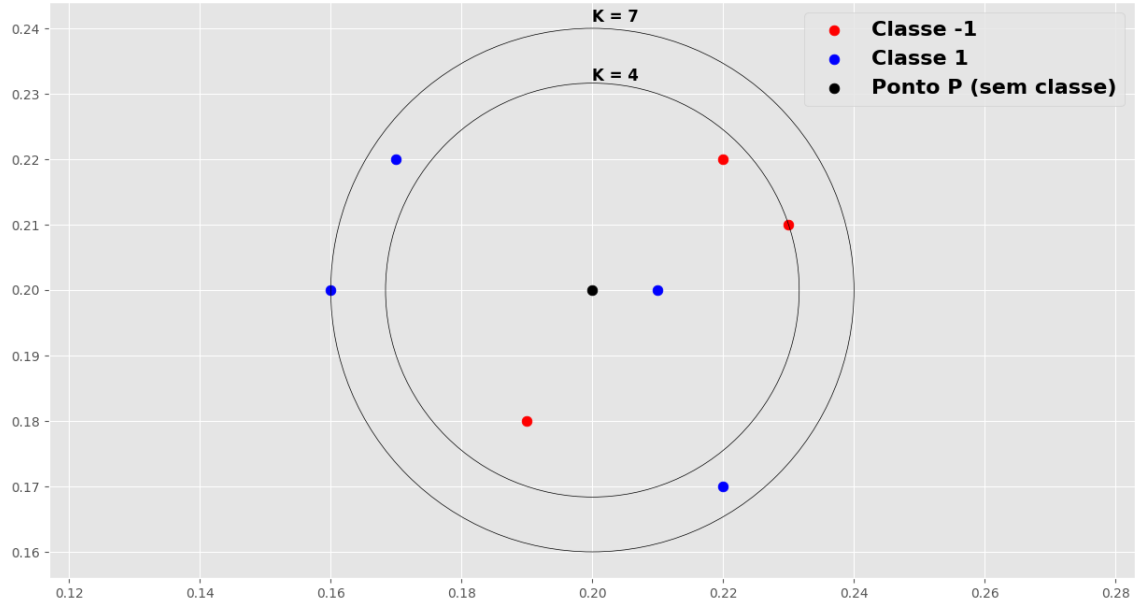
Para um dado número K, a classe de um elemento x do domínio será inferida da forma mostrada no pseudocódigo 1.

Código 1 – Pseudocódigo do procedimento de KNN

Entrada: Um elemento x do Domínio para ser classificado
Entrada: O número K de vizinhos que devem ser considerados
Saída: O rótulo da classe a qual x pertence
início
  Calcular as distâncias entre x e cada um dos elementos do conjunto de treinamento;
  Identificar as classes dos K elementos do conjunto de treinamento mais próximos de x (menores distâncias);
  Computar a frequência das classes identificadas;
  retorna *O rótulo da classe que obteve a maior frequência*;
fim

Por exemplo, considere o conjunto de pontos em $\mathbb{R}^2$ que pertencem a duas classes distintas $Y = \{-1, 1\}$ e um ponto P do qual deseja-se inferir a classe, dispostos da forma que é mostrada na Figura 9.

Figura 9 – Exemplo de classificação utilizando o K-vizinhos mais próximos



Para $K = 4$ o algoritmo KNN irá inferir que o ponto P pertence a classe -1, visto que dos 4 pontos mais próximos de P, três pertencem a classe -1 e somente um pertence a classe 1.

Para $K = 7$ o algoritmo KNN irá inferir que o ponto P pertence a classe 1, visto que dos 7 pontos mais próximos de P, três pertencem a classe -1 e quatro pertencem a classe 1.

3.1.3 Floresta Aleatória

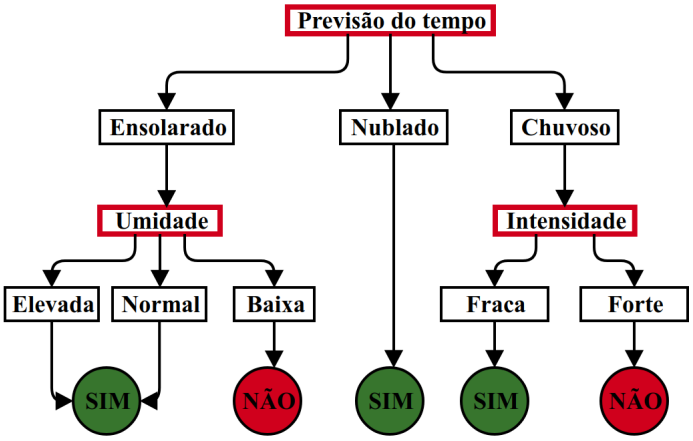
Uma árvore de decisão é um classificador que infere a classe de um elemento do Domínio percorrendo uma árvore da sua raiz até uma de suas folhas, a classe é definida pela folha alcançada ao fim deste percurso. O caminho da raiz até a folha comumente é definido por operações de *thresholding* realizadas nos nós internos da árvore de decisão utilizando os valores do vetor de características do elemento a ser classificado (SHALEV-SHWARTZ; BEN-DAVID, 2014).

Um exemplo de árvore de decisão que, dado um elemento do Domínio e seu respectivo vetor de características com os atributos

$$x_i = \{\text{Previsão do Tempo; Umidade do ar; Intensidade da chuva}\},$$

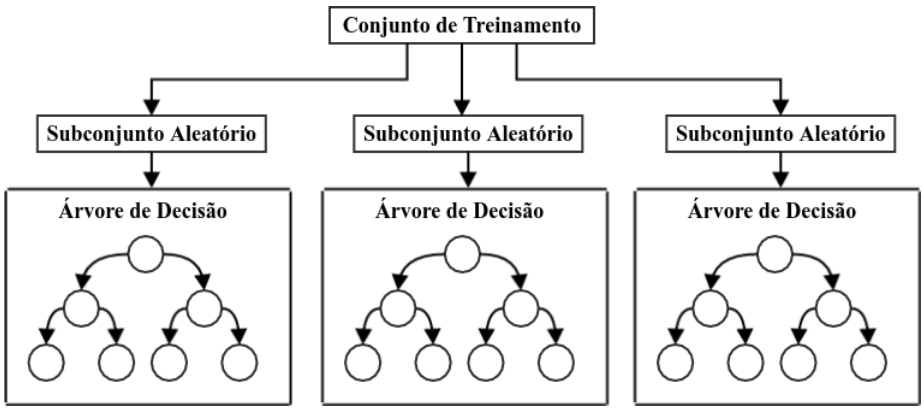
e que classifica se o dia é favorável ou não para realização de uma partida de tênis, $Y = \{\text{Sim; Não}\}$, pode ser vista na Figura 10.

Figura 10 – Exemplo de uma árvore de decisão



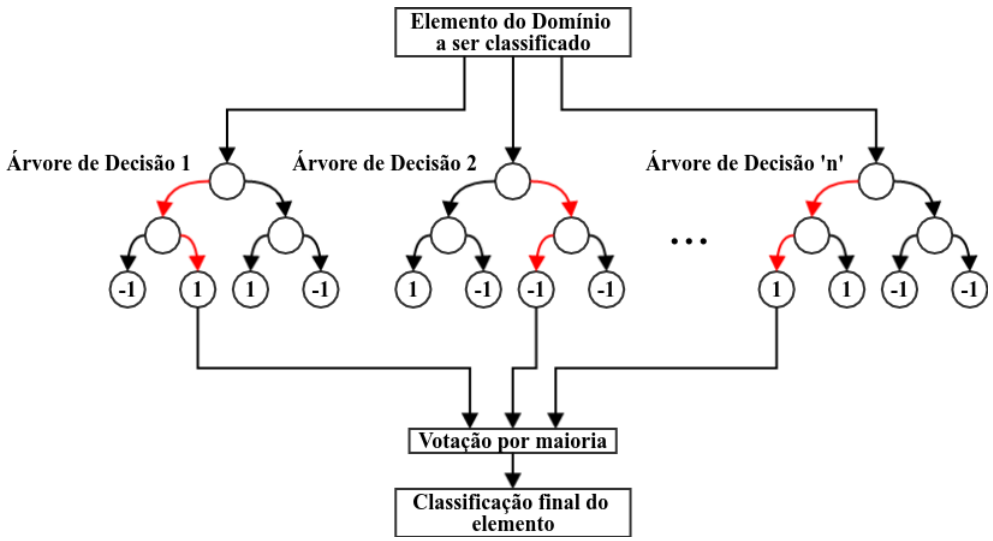
Uma Floresta aleatória é uma coleção de árvores de decisão em que a predição é realizada pela votação por maioria dos resultados de todas as árvores. O algoritmo de Floresta Aleatória cria subconjuntos aleatórios com os elementos do vetor de características e cada um destes subconjuntos dá origem a uma árvore de decisão como ilustrado na Figura 11. Este processo geralmente leva a criação de modelos melhores (SHALEV-SHWARTZ; BEN-DAVID, 2014).

Figura 11 – Subconjuntos de treinamento aleatórios para construção de uma Floresta Aleatória



Após construída a floresta, cada novo elemento do domínio a ser classificado é avaliado por cada uma das árvores de decisão e por meio de uma votação por maioria dos seus resultados a classe do elemento é inferida como ilustrado na Figura 12. O único parâmetro deste classificador é a quantidade de árvores aleatórias que serão criadas.

Figura 12 – Classificação utilizando uma Floresta Aleatória

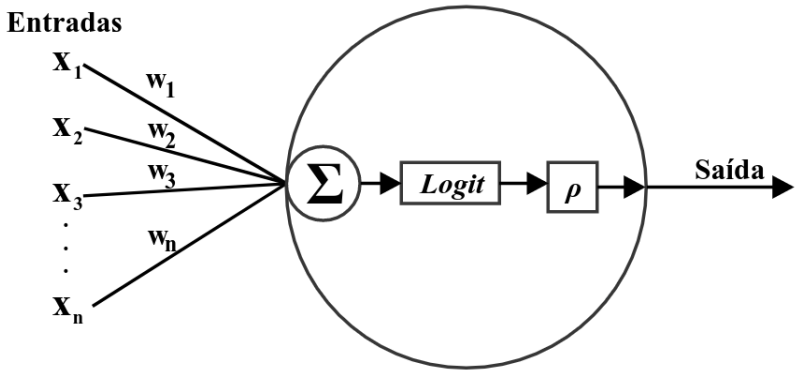


3.1.4 Redes Neurais

Um rede neural é um modelo computacional inspirado na estrutura neural do cérebro humano. Um modelo simplificado do cérebro consiste numa grande quantidade de pequenos dispositivos computacionais conectados entre si que são capazes de executar tarefas de alta complexidade computacional. De maneira formal uma rede neural pode ser descrita como um grafo $G = (V, E)$ em que os vértices (V) correspondem aos neurônios e as arestas (E) as ligações entre eles, e os neurônios estão conectados em uma rede do tipo alimentação avante (*feed forward*), ou seja, o grafo é acíclico (não contém ciclos) (SHALEV-SHWARTZ; BEN-DAVID, 2014).

O grafo $G = (V, E)$ tem um "peso" associado a cada aresta, e cada neurônio recebe um número n de entradas, x_1, x_2, ..., x_n, que são multiplicadas por seus respectivos pesos w_1, w_2, ..., w_n. A soma das entradas ponderadas dos neurônios é chamado de *logit* ($\sum_{i=0}^{n} w_i x_i$). O valor *logit* passa por uma função $\rho : \mathbb{R} \to \mathbb{R}$ que calcula o valor da saída do neurônio. A função ρ é chamada "função de ativação" do neurônio. A representação simplificada do modelo de um neurônio numa rede neural pode ser visto na Figura 13.

Figura 13 – Modelo simplificado de um neurônio



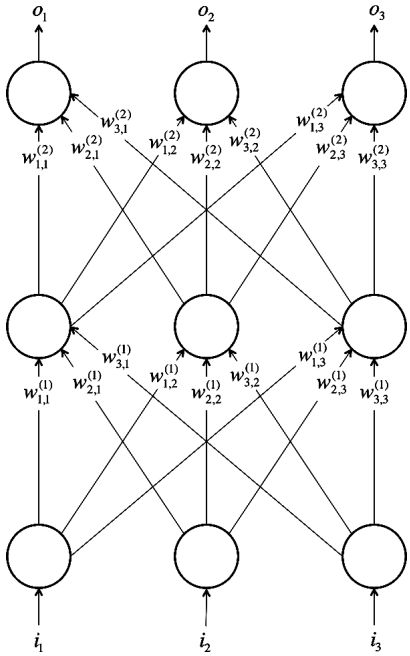
Desta forma, um único neurônio realiza o cálculo expresso pela equação

$$y = \rho(w \cdot x + b)\,, \tag{3.7}$$

onde b é o termo chamado de *bias*, um parâmetro opcional responsável por adicionar um *offset* ao *logit*.

Uma rede neural é dividida em camadas, e cada camada é um subconjunto dos neurônios do grafo $G(V, E)$ que compartilham as mesmas propriedades. A primeira camada é chamada de camada de entrada, é considerada a "camada mais baixa" da rede neural (*bottom layer*) e é responsável por receber os valores que serão computados pela rede. As camadas intermediárias são chamadas de "camadas ocultas" (*hidden layers*), a primeira camada oculta recebe como entrada as saídas da camada de entrada, realiza sua computação e repassa os valores para a próxima camada oculta. A camada de saída é considerada "a mais alta" da rede (*top layer*), recebe como entrada os valores de saída da última camada oculta e é responsável por calcular a saída final da rede neural. A ilustração de uma rede neural pode ser vista na Figura 14

Figura 14 – Exemplo de uma rede neural multicamadas



Fonte: Buduma (2017, p. 10)

Caso a função de ativação do neurônio seja

$$\rho(x) = \begin{cases} 1 & \text{se } (w \cdot x + b) \geq 0 \\ 0 & \text{se } x \leq 0 \end{cases} \tag{3.8}$$

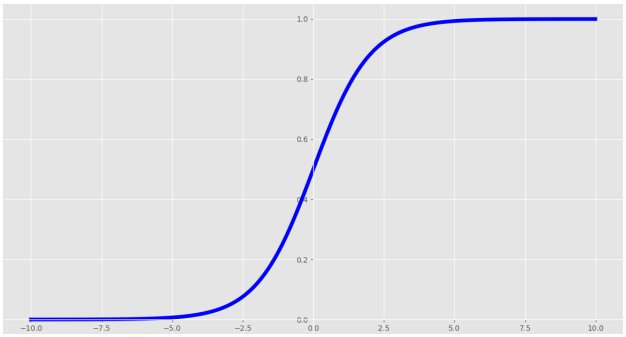
o neurônio recebe o nome de *Perceptron*, entretanto na prática essa função não é muito utilizada. Atualmente as três principais funções de ativação usadas em redes neurais são: Sigmoide, Tangente Hiperbólica e ReLU (LECUN et al., 2015; BUDUMA, 2017; STANFORD UNIVERSITY, 2017).

A função sigmoide é definida como

$$\rho(z) = \frac{1}{1 + e^{-z}} \quad . \tag{3.9}$$

Quando utilizada como função de ativação seu efeito é o de limitar a saída do neurônio para valores entre 0 e 1. *Logits* muito negativos tornam-se 0 e *logits* muito positivos tornam-se 1. Sua adoção deu-se pelo fato de sua curva modelar com mais fidelidade a ativação de um neurônio real, em que 0 seria o neurônio inativo e 1 o neurônio ativo, porém de maneira mais suave quando comparado com a equação do *Perceptron*. Entretanto o seu desempenho em termos de velocidade de aprendizagem da rede, estabilização dos valores da saída e saturação dos neurônios é inferior quando comparado com as funções tangente hiperbólica e ReLU.

Figura 15 – Curva característica da função sigmoide no intervalo -10 a 10

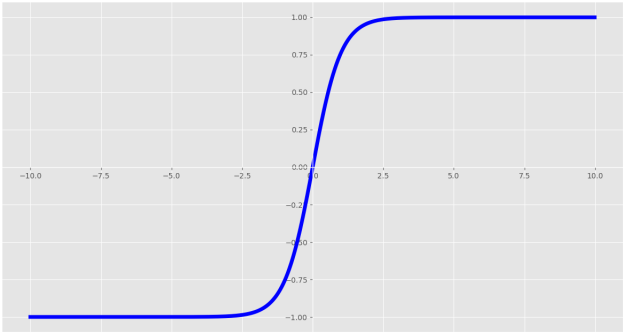


A função tangente hiperbólica é definida como

$$\rho(z) = tanh(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \quad . \tag{3.10}$$

Quando utilizada como função de ativação seu efeito é o de limitar a saída do neurônio para valores entre -1 e 1. *Logits* muito negativos tornam-se -1 e *logits* muito positivos tornam-se 1. A vantagem de utilizar está função de ativação é que os valores da saída do neurônio tornam-se mais próximos de uma distribuição normal centrada no zero. Devido a esta particularidade a saída do neurônio tende a ser mais estável. Na prática a função tangente hiperbólica é preferível à função sigmoide.

Figura 16 – Curva característica da função tangente hiperbólica no intervalo -10 a 10



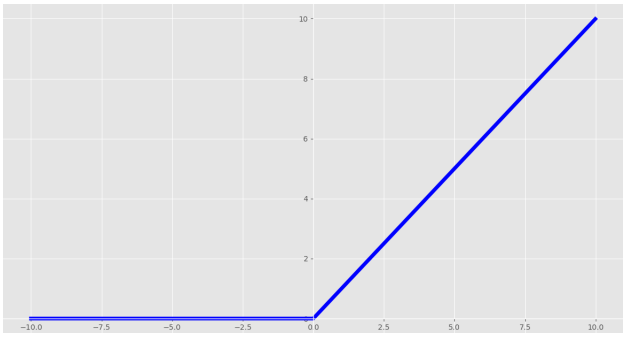
A função ReLU é definida como

$$\rho(z) = max(0, z) \quad . \tag{3.11}$$

ReLU é a sigla para *Rectified Linear Unit* ou Unidade Linear Retificada, proposta por Nair e Hinton (2010) tornou-se, nos últimos anos, a função de ativação mais utilizada na construção de redes neurais e redes neurais convolucionais. Dado um *logit* z, a função computa $max(0, z)$, ou seja, é realizado um *threshold* em 0 com o *logit*. Na literatura há registro de que a função de ativação ReLU apresentou um desempenho 6 vezes maior na velocidade de aprendizagem

quando comparada com as funções sigmoide e tangente hiperbólica (KRIZHEVSKY et al., 2017). Outro ponto positivo desta função é o seu baixo custo computacional visto que não envolve a computação de exponenciais em seu cálculo.

Figura 17 – Curva característica da função ReLU no intervalo -10 a 10



Outra função de ativação utilizada com redes neurais é a *Softmax*, entretanto ela só é utilizada na camada de saída. Sua principal propriedade é produzir uma distribuição de probabilidades na saída da rede neural com base nos *logits*.

Seja $z_i = (z_1, z_2, \cdots, z_k) \in \mathbb{R}^k$ com $i = \{1, 2, \cdots, k\}$ o vetor com os *logits* de uma camada de saída com k neurônios. A função *Softmax* recebe como entrada o vetor z_i e produz como saída uma distribuição de k probabilidades discretas. Sua equação matemática é

$$\rho(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad . \tag{3.12}$$

A função *Softmax* é utilizada em conjunto com a função de perda (*loss function*) da entropia cruzada (*cross entropy*). A entropia cruzada de duas distribuições de probabilidade discretas p e q é dada pela equação

$$H(p, q) = -\sum_{i} p_i \log(q_i) \quad . \tag{3.13}$$

O cálculo da entropia cruzada permite mensurar a diferença entre a distribuição de probabilidade da saída da rede neural q, gerada pela *Softmax*, e a distribuição de probabilidade esperada p (GOODFELLOW et al., 2016).

Por exemplo, seja x um elemento de um domínio e que pode pertencer a 4 classes distintas $Y = \{C_1, C_2, C_3, C_4\}$ e seja $z = (-0,5; 1,8; 0,2; 1,5)$ os *logits* da última camada de uma rede neural que classifica x em uma dentre as 4 classes possíveis. A função de ativação *Softmax* transforma z na distribuição de probabilidades q da seguinte forma:

$$\sum_{j=1}^{k} e^{z_j} = e^{-0,5} + e^{1,8} + e^{0,2} + e^{-1,5} \simeq 8,1$$

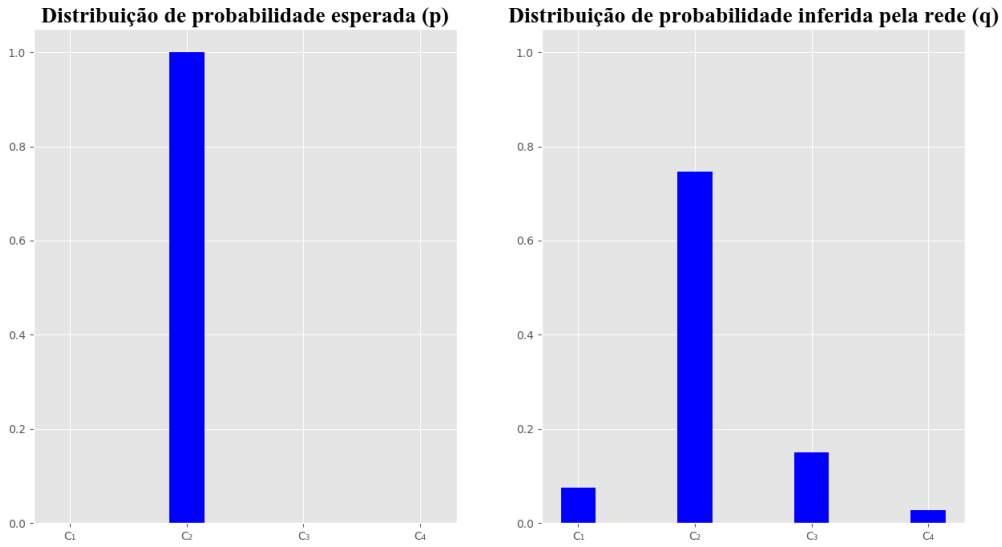
$$q_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \simeq (\frac{e^{-0,5}}{8,1}; \frac{e^{1,8}}{8,1}; \frac{e^{0,2}}{8,1}; \frac{e^{1,5}}{8,1})$$

$$q_i \simeq (0,075; 0,747; 0,150; 0,028)$$

$$\sum_{i} q_i = 1$$

No caso do elemento x pertencer a classe C_2, a distribuição de probabilidade esperada como resultado seria $p_i = (0, 1, 0, 0)$, ou seja, num cenário ideal a rede neural deveria prever com 100% de certeza que x é da classe C_2. As duas distribuições podem ser vistas na Figura 18.

Figura 18 – Distribuição de probabilidade esperada *versus* a inferida



A semelhança da distribuição obtida "q" com relação a distribuição esperada "p" pode ser mensurada com o cálculo da entropia cruzada da seguinte forma

$$H(p, q) = -\sum_{i} p_i \log(q_i) = -(0 \cdot \log(0,075) + 1 \cdot \log(0,747) + 0 \cdot \log(0,150) + 0 \cdot \log(0,028))$$

$$H(p, q) \simeq 0,292$$

Quanto menor for a entropia cruzada, mais semelhante serão as duas distribuições e por consequência mais acurado se torna o modelo (BUDUMA, 2017; STANFORD UNIVERSITY, 2017; GOODFELLOW et al., 2016).

3.2 Redes Neurais Convolucionais

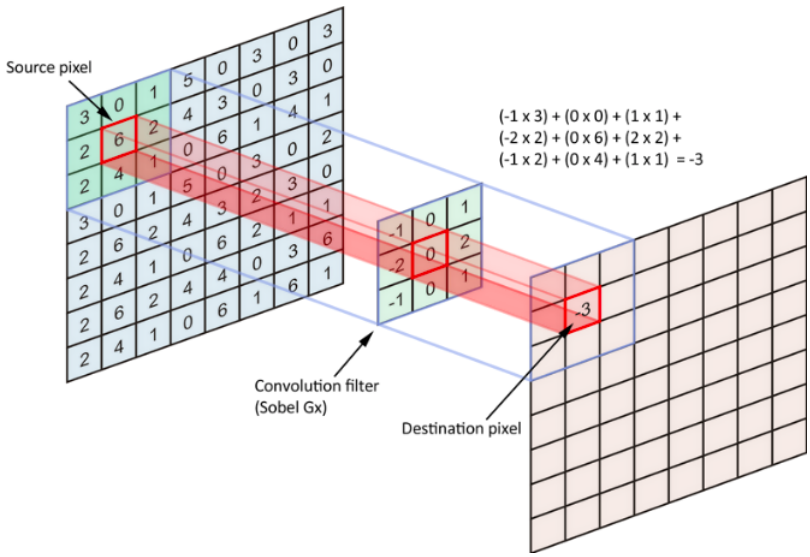
Em processamento de imagens um *kernel* ou matriz de convolução é uma matriz, usualmente quadrada, que é utilizada para realizar transformações como desfoque, ajuste de

nitidez, detecção de bordas, remoção de ruídos e suavização em imagens por meio da operação matemática da convolução. Cada *pixel* resultante da convolução da imagem com o *kernel* é calculado colocando-se o *kernel* no ponto de interesse e, em seguida os valores dos *pixels* adjacentes são multiplicados pelos coeficientes do *kernel*. Os valores destas multiplicações são então somados para dar origem ao novo *pixel*. Caso seja necessário o *kernel* é movido verticalmente e horizontalmente pela imagem (NIXON, 2008). De maneira formal a convolução de um *kernel* K com uma imagem bidimensional I é dada pela equação

$$S(i,j) = K * I = \sum_m \sum_n K(m,n)I(i-m, j-n) \quad . \tag{3.14}$$

Um exemplo de cálculo de um *pixel* pelo processo de convolução está ilustrado na Figura 19.

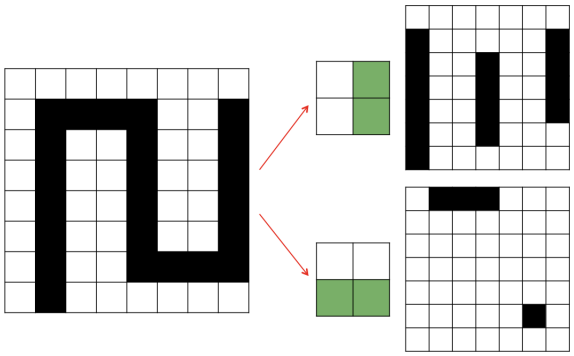
Figura 19 – Processo de Convolução de uma imagem com um *kernel*



Fonte: freeCodeCamp (2018)

As Redes Neurais Convolucionais, ou simplesmente Redes Convolucionais, compartilham similaridades com as redes neurais apresentadas na subseção 3.1.4, entretanto a principal diferença é que o seu desenvolvimento tem como premissa que suas camadas realizarão operações de convolução. Na literatura de redes convolucionais o *kernel* recebe o nome de filtro, e o resultado da convolução de uma imagem com um filtro recebe o nome de mapa de características (*feature map*). Dentro da rede convolucional o filtro cumpre o papel de um extrator ou detector de características das imagens (BUDUMA, 2017; GOODFELLOW et al., 2016).

Figura 20 – Exemplos de filtros capazes de identificar linhas verticais e horizontais de uma imagem



Fonte: Buduma (2017, p. 92)

As redes convolucionais são compostas por 3 tipos fundamentais de camadas: as *camadas convolucionais*, as *camadas de pooling* também chamadas de *camadas de sub-amostragem* ou *camadas agregação/agrupamento* e, por fim, a *camada totalmente conectada* (STANFORD UNIVERSITY, 2017).

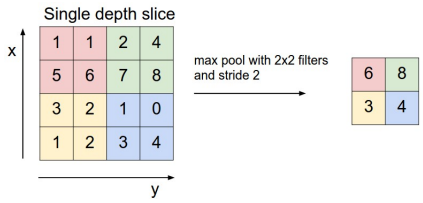
A camada convolucional é o principal componente das Redes Convolucionais, responsável pelo maior custo computacional dentre todos os outros tipos de camada. É composta por um conjunto de filtros cujos coeficientes serão aprendidos no processo de treinamento. Os filtros de uma camada convolucional são volumétricos, ou seja, possuem 3 dimensões: altura, largura e profundidade ($K \in \mathbb{R}^{m,n,d}$). O objetivo dos filtros é extrair os melhores descritores das imagens de entrada para utilizá-los na tarefa na classificação. Por exemplo, a primeira camada convolucional que recebe como entrada imagem bruta pode aprender filtros capazes de extrair bordas, segmentar regiões ou identificar cores na imagem.

A operação de convolução é guiada por dois parâmetros: o passo (*stride*) e o preenchimento (*padding*). O passo define como os filtros devem percorrer a imagem durante a convolução, quando o passo é igual a 1 o filtro se move 1 *pixel* de cada vez, quando igual a 2, a convolução é feita a cada 2 *pixels*. O preenchimento define como as bordas serão tratadas na convolução, duas opções são comuns: a *same* que copia os *pixels* da borda para poder realizar a convolução com todos os *pixels* da imagem, e a *valid* que realiza a convolução somente com os *pixels* que caibam dentro do filtro, sem realizar preenchimento nas bordas.

A camada de agrupamento é inserida entre as camadas convolucionais para reduzir a dimensão espacial dos mapas de características e, por consequência, o custo computacional da rede. A operação de agrupamento da rede também serve para controlar o *overfitting*. *Overfitting* é quando o modelo se especializa na classificação dos elementos do conjunto de treinamento e perde a generalidade na classificação dos outros elementos do domínio. Há duas formas de realizar o agrupamento: com uma operação de *max* (*Max Pooling*) ou com uma operação de média (*Average Pooling*). Comumente é utilizado um filtro de tamanho 2x2 com um passo de

2, desta forma cada operação computa o máximo ou a média entre 4 números para gerar 1 resultado, ou seja, descartando 75% dos coeficientes de uma mapa de características. O processo de agrupamento pode ser visto na Figura 21.

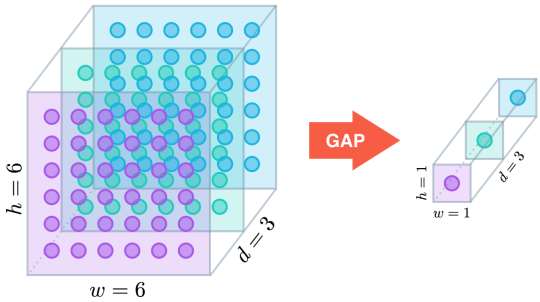
Figura 21 – Exemplo de operação de agrupamento com *Max Polling*



Fonte: Stanford University (2017)

A camada totalmente conectada é uma rede neural composta por uma camada de entrada, camadas ocultas e uma camada de saída com a mesma estrutura que é apresentada na subseção 3.1.4. A conexão entre a última camada convolucional com a camada de totalmente conectada pode ser feita de várias formas diferentes, entretanto as duas são mais comuns são: *Global Max Pooling*, em que é realizada uma operação de *max* com todos os elementos do mapa de características, desta forma cada mapa resulta em 1 elemento do vetor de entrada; e *Global Average Pooling*, no qual a média de todos os elementos do mapa de características dará origem a 1 elemento do vetor de entrada. Uma ilustração da operação *Global Average Polling* pode ser visto na Figura 22.

Figura 22 – Ilustração da operação de *Global Average Polling*



Fonte: Alexis Cook (2017)

As arquiteturas mais comuns de redes convolucionais são compostas por pilhas de camadas convolucionais que utilizam ReLU como função de ativação, seguidas de camadas de agrupamento, por fim é adicionada uma camada totalmente conectada cuja saída utiliza a função *Softmax* (STANFORD UNIVERSITY, 2017). Existem outros tipos de camadas que realizam outras funções dentro da rede como a camada de *Dropout* (HINTON et al., 2012) e a camada de *Batch Normalization* (IOFFE; SZEGEDY, 2015). A camada de *Dropout* é utilizada para desativar aleatoriamente uma parte das conexões entre as camadas durante o processo de treinamento,

de acordo com o trabalho de Srivastava et al. (2014) está técnica reduz a chance de *overfitting* e melhora a generalização da rede neural. A camada de *Batch Normalization* é utilizada para normalizar (média igual a 0 e desvio padrão igual a 1) os valores que são passados entre as camadas dentro da rede, segundo Ioffe e Szegedy (2015) a utilização deste tipo de camada aumenta a velocidade de aprendizagem e estabilidade da rede.

As arquiteturas de redes convolucionais variam entre si no número de camadas convolucionais e nos parâmetros da convolução como o valor de passo nas camadas de agrupamento, nas funções de ativação utilizadas, na quantidade e tamanho dos filtros entre outros aspectos. Uma das referências em arquiteturas de redes convolucionais é o desafio *Imagenet* ou *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC) cujo o objetivo é avaliar metodologias para detecção e classificação de imagens e permitir aos pesquisadores compartilhar conhecimentos e acompanhar o progresso na área de detecção e classificação de imagens. O desafio consiste em identificar a classe de mais de 14 milhões de imagens entre mais de 20 mil classes distintas (RUSSAKOVSKY et al., 2015).

Das arquiteturas propostas do desafio Imagenet, 5 são utilizadas neste trabalho: a VGG16 e VGG19 apresentadas na subseção 3.2.1, a *InceptionV3* apresentada na subseção 3.2.2 a *Xception* apresentada na subseção 3.2.3 e a *ResNet50* apresentada na subseção 3.2.4.

3.2.1 VGG16 e VGG19

A arquitetura vice-campeã no ILSVRC do ano de 2014 foi a VGGNet, proposta pelo VGG (*Visual Geometry Group*) da Universidade de *Oxford* (SIMONYAN; ZISSERMAN, 2014). Em seu artigo foram apresentadas 6 configurações de redes convolucionais, a VGG11, VGG11 (LRN), VGG13, VGG16 (conv1), a VGG16 e VGG19. Cada uma destas redes tem a quantidade de camadas igual ao número associado ao seu nome. A premissa desta arquitetura é que a profundidade da rede neural é um parâmetro importante na acurácia da classificação.

As camadas convolucionais usam ReLU como função de ativação com exceção da LRN (*Local Response Normalization*), os filtros de convolução tem tamanho 3x3 com exceção da camada "conv1" que utiliza filtros de tamanho 1x1. As convoluções acontecem com passo igual a 1 e preenchimento do tipo *same*. As camadas de agrupamento usam uma janela de 2x2 com passo igual a 2. Um resumo das configurações da arquitetura VGGNet pode ser visto no Quadro 1.

Quadro 1 – Variações da arquitetura VGGNet. A profundidade da rede aumenta de A para E, cada nova camada adicionada está marcada em negrito. O nome das camadas segue o seguinte padrão: conv<tamanho dos filtros utilizados na camada>-<quantidade de filtros utilizados na camada>

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

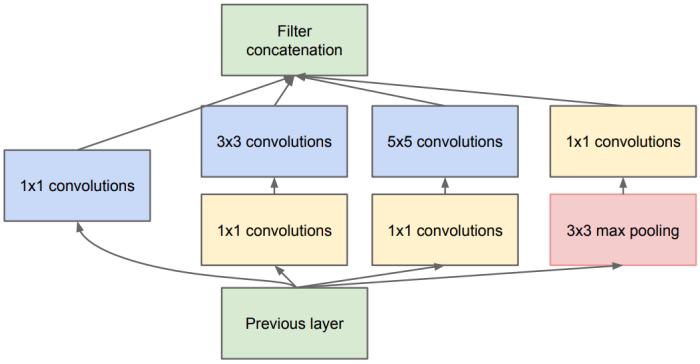
Fonte: Simonyan e Zisserman (2014)

Das configurações presentes no Quadro 1 a D (VGG-16) e E (VGG-19) foram as que obtiveram as maiores acurácias no ILSVRC.

3.2.2 InceptionV3

A arquitetura campeã no ILSVRC no ano de 2014 foi a *GoogleNet / Inception* proposta por Szegedy et al. (2014). A arquitetura baseia-se no uso do módulo *inception* e na premissa de que o tamanho dos filtros de convolução é um fator que afeta o desempenho da rede. Um filtro grande é preferível para extrair características distribuídas globalmente nas imagens, por outro lado um filtro pequeno é mais eficaz para as características distribuídas localmente. O módulo *inception* busca atender a estes dois requisitos realizando convoluções com 3 filtros de tamanhos 1x1, 3x3 e 5x5 e ao final concatenar os resultados destas convoluções. Uma representação do módulo *inception* pode ser vista na Figura 23.

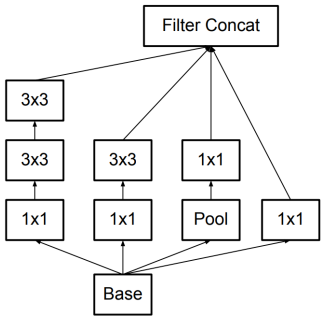
Figura 23 – Módulo *inception*



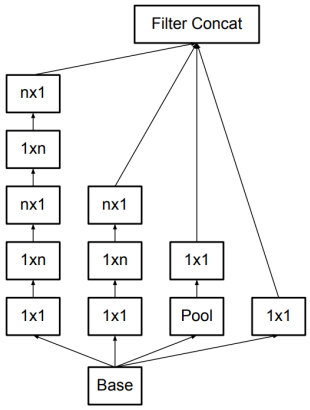
Fonte: Szegedy et al. (2014)

Atualmente existem mais 3 versões além da *Inception*, a *InceptionV2*, *InceptionV3* e *InceptionV4*. Cada uma destas versões apresenta modificações nos módulos *inceptions*. A *InceptionV3* (utilizada neste trabalho) apresenta 3 módulos *inception* diferentes, o módulo *inception A* em que as convoluções 5x5 são substituídas por 2 convoluções 3x3, o módulo *inception B* que substitui as convoluções nxn do *inception A* por duas convoluções 1xn e nx1 e por fim o módulo *inception C* que tem sua própria construção que é diferente dos demais.

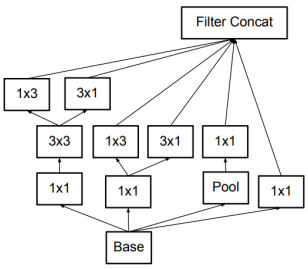
Figura 24 – Módulos *inception* da *InceptionV3*



(a) Módulo *inception* A



(b) Módulo *inception* B



(c) Módulo *inception* C

Fonte: Szegedy et al. (2016)

Todas as camadas convolucionais utilizam ReLU e entre cada camada foi adicionada uma camada de *Batch Normalization*. A arquitetura da *InceptionV3* está colocada no Quadro 2.

Quadro 2 – Camadas da Arquitetura *InceptionV3*

Tipo da camada	**Tamanho dos filtros / passo** (ou comentário)	**Formato da entrada**
Convolucional	3x3/2	299x299x3
Convolucional	3x3/1	149x149x32
Convolucional	3x3/1	147x147x32
Agrupamento	3x3/2	147x147x64
Convolucional	3x3/1	73x73x64
Convolucional	3x3/2	71x71x80
Convolucional	3x3/1	35x35x192
3 × Inception	Como na Figura 24a	35x35x288
5 × Inception	Como na Figura 24b	17x17x768
2 × Inception	Como na Figura 24c	8x8x1280
Agrupamento	8x8	8x8x2048
Totalmente conectada	logits	1x1x2048
Softmax	Classificação	1x1x1000

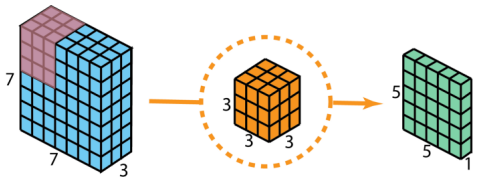
Fonte: Szegedy et al. (2016)

3.2.3 Xception

A arquitetura vice-campeã do ILSVRC no ano de 2015 foi a Xception, que é a abreviação para *Extreme version of Inception* proposta por Chollet (2017) que estende o conceito de realizar várias convoluções com filtros diferentes do módulo *inception*, apresentada na *InceptionV3*, com a utilização do conceito de convoluções separáveis em profundidade (*Depthwise Separable Convolutions*).

Por exemplo, a convolução de uma imagem 7x7x3 com um filtro de tamanho 3x3x3 produz um mapa de características de tamanho 5x5x1 (sem preenchimento das bordas e passo igual a 1) como pode ser visto na Figura 25.

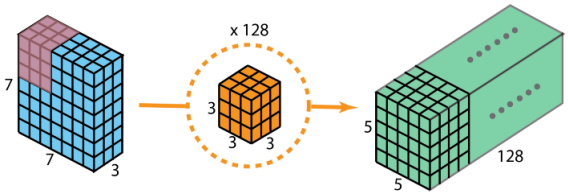
Figura 25 – Convolução de uma imagem 7x7x3 com um filtro 3x3x3



Fonte: Kunlun Bai (2011)

Caso a camada convolucional tenha 128 filtros, o resultado final produzido pela camada será um mapa de características com tamanho 5x5x128 como pode ser visto na Figura 26.

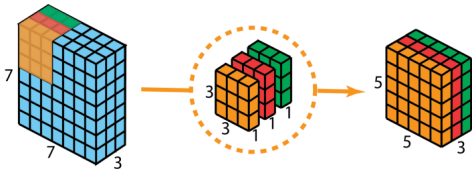
Figura 26 – Convolução de uma imagem 7x7x3 com 128 filtros 3x3x3



Fonte: Kunlun Bai (2011)

Na convolução separável em profundidade o cálculo da saída é feito em duas partes. A primeira é a convolução em profundidade (*depthwise convolution*) que, ao invés de utilizar 1 filtro 3x3x3 são utilizados 3 filtros 3x3x1. Cada um destes 3 filtros será convolvido com 1 canal da imagem, produzindo 3 mapas de características com tamanho 5x5x3 como visto na Figura 27.

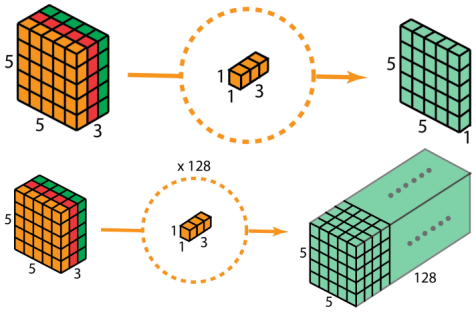
Figura 27 – Convolução em profundidade



Fonte: Kunlun Bai (2011)

A segunda parte da convolução separável em profundidade é a convolução ponto a ponto (*Pointwise Convolution*) em que 128 filtros de tamanho 1x1x3 são convolvidos com o mapa de características 5x5x3 como visto na Figura 28.

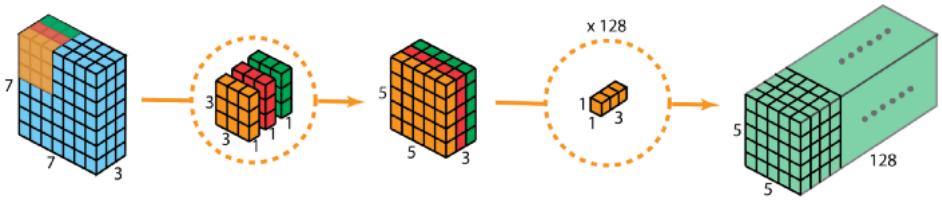
Figura 28 – Convolução ponto a ponto



Fonte: Kunlun Bai (2011)

O procedimento completo da convolução separável em profundidade está ilustrado na Figura 29.

Figura 29 – Convolução Separável em Profundidade

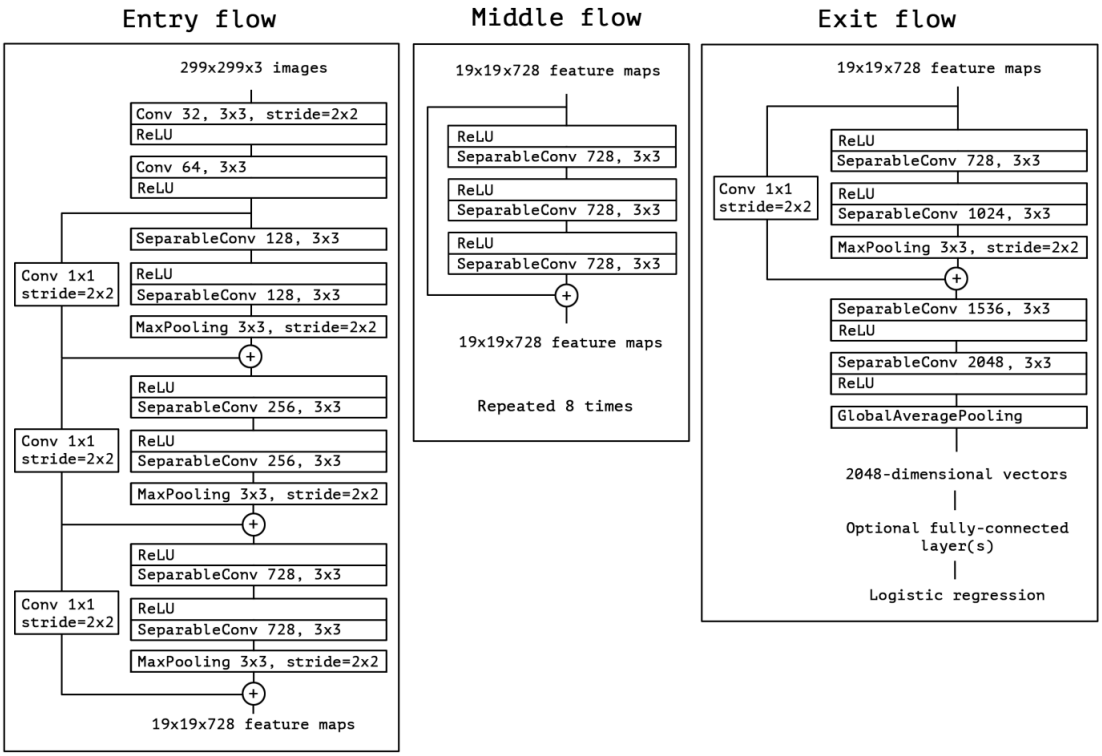


Fonte: Kunlun Bai (2011)

Tanto a convolução original como a convolução separável em profundidade irão produzir o mesmo volume de saída, no caso do exemplo mostrado é o volume de 5x5x128.

A arquitetura Xception é composta por 36 camadas convolucionais que realizam a convolução separável em profundidade, estruturadas em 14 módulos. Os módulos possuem conexões residuais (conexões entre camadas) com exceção do primeiro e último módulo. O resumo desta arquitetura pode ser visto na Figura 30.

Figura 30 – Arquitetura Xception. Todas as convoluções são separáveis em profundidade e após cada convolução há uma camada de *batch normalization*



Fonte: Chollet (2017)

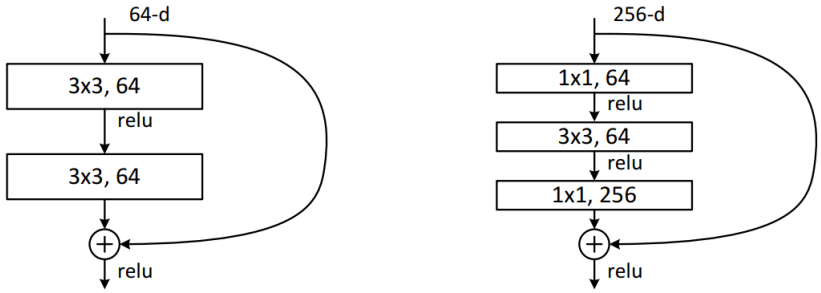
A Xception conseguiu resultados melhores do que a *InceptionV3* no conjunto de imagens do ILSVRC utilizando menos parâmetros (23, 626, 728 da *InceptionV3* contra 22, 855, 952 da Xception) e com uma maior velocidade de aprendizagem (CHOLLET, 2017).

3.2.4 ResNet50

As maior parte das arquiteturas de redes convolucionais são constituídas de pilhas de camadas convolucionais em que cada camada recebe como entrada os dados da camada anterior, realiza sua computação e passa os resultados para a próxima camada. Entretanto a proposta de arquitetura feita por He et al. (2016) utiliza "conexões de atalho" (*skip connections* ou *short-cuts*) entre as camadas convolucionais.

O projeto da ResNet foi inspirado na VGGNet em que as camadas de convolução utilizam filtros de tamanho 3x3, entretanto são utilizados menos filtros. A função de ativação das camadas convolucionais é a ReLu e são adicionadas camadas de *batch normalization* entre elas. As conexões de atalho entre as camadas para a passagem dos resíduos seguem o modelo mostrado na Figura 31.

Figura 31 – Exemplo de como é feita a passagem de resíduo entre as camadas convolucionais da ResNet



Fonte: He et al. (2016)

Em seu trabalho He et al. (2016) apresentou 5 configurações para ResNet: a Resnet-18, ResNet-34, ResNet-50, ResNet-101 e ResNet-152. O número que acompanha o nome é referente a quantidade de camadas convolucionais da rede. Destas configurações somente a ResNet-50, ResNet-101 e ResNet-152 foram avaliadas com o conjunto de imagens do ILSVRC e obtiveram acurácias (*top-5*) de 0, 921, 0, 928 e 0, 931 respectivamente. A quantidade de parâmetros treinamento de cada rede aumenta de 25.636.712 na ResNet-50 para 44.707.176 na ResNet-101 e chega até 60.419.944 na ResNet-152. O aumento na quantidade de parâmetros de treinamento resultou num aumento da acurácia alcançada pela arquitetura. A versão que venceu o ILSVRC em 2015 foi a ResNet-152.

3.3 Análise de Componentes Principais

A redução de dimensionalidade é o processo pelo qual dados de um espaço com muitas dimensões são mapeados para um espaço com poucas dimensões. Este processo é relacionado com o conceito de compressão, estudado em teoria da informação. Dados que são representados com muitas dimensões apresentam desafios computacionais relevantes em termos de armazenamento e

tempo de processamento, por isto, sempre que possível, é desejável a utilização de algum método de compressão. Outra desvantagem de dados com muitas dimensões é a perda de generalização dos modelos quando utilizados em algoritmos de aprendizagem de máquina Shalev-Shwartz e Ben-David (2014).

De maneira geral os métodos de redução de dimensionalidade lineares têm como objetivo transformar um elemento pertencente ao $\mathbb{R}^d$ para o $\mathbb{R}^n$ com $n < d$ por meio da multiplicação por uma matriz $W \in \mathbb{R}^{n,d}$.

Neste trabalho é utilizado o método de redução de dimensionalidade chamado Análise de Componentes Principais (Principal Component Analysis) PCA. Neste método a compressão e recuperação da informação é feita com uma transformação linear em que a diferença quadrática média entre o elemento recuperado da transformação e o elemento original é mínima. O problema matemático que o PCA resolve é, para $x \in \mathbb{R}^d$ encontrar uma matriz de compressão W e a matriz de descompressão U tais que

$$\text{argmin}_{W \in \mathbb{R}^{n,d}, U \in \mathbb{R}^{d,n}} ||x - UWx||_2^2 \quad . \tag{3.15}$$

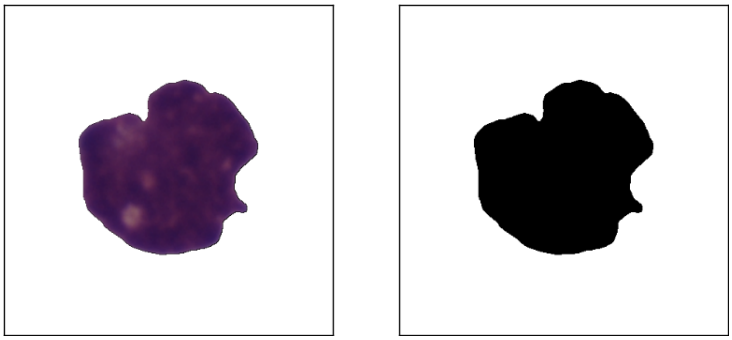
3.4 *Image Biomarker Standardisation Initiative* (IBSI)

A *Image Biomarker Standardisation Initiative* (IBSI) é uma colaboração internacional entre pesquisadores de Imagiologia Médica que tem como objetivo padronizar os descritores (características) extraídos de imagens médicas. A padronização é feita com publicação de um manual de referência que contém as definições matemáticas e nomenclaturas das principais características (ZWANENBURG et al., 2016). O padrão IBSI foi utilizado em dois conjuntos de características: as estatísticas de 1ª ordem (*Intensity Histogram Features*) e com as matrizes de co-ocorrência GLCM, GLSZM, GLRLM, NGTDM e GLDM.

O cálculo destas características foi feito com o auxílio da biblioteca de código aberto *pyradiomics*, especializada na extração de características radiômicas de imagens médicas seguindo padronização da IBSI (GRIETHUYSEN et al., 2017).

Somente os *pixels* que fazem parte da região de interesse da imagem (ROI) são utilizados nos cálculos, sendo ignorado aqueles pertencentes ao plano de fundo. Neste trabalho a ROI é definida pelos *pixels* que compõem o linfócito, e são segmentados do plano de fundo por uma máscara binária criada com operações de *thresholding* a partir das imagens dos linfócitos. Um exemplo de linfócito e sua respectiva máscara pode ser visto na Figura 32.

Figura 32 – Imagem de um linfócito e sua respectiva máscara que segmenta a sua ROI



Na subseção 3.4.1 serão apresentados os cálculo das estatísticas de 1ª ordem e na subseção 3.4.2 serão apresentados os cálculos das características de textura extraídas das matrizes de co-ocorrência de acordo com a documentação da biblioteca *pyradiomics*.

3.4.1 Estatísticas de 1ª Ordem

As estatísticas de 1ª ordem ou *Intensity Histogram Features* (ZWANENBURG et al., 2016) são computadas a partir dos *bins* dos histogramas das imagens. Seja n_p igual ao número de *pixels* da ROI e $X_i = (X_1, X_2, \cdots, X_{n_p})$ o conjunto que contém estes *pixels*. Seja $P_i = (P_1, P_2, \cdots, P_{n_g})$ o histograma de X_i com a contagem de frequência dos n_g níveis discretos de intensidade da imagem. Para imagens em escala de cinza n_g é igual a 255, com isto o histograma será da forma $P_i = (P_1, P_2, \cdots, P_{255})$. Deste modo a probabilidade de ocorrência de cada nível discreto de intensidade é definida como $p_i = \dfrac{P_i}{n_p}$. O cálculo das estatísticas de 1ª ordem segue as seguintes formulações:

Média Média aritmética de todos os *pixels* da ROI

$$\overline{X} = \frac{1}{n_p} \sum_{i=1}^{n_p} X_i \tag{3.16}$$

Desvio Padrão O desvio que os *pixels* da ROI tem de sua média

$$s = \sqrt{\frac{1}{n_p} \sum_{i=1}^{n_p} \left(X_i - \overline{X}\right)^2} \tag{3.17}$$

Variância A variância dos *pixels* da ROI.

$$s^2 = \frac{1}{n_p} \sum_{i=1}^{n_p} \left(X_i - \overline{X}\right)^2 \tag{3.18}$$

Valor Mínimo O valor do *Pixel* com menor intensidade dentro da ROI

$$\textit{valor mínimo} = \min(X_i) \tag{3.19}$$

Valor Máximo O valor do *Pixel* com maior intensidade dentro da ROI

$$\textit{valor máximo} = \max(X_i) \tag{3.20}$$

Intervalo de variação Diferença entre o valor mínimo e o valor máximo

$$\textit{intervalo de variação} = \max(X_i) - \min(X_i) \tag{3.21}$$

Valor Quadrático Médio É a raiz quadrada da média do quadrado de todos os *pixels* da ROI

$$\text{RMS} = \sqrt{\frac{1}{n_p}\sum_{i=1}^{n_p} X_i^2} \tag{3.22}$$

Desvio padrão absoluto da média É a média das diferenças entre cada *pixel* e a sua média

$$\text{MAD} = \frac{1}{n_p}\sum_{i=1}^{n_p} |X_i - \overline{X}| \tag{3.23}$$

Desvio padrão absoluto da média robusta É a média das diferenças entre cada *pixel* e sua média no subconjunto de *pixels* com níveis de cinza entre o 10º percentil e 90º percentil

$$\text{rMAD} = \frac{1}{N_{10,90}}\sum_{i=0}^{N_{10,90}} |X_{10,90_i} - \overline{X_{10,90}}| \tag{3.24}$$

Uniformidade É definida pelo somatório do quadrado das probabilidades de ocorrência de cada nível discreto de intensidade. Para histogramas em que as maior parte das intensidades está em um único *bin*, a Uniformidade se aproxima de 1

$$\text{Uniformidade} = \sum_{i=1}^{n_g} {p_i}^2 \tag{3.25}$$

Energia É o somatório do quadrado dos *pixels* na ROI

$$\text{Energia} = \sum_{i=1}^{n_p} {X_i}^2 \tag{3.26}$$

Entropia É o conceito da teoria da informação que mede a quantidade de informação que está contida na ROI. É utilizada a definição de entropia de Shannon

$$\text{Entropia} = -\sum_{i=1}^{n_g} p_i \log_2(p_i) \tag{3.27}$$

Curtose É a medida do achatamento do histograma: quanto maior seu valor, mais alongada será a cauda do histograma Sejam μ_4 e σ respectivamente o quarto momento central e o desvio padrão do histograma, a Curtose será definida como

$$\text{Curtose} = \frac{\mu_4}{\sigma^4} = \frac{\frac{1}{n_p}\sum_{i=1}^{n_p}\left(X_i - \overline{X}\right)^4}{\left(\frac{1}{n_p}\sum_{i=1}^{n_p}\left(X_i - \overline{X}\right)^2\right)^2} \tag{3.28}$$

Assimetria É a medida da falta de simetria do histograma com relação a média. Se a assimetria de uma distribuição é maior do que 0 sua cauda direita é mais pesada, se é menor do que 0 sua cauda esquerda é mais pesada e se é igual a 0 a distribuição é simétrica com relação a média. Sejam μ_3 e σ respectivamente o terceiro momento central e o desvio padrão do histograma, a Assimetria será definida como

$$\text{Assimetria} = \frac{\mu_3}{\sigma^3} = \frac{\frac{1}{n_p}\sum_{i=1}^{n_p}\left(X_i - \overline{X})^3\right)}{\left(\sqrt{\frac{1}{n_p}\sum_{i=1}^{n_p}\left(X_i - \overline{X}\right)^2}\right)^3} \tag{3.29}$$

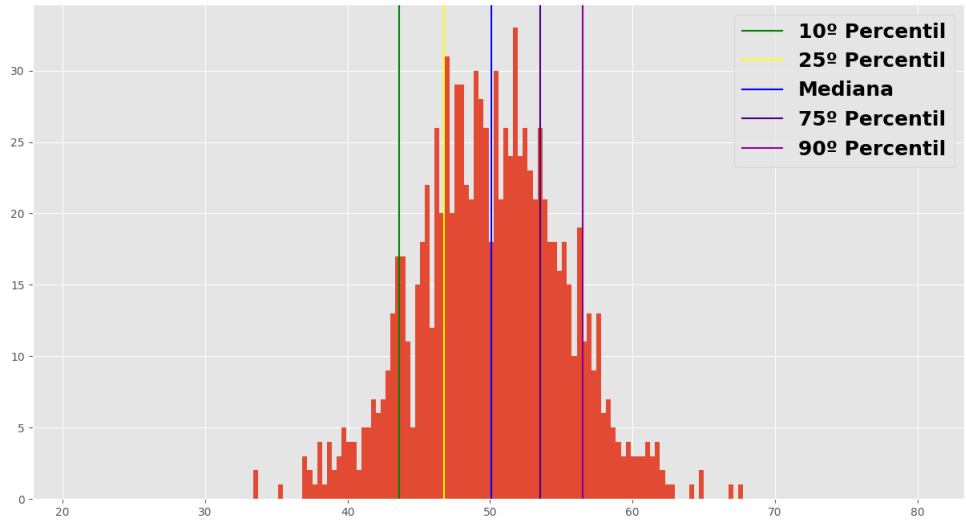
Mediana Valor que separa a metade menor da metade maior dos *pixels* da ROI. Ver Figura 33

10º percentil Valor que separa 10% dos menores valores de *pixels* dos 90% maiores. Ver Figura 33

90º percentil Valor que separa 90% dos menores valores de *pixels* dos 10% maiores. Ver Figura 33

Variação interquartil E a diferença entre o 25º percentil e o 75º percentil

Figura 33 – Exemplo de Mediana, 10º, 90º, 25º e 75º percentis



3.4.2 Características de Textura

A textura oferece a impressão visual de rugosidade ou suavidade de uma superfície, e contém as informações sobre a distribuição espacial das variações de tonalidade de uma imagem. Uma das formas de definir a textura é como as variações locais em valores de *pixels* que se repetem, de maneira regular ou aleatória, ao longo de uma imagem (NASCIMENTO et al., 2003).

As matrizes de Co-ocorrência são tabulações das quantidades de combinações diferentes de valores de intensidade (níveis de cinza) que ocorrem nos *pixels* em uma imagem. A ideia principal destas matrizes é representar as texturas de uma imagem através de um conjunto de estatísticas que são calculadas com base nas variações dos *pixels* da imagem. A IBSI padroniza 5 matrizes de co-ocorrência: a GLCM apresentada na seção 3.4.2.1, a GLSZM apresentada na seção 3.4.2.2, a GLRLM apresentada na seção 3.4.2.3, a NGTDM apresentada na seção 3.4.2.4 e por fim a GLDM apresentada na seção 3.4.2.5.

3.4.2.1 GLCM

GLCM é a sigla para *Gray Level Co-occurrence Matrix* ou Matriz de Co-ocorrência de Níveis de Cinza. Seja uma imagem com n_g níveis discretos de intensidade, a GLCM é uma matriz de tamanho $n_g \times n_g$ que descreve a probabilidade conjunta $p(i, j|\sigma, \theta)$ das ocorrências de intensidade dos *pixels* da imagens. Cada elemento (i, j) da matriz representa a quantidade de combinações de intensidades dos níveis i e j que ocorreram em 2 *pixels* da imagem separados pela distância σ no ângulo θ (PYRADIOMICS, 2019). Por exemplo, seja a imagem 5x5 com 5 níveis discretos de intensidade em 3.30, para $\sigma = 1$ (*pixels* com distância 1) e $\theta = 0$ (plano

horizontal, somente *pixels* à direita e à esquerda) a matriz GLCM computada é aquela em 3.31.

$$\mathbf{I} = \begin{bmatrix} 1 & 2 & 5 & 2 & 3 \\ 3 & 2 & 1 & 3 & 1 \\ 1 & 3 & 5 & 5 & 2 \\ 1 & 1 & 1 & 1 & 2 \\ 1 & 2 & 4 & 3 & 5 \end{bmatrix} \tag{3.30}$$

$$\mathbf{p} = \begin{bmatrix} 6 & 4 & 3 & 0 & 0 \\ 4 & 0 & 2 & 1 & 3 \\ 3 & 2 & 0 & 1 & 2 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 3 & 2 & 0 & 2 \end{bmatrix} \tag{3.31}$$

Por exemplo $p(1,2) = 4$, logo houve 4 ocorrências dos níveis de intensidades 1 e 2 separadas a uma distância de 1 *pixel*, as 4 ocorrências foram:

$$\big\{\big(I(1,1),\ I(1,2)\big);\ \big(I(2,3),\ I(2,2)\big);\ \big(I(4,4),\ I(4,5)\big);\ \big(I(5,1),\ I(5,2)\big)\big\}\ .$$

Sejam:

- $P(i,j)$ a matriz GLCM
- $p(i,j) = \dfrac{P(i,j)}{\sum P(i,j)}$
- n_g a quantidade de níveis discretos de intensidade da imagem
- $p_x(i) = \sum_{j=1}^{n_g} P(i,j)$ a probabilidade marginal das linhas
- $p_y(j) = \sum_{i=1}^{n_g} P(i,j)$ a probabilidade marginal das colunas
- $\mu_x = \sum_{i=1}^{n_g} p_x(i)i$
- $\mu_y = \sum_{j=1}^{n_g} p_y(j)j$
- σ_x o desvio padrão de $p_x(i)$
- σ_y o desvio padrão de $p_y(i)$

- $p_{x+y}(k) = \sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j),$ em que $i + j = k,$ e $k = 2, 3, \ldots, 2n_g$
- $p_{x-y}(k) = \sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j),$ em que $|i - j| = k,$ e $k = 0, 1, \ldots, n_g - 1$
- $\text{HXY2} = -\sum_{i=1}^{N_g} \sum_{j=1}^{n_g} p_x(i) p_y(j) \log_2 \left(p_x(i) p_y(j)\right)$
- $\text{HXY} = -\sum_{i=1}^{N_g} \sum_{j=1}^{n_g} p(i,j) \log_2 \left(p(i,j)\right)$

Os valores calculados a partir de uma matriz GLCM são os seguintes::

Autocorrelação

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j) ij \tag{3.32}$$

Média conjunta

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j) i \tag{3.33}$$

Proeminência dos grupos

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} \left(i + j - \mu_x - \mu_y\right)^4 p(i,j) \tag{3.34}$$

Tonalidade dos grupos

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} \left(i + j - \mu_x - \mu_y\right)^3 p(i,j) \tag{3.35}$$

Tendência dos grupos

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} \left(i + j - \mu_x - \mu_y\right)^2 p(i,j) \tag{3.36}$$

Contraste

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} (i - j)^2 p(i,j) \tag{3.37}$$

Correlação

$$\frac{\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j) ij - \mu_x \mu_y}{\sigma_x(i) \sigma_y(j)} \tag{3.38}$$

Variação da média

$$\sum_{k=0}^{n_g-1} k p_{x-y}(k) \tag{3.39}$$

Variação da entropia

$$\sum_{k=0}^{n_g-1} p_{x-y}(k) \log_2 \left(p_{x-y}(k)\right) \tag{3.40}$$

Variação da variância

$$\sum_{k=0}^{n_g-1} (k - \text{Variação da média})^2 p_{x-y}(k) \tag{3.41}$$

Energia conjunta

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} \left(p(i,j)\right)^2 \tag{3.42}$$

Entropia conjunta

$$-\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j) \log_2 \left(p(i,j)\right) \tag{3.43}$$

Medida Informacional de Correlação 1

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j) \log_2 \left(p(i,j)\right) - \sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j) \log_2 \left(p_x(i) p_y(j)\right) \tag{3.44}$$

Medida Informacional de Correlação 2

$$\sqrt{1 - e^{-2(HXY2 - HXY)}} \tag{3.45}$$

Variação do Momento Inverso

$$\sum_{k=0}^{n_g-1} \frac{p_{x-y}(k)}{1 + k^2} \tag{3.46}$$

Coeficiente máximo de correlação

$$\sum_{k=0}^{n_g} \frac{p(i,k) p(j,k)}{p_x(i) p_y(k)} \tag{3.47}$$

Variação do momento inverso normalizado

$$\sum_{k=0}^{n_g-1} \frac{p_{x-y}(k)}{1+\left(\frac{k^2}{n_g^2}\right)} \tag{3.48}$$

Variação inversa

$$\sum_{k=0}^{n_g-1} \frac{p_{x-y}(k)}{1+k} \tag{3.49}$$

Variação inversa normalizada

$$\sum_{k=0}^{n_g-1} \frac{p_{x-y}(k)}{1+\left(\frac{k}{n_g}\right)} \tag{3.50}$$

Variância inversa

$$\sum_{k=1}^{n_g-1} \frac{p_{x-y}(k)}{k^2} \tag{3.51}$$

Probabilidade máxima

$$\max\left(p(i,j)\right) \tag{3.52}$$

Soma média

$$\sum_{k=2}^{2n_g} p_{x+y}(k)k \tag{3.53}$$

Soma da entropia

$$\sum_{k=2}^{2n_g} p_{x+y}(k)\log_2\left(p_{x+y}(k)\right) \tag{3.54}$$

Soma dos quadrados

$$\sum_{i=1}^{n_g}\sum_{j=1}^{n_g}(i-\mu_x)^2 p(i,j) \tag{3.55}$$

3.4.2.2 GLSZM

GLSZM é a sigla para *Gray Level Size Zone Matrix* ou Matriz de Tamanho das Regiões de Nível de Cinza. A matriz quantifica as regiões na imagem que possuem os mesmos níveis de cinza. Cada elemento (i, j) da matriz é igual à quantidade de regiões com *pixels* de intensidade i e tamanho j (em *pixels*) presentes na imagem (PYRADIOMICS, 2019). Por exemplo, seja

a imagem 5x5 com 5 níveis discretos de intensidade em 3.56, a matriz GLSZM computada é aquela em 3.57.

$$\mathbf{I} = \begin{bmatrix} 5 & 2 & 5 & 4 & 4 \\ 3 & 3 & 3 & 1 & 3 \\ 2 & 1 & 1 & 1 & 3 \\ 4 & 2 & 2 & 2 & 3 \\ 3 & 5 & 3 & 3 & 2 \end{bmatrix} \tag{3.56}$$

$$\mathbf{p} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.57}$$

O elemento $p(3,1) = 1$ significa que dos *pixels* de intensidade igual a 3 há 1 região com tamanho de 1 *pixel* $\big(I(5,1)\big)$.

O elemento $p(3,3) = 1$ significa que dos *pixels* de intensidade igual a 3 há 1 região com tamanho de 3 *pixels* $\big(I(2,1),\ I(2,2),\ I(2,3)\big)$.

O elemento $p(3,5) = 1$ significa que dos *pixels* de intensidade igual a 3 há 1 região com tamanho de 5 *pixels* $\big(I(2,5),\ I(3,5),\ I(4,5),\ I(5,4),\ I(5,3)\big)$.

Sejam:

- $P(i,j)$ a matriz GLSZM
- $p(i,j) = \frac{P(i,j)}{N_z}$
- n_g a quantidade de níveis de intensidade da imagem
- n_s o quantidade de tamanhos possíveis de área
- n_p a quantidade de *pixels* da imagem
- $n_z = \sum_{i=1}^{n_g} \sum_{j=1}^{N_s} P(i,j)$

Os valores calculados a partir de uma matriz GLSZM são os seguintes::

Ênfase em pequenas regiões

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_s}\frac{P(i,j)}{j^2}}{n_z} \tag{3.58}$$

Ênfase em grandes regiões

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_s}P(i,j)j^2}{n_z} \tag{3.59}$$

Não uniformidade dos níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\left(\sum_{j=1}^{n_s}P(i,j)\right)^2}{n_z} \tag{3.60}$$

Não uniformidade dos níveis de cinza normalizado

$$\frac{\sum_{i=1}^{n_g}\left(\sum_{j=1}^{n_s}P(i,j)\right)^2}{n_z^2} \tag{3.61}$$

Regiões não uniformes

$$\frac{\sum_{j=1}^{n_s}\left(\sum_{i=1}^{n_g}P(i,j)\right)^2}{n_z} \tag{3.62}$$

Regiões não uniformes normalizado

$$\frac{\sum_{j=1}^{n_s}\left(\sum_{i=1}^{n_g}P(i,j)\right)^2}{n_z^2} \tag{3.63}$$

Porcentagem das regiões

$$\frac{n_z}{n_p} \tag{3.64}$$

Variância dos níveis de cinza

$$\begin{gathered}\sum_{i=1}^{n_g}\sum_{j=1}^{n_s} p(i,j)(i-\mu)^2 \\ \mu = \sum_{i=1}^{n_g}\sum_{j=1}^{n_s} p(i,j)i\end{gathered} \tag{3.65}$$

Variância das regiões

$$\begin{gathered}\sum_{i=1}^{n_g}\sum_{j=1}^{n_s} p(i,j)(j-\mu)^2 \\ \mu = \sum_{i=1}^{n_g}\sum_{j=1}^{n_s} p(i,j)j\end{gathered} \tag{3.66}$$

Entropia das regiões

$$-\sum_{i=1}^{n_g}\sum_{j=1}^{n_s} p(i,j)\log_2(p(i,j)) \tag{3.67}$$

Ênfase nas regiões com os menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_s}\frac{P(i,j)}{i^2}}{n_z} \tag{3.68}$$

Enfase nas regiões com os maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_s} P(i,j)i^2}{n_z} \tag{3.69}$$

Pequenas regiões com os menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_s}\frac{P(i,j)}{i^2j^2}}{n_z} \tag{3.70}$$

Pequenas regiões com os maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_s}\frac{P(i,j)i^2}{j^2}}{n_z} \tag{3.71}$$

Grandes regiões com os menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_s}\frac{P(i,j)j^2}{i^2}}{n_z} \tag{3.72}$$

Grandes regiões com os maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_s}P(i,j)i^2j^2}{n_z} \tag{3.73}$$

3.4.2.3 GLRLM

GLRLM é a sigla para *Gray Level Run Length Matrix* ou Matriz de Sequência de Níveis de Cinza. A matriz quantifica as as sequências de *pixels* que possuem o mesmo valor de intensidade. Os elementos da matriz GLRLM são da forma $P(i, j|\theta)$ e representam quantas sequências com *pixels* de intensidade i e comprimento j no ângulo θ ocorreram na imagem (KITWARE INC. 2019, 2019). Por exemplo, seja I a imagem 5x5 com 5 níveis discretos de intensidade em 3.74, para $\theta = 0$ (plano horizontal, somente *pixels* à direita e à esquerda) a matriz GLRLM computada é aquela em 3.75.

$$\mathbf{I} = \begin{bmatrix} 5 & 2 & 5 & 4 & 4 \\ 3 & 3 & 3 & 1 & 3 \\ 2 & 1 & 1 & 1 & 3 \\ 4 & 2 & 2 & 2 & 3 \\ 3 & 5 & 3 & 3 & 2 \end{bmatrix} \tag{3.74}$$

$$\mathbf{p} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 & 0 \\ 4 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.75}$$

O elemento $P(1, 1) = 1$ significa que há 1 sequência de tamanho 1 com os *pixels* de intensidade 1 ($I_{2,4}$). O elemento $P_{1,3} = 1$ significa que há 1 sequência de tamanho 3 com os *pixels* de intensidade 1 $\big(I(3, 1),\ I(3, 2),\ I(3, 3)\big)$.

Sejam:

- $P(i, j|\theta)$ a matriz GLRLM
- $p(i, j|\theta) = \frac{P(i,j|\theta)}{n_r}$

- n_g a quantidade de níveis de intensidade da imagem
- n_r a quantidade de tamanhos possíveis de sequência
- n_p a quantidade de *pixels* da imagem
- $n_r(\theta) = \sum_{i=1}^{n_g} \sum_{j=1}^{n_r} P(i, j|\theta)$

Os valores calculados a partir de uma matriz GLRLM são os seguintes:

Ênfase nas sequências curtas

$$\frac{\sum_{i=1}^{n_g} \sum_{j=1}^{n_r} \frac{P(i, j|\theta)}{j^2}}{n_r(\theta)} \tag{3.76}$$

Ênfase nas sequência longas

$$\frac{\sum_{i=1}^{n_g} \sum_{j=1}^{n_r} P(i, j|\theta) j^2}{n_r(\theta)} \tag{3.77}$$

Não uniformidade nos níveis de cinza

$$\frac{\sum_{i=1}^{n_g} \left(\sum_{j=1}^{n_r} P(i, j|\theta) \right)^2}{n_r(\theta)} \tag{3.78}$$

Não uniformidade nos níveis de cinza normalizado

$$\frac{\sum_{i=1}^{n_g} \left(\sum_{j=1}^{n_r} P(i, j|\theta) \right)^2}{n_r(\theta)^2} \tag{3.79}$$

Sequências não uniformes

$$\frac{\sum_{j=1}^{n_r} \left(\sum_{i=1}^{n_g} P(i, j|\theta) \right)^2}{n_r(\theta)} \tag{3.80}$$

Sequências não uniformes normalizado

$$\frac{\sum_{j=1}^{n_r} \left(\sum_{i=1}^{n_g} P(i, j|\theta) \right)^2}{n_r(\theta)^2} \tag{3.81}$$

Porcentagem das sequências

$$\frac{n_r(\theta)}{n_p} \tag{3.82}$$

Variância dos níveis de cinza

$$\begin{aligned} &\sum_{i=1}^{n_g}\sum_{j=1}^{n_r} p(i,j|\theta)(i-\mu)^2 \\ &\mu = \sum_{i=1}^{n_g}\sum_{j=1}^{n_r} p(i,j|\theta)i \end{aligned} \tag{3.83}$$

Variância das sequência

$$\begin{aligned} &\sum_{i=1}^{n_g}\sum_{j=1}^{n_r} p(i,j|\theta)(j-\mu)^2 \\ &\mu = \sum_{i=1}^{n_g}\sum_{j=1}^{n_r} p(i,j|\theta)j \end{aligned} \tag{3.84}$$

Entropia das sequências

$$-\sum_{i=1}^{n_g}\sum_{j=1}^{n_r} p(i,j|\theta)\log_2(p(i,j|\theta)) \tag{3.85}$$

Sequências com os menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_r}\frac{P(i,j|\theta)}{i^2}}{n_r(\theta)} \tag{3.86}$$

Sequências com os maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_r} P(i,j|\theta)i^2}{n_r(\theta)} \tag{3.87}$$

Sequências pequenas com os menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_r}\frac{P(i,j|\theta)}{i^2 j^2}}{n_r(\theta)} \tag{3.88}$$

Sequências pequenas com os maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_r}\frac{P(i,j|\theta)i^2}{j^2}}{n_r(\theta)} \tag{3.89}$$

Sequências longas com os menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_r}\frac{P(i,j|\theta)j^2}{i^2}}{n_r(\theta)} \tag{3.90}$$

Sequências longas com os maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_r}P(i,j|\theta)i^2j^2}{n_r(\theta)} \tag{3.91}$$

3.4.2.4 NGTDM

NGTDM é a sigla para *Neighbouring Gray Tone Difference Matrix* ou Matriz de Diferença de Tons de Cinza Vizinhos. A matriz quantifica a diferença entre um valor de cinza de um *pixel* da média de valores dos *pixels* vizinhos a uma distância σ (AMADASUN; KING, 1989; PYRADIOMICS, 2019). Por exemplo, seja a imagem 5x5 com 5 níveis discretos de intensidade em 3.92, a matriz NGTDM computada com $\sigma = 1$ é aquela em 3.93.

$$\mathbf{I} = \begin{bmatrix} 1 & 2 & 5 & 2 \\ 3 & 5 & 1 & 3 \\ 1 & 3 & 5 & 5 \\ 3 & 1 & 1 & 1 \end{bmatrix} \tag{3.92}$$

i	P_i	p_i	s_i
1	6	0.375	13, 35
2	2	0.125	2, 00
3	4	0.25	2, 63
4	0	0.00	0, 00
5	4	0.25	10, 075

(3.93)

Em que i são os níveis discretos de intensidade, P_i o número de *pixels* com o nível de intensidade i, p_i a probabilidade de ocorrência do nível de intensidade i e s_i a soma das diferenças absolutas para o nível de intensidade i e calculado da forma mostrada em 3.94.

$$\begin{aligned} s_1 &= |1 - 10/3| + |1 - 30/8| + |1 - 15/5| + |1 - 13/5| + |1 - 15/5| + |1 - 11/3| = 13,35 \\ s_2 &= |2 - 15/5| + |2 - 15/5| = 2 \\ s_3 &= |3 - 12/5| + |3 - 18/5| + |3 - 20/8| + |3 - 5/3| = 3,03 \\ s_5 &= |5 - 14/5| + |5 - 18/5| + |5 - 20/8| + |5 - 11/5| = 10,075 \end{aligned} \tag{3.94}$$

Sejam:

- $i = 0,\ 1,\ 2,\ 3,\ \cdots,\ g$ os níveis discretos de intensidade existentes na ROI
- P_i a quantidade de *pixels* com intensidade i
- n_p a quantidade total de *pixels* da ROI
- p_i a probabilidade de ocorrência do nível de intensidade i
- s_i a soma das diferenças absolutas entre cada *pixel* com nível de intensidade i e a média de intensidade dos seus *pixels* vizinhos (calculado da forma que é mostrada em 3.94)
- $N_{v,p}$ a quantidade de *pixels* com pelo menos 1 vizinho na ROI
- $N_{g,p}$ a quantidade de níveis de intensidade com $p_i \neq 0$

Os valores calculados a partir de uma matriz NGTDM são os seguintes:

Rugosidade

$$\frac{1}{\sum_{i=1}^{g} p_i s_i} \tag{3.95}$$

Contraste

$$\left(\frac{1}{N_{g,p}(N_{g,p}-1)} \sum_{i=1}^{g}\sum_{j=1}^{g} p_i p_j (i-j)^2\right)\left(\frac{1}{N_{v,p}} \sum_{i=1}^{g} s_i\right), \text{ em que } p_i \neq 0, p_j \neq 0 \tag{3.96}$$

Ocupação

$$\frac{\sum_{i=1}^{g} p_i s_i}{\sum_{i=1}^{g}\sum_{j=1}^{g} |i p_i - j p_j|}, \text{ em que } p_i \neq 0, p_j \neq 0 \tag{3.97}$$

Complexidade

$$\frac{1}{N_{v,p}} \sum_{i=1}^{g} \sum_{j=1}^{g} |i-j| \frac{p_i s_i + p_j s_j}{p_i + p_j}, \text{ em que } p_i \neq 0, p_j \neq 0 \tag{3.98}$$

Força

$$\frac{\sum_{i=1}^{g} \sum_{j=1}^{g} (p_i + p_j)(i-j)^2}{\sum_{i=1}^{g} s_i}, \text{ em que } p_i \neq 0, p_j \neq 0 \tag{3.99}$$

3.4.2.5 GLDM

GLDM é a sigla para *Gray Level Dependence Matrix* ou Matriz de Dependência de Níveis de Cinza. A matriz quantifica as dependências de níveis de cinza que existem em uma imagem. Um *pixel* **p'** é dependente de um *pixel* central **p** se a distância entre estes os dois *pixels* for menor que um dado Δ ($||p - p'|| \leq \delta$) e a diferença entre seus valores de intensidades for menor que um dado α ($|p - p'| \leq \alpha$). Na matriz GLDM cada elemento (i, j) representa o número de vezes que um *pixel* de intensidade i teve j *pixels* dependentes (ZWANENBURG et al., 2016). Por exemplo, seja a imagem 5x5 com 5 níveis discretos de intensidade em 3.100, a matriz GLDM computada com $\alpha = 0$ e $\Delta = 1$ é aquela em 3.101.

$$\mathbf{I} = \begin{bmatrix} 5 & 2 & 5 & 4 & 4 \\ 3 & 3 & 3 & 1 & 3 \\ 2 & 1 & 1 & 1 & 3 \\ 4 & 2 & 2 & 2 & 3 \\ 3 & 5 & 3 & 3 & 2 \end{bmatrix} \tag{3.100}$$

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 2 & 1 \\ 1 & 2 & 3 & 0 \\ 1 & 4 & 4 & 0 \\ 1 & 2 & 0 & 0 \\ 3 & 0 & 0 & 0 \end{bmatrix} \tag{3.101}$$

Sejam:

- $P(i, j)$ a matriz GLDM
- n_g a quantidade de níveis de intensidade da imagem
- n_d a quantidade de tamanhos possíveis de dependências

- n_z a quantidade de regiões de dependência da imagem que é igual a $\sum_{i=1}^{n_g} \sum_{j=1}^{n_d} P(i,j)$
- $p(i,j)$ a matriz GLDM normalizada definida como $p(i,j) = \dfrac{P(i,j)}{n_z}$

Os valores calculados a partir de uma matriz GLDM são os seguintes:

Ênfase nas pequenas dependência

$$\frac{\sum_{i=1}^{n_g} \sum_{j=1}^{n_d} \frac{P(i,j)}{i^2}}{n_z} \tag{3.102}$$

Ênfase nas grandes dependências

$$\frac{\sum_{i=1}^{n_g} \sum_{j=1}^{n_d} P(i,j) j^2}{n_z} \tag{3.103}$$

Não uniformidade dos níveis de cinza

$$\frac{\sum_{i=1}^{n_g} \left(\sum_{j=1}^{n_d} P(i,j) \right)^2}{n_z} \tag{3.104}$$

Não uniformidade das dependências

$$\frac{\sum_{j=1}^{n_d} \left(\sum_{i=1}^{n_g} P(i,j) \right)^2}{n_z} \tag{3.105}$$

Não uniformidade das dependência normalizado

$$\frac{\sum_{j=1}^{n_d} \left(\sum_{i=1}^{n_g} P(i,j) \right)^2}{n_z^2} \tag{3.106}$$

Variância nos níveis de cinza

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_d} p(i,j)(i-\mu)^2, \text{ em que } \mu = \sum_{i=1}^{n_g} \sum_{j=1}^{n_d} i p(i,j) \tag{3.107}$$

Variância das dependências

$$\sum_{i=1}^{n_g} \sum_{j=1}^{n_d} p(i,j)(j-\mu)^2, \text{ em que } \mu = \sum_{i=1}^{n_g} \sum_{j=1}^{n_d} j p(i,j) \tag{3.108}$$

Entropia das dependências

$$-\sum_{i=1}^{n_g}\sum_{j=1}^{n_d} p(i,j)\log_2(p(i,j)) \tag{3.109}$$

Ênfase nos menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_d}\frac{P(i,j)}{i^2}}{N_z} \tag{3.110}$$

Ênfase nos maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_d} P(i,j)i^2}{n_z} \tag{3.111}$$

Pequenas dependências com os menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_d}\frac{P(i,j)}{i^2j^2}}{n_z} \tag{3.112}$$

Pequenas dependências com os maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_d}\frac{P(i,j)i^2}{j^2}}{n_z} \tag{3.113}$$

Grandes dependências com os menores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_d}\frac{P(i,j)j^2}{i^2}}{n_z} \tag{3.114}$$

Grandes dependências com os maiores níveis de cinza

$$\frac{\sum_{i=1}^{n_g}\sum_{j=1}^{n_d} P(i,j)i^2j^2}{n_z} \tag{3.115}$$

3.5 Características Morfológicas

As características morfológicas associadas à cor dos linfócitos utilizadas neste trabalho foram: *matiz*, *saturação* e *valor*, que são as médias de cada canal de cor da imagem no espaço de cores HSV; *vermelho*, *verde* e *azul* que são as média de cada canal de cor da imagem no espaço de cores RGB.

Com relação as características morfológicas associadas ao contorno e formato do linfócito tem-se: *área*, *excentricidade*, *elongação* e *perímetro do linfócito*; *área convexa*; *perímetro convexo*; *tamanho do eixo maior*; *tamanho do eixo menor*; *proporção*; *diâmetro equivalente*; *retangularidade*; *compacidade*; *convexidade* e *consistência*. O cálculo destas características é feito com as seguintes equações:

Área A área, em *pixels*, ocupada pelo linfócito na imagem

Perímetro Perímetro do linfócito

Área Convexa Área do menor casco convexo (*Convex Hull*) que contenha o linfócito

Perímetro Convexo Perímetro do menor casco convexo (*Convex Hull*) que contenha o linfócito

Diâmetro Equivalente

$$\sqrt{\frac{4 \cdot \text{Área}}{\pi}} \tag{3.116}$$

Compacidade

$$\frac{4 \cdot \pi \cdot \text{Área}}{(\text{Perímetro})^2} \tag{3.117}$$

Convexidade

$$\frac{\text{Perímetro Convexo}}{\text{Perímetro}} \tag{3.118}$$

Consistência

$$\frac{\text{Área}}{\text{Área Convexa}} \tag{3.119}$$

Excentricidade A excentricidade da menor elipse que contenha o linfócito

Eixo Maior Eixo Maior da Elipse que contenha o linfócito

Eixo Menor Eixo Menor da Elipse que contenha o linfócito

Proporção

$$\frac{\text{Eixo Menor}}{\text{Eixo Maior}} \tag{3.120}$$

Elongação

$$1 - \frac{\text{Eixo Menor}}{\text{Eixo Maior}} \tag{3.121}$$

Retangularidade

$$\frac{\text{Área}}{\text{Eixo Maior} \cdot \text{Eixo Menor}} \tag{3.122}$$

3.6 Características de Contorno

O momento bi-dimensional de uma imagem binária $I(x, y)$ de ordem p e q é dado pela equação

$$m_{p,q} = \sum_x \sum_y x^p y^q I(x, y) \Delta A, \text{ onde } \Delta A \text{ é a área de 1 } pixel. \tag{3.123}$$

As coordenadas $\overline{x}$ e $\overline{y}$ do centroide de uma imagem binária é dada pelas equações

$$\begin{aligned} \overline{x} &= \frac{m_{1,0}}{m_{0,0}} \\ \overline{y} &= \frac{m_{0,1}}{m_{0,0}} \end{aligned} \tag{3.124}$$

Seja

$$c(t) = \left\{ \big(x(1), y(1)\big),\ \big(x(2), y(2)\big),\ \cdots,\ \big(x(n), y(n)\big) \right\} \quad t = 1, 2, 3, \cdots, n \tag{3.125}$$

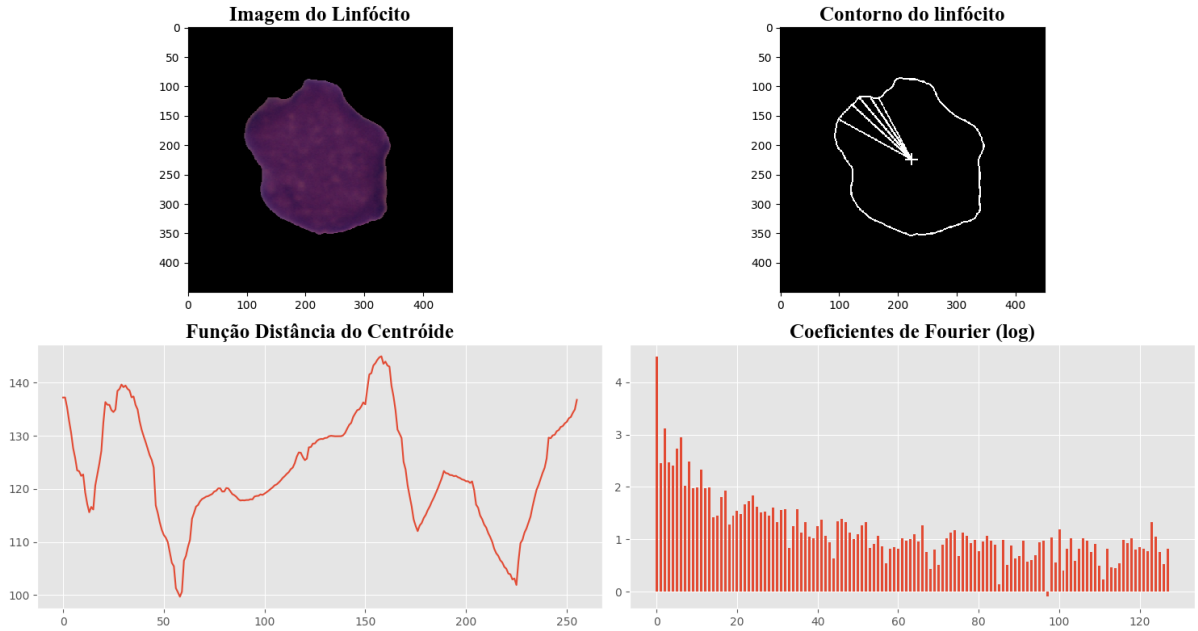
o conjunto dos n pontos que definem as coordenadas dos *pixels* que fazem parte do contorno de um objeto, e seja $(\overline{x}, \overline{y})$ o seu centroide. A função distância do centroide (*Centroid Distance Function*) é formada pelas distâncias euclidianas entre o centroide e os pontos do contorno conforme a equação 3.126.

$$r(t) = ||c(t) - (\overline{x}, \overline{y})|| = \sqrt{\big(x(t) - \overline{x}\big)^2 + \big(y(t) - \overline{y}\big)^2} \quad t = 1, 2, 3, 4, \cdots, n \tag{3.126}$$

Em processamento de imagens a função distância do centroide, às vezes chamada de assinatura da forma (*shape signature*), é considerada um descritor do formato de objetos. O trabalho de COSGRIFF (1960) foi o primeiro a propor o uso dos descritores de *fourier* para identificar objetos a partir da sua assinatura da forma. A ideia é representar o contorno por um conjunto de números que represente as frequências que formam o contorno do objeto. No caso dos descritores de *fourier* este conjunto é formado pelos coeficientes da transformada discreta de fourier da função distância do centroide (ALHILAL et al., 2015; UCI DEPARTMENT OF MATHEMATICS, 2011; NIXON, 2008).

Neste trabalho os coeficientes da transformada de fourier da função distância computada dos linfócitos foram utilizados como características. Um exemplo ilustrativo deste processo pode ser visto na Figura 34.

Figura 34 – Exemplo dos descritores de Fourier da função distância do centroide. Os coeficientes de *fourier* passaram por uma transformação logarítmica para melhor visualização gráfica.



3.7 Características da Transformada Discreta do Cosseno

A Transformada Discreta do Cosseno é uma transformada real $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ que apresenta propriedades de compactação de energia (NIXON, 2008). Seja $I_{x,y}$ uma imagem de tamanho $N \times N$, os componentes espectrais $DP_{u,v}$ da transformada do cosseno serão calculados pela equação

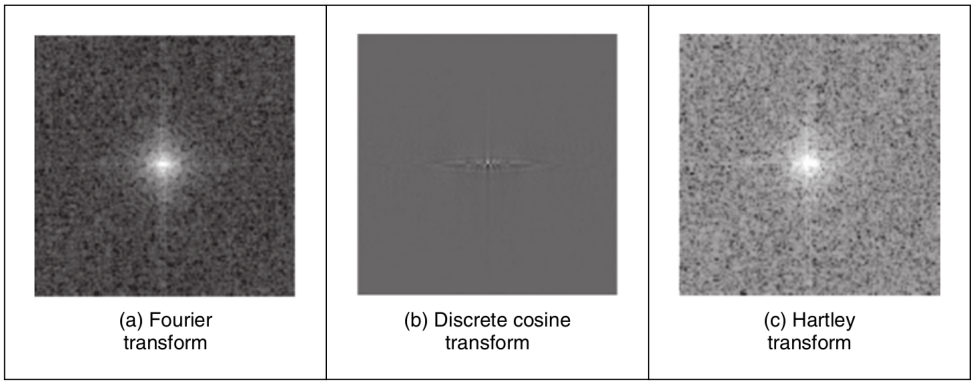
$$DP_{u,v} = \begin{cases} \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} I_{x,y} & \text{se} \quad u = 0 \; e \; v = 0 \\ \frac{2}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} I_{x,y} \cos(\frac{(2x+1)u\pi}{2N}) \cos(\frac{(2x+1)v\pi}{2N}) & . \end{cases} \tag{3.127}$$

A transformada inversa será dada pela equação

$$I_{x,y} = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} DP_{u,v} \cos(\frac{(2x+1)u\pi}{2N}) \cos(\frac{(2x+1)v\pi}{2N}) \tag{3.128}$$

A propriedade de compactar a energia dos componentes espectrais de alta energia das imagens é um dos motivos da DCT ser a forma de codificação padrão dos formatos JPEG e MPEG. Está propriedade pode ser vista na Figura 35.

Figura 35 – Comparação dos resultados de 3 transformações diferentes com imagem da Lena. Visualmente nota-se que a DCT tem a distribuição mais compacta quando comparada com as outras transformadas.



Fonte: Nixon (2008)

A contraparte da DCT é a Transformada Discreta do Seno (DST), entretanto, por usar como base o seno, que é uma função ímpar, essa transformada tem propriedades menos desejáveis quando comparada com a DCT e, por consequência, apresenta menos aplicações.

3.8 Métricas de desempenho da Classificação Binária

A classificação binária é a tarefa de classificar os elementos de um domínio entre duas classes distintas. O resultado gerado a partir de uma classificação binária pode ser: verdadeiro positivo, falso positivo, falso negativo ou verdadeiro negativo. A formação destes resultados está presente no Quadro 3.

Quadro 3 – Tipos de resultados obtidos na classificação binária

Resultados		Valor real da observação	
		Positivo	**Negativo**
Valor inferido da observação	**Positivo**	Verdadeiro Positivo	Falso Positivo
	Negativo	Falso Negativo	Verdadeiro Negativo

O Verdadeiro Positivo (VP) ocorre quando o elemento é positivo e o modelo infere como positivo, o Falso Positivo (FP) ocorre quando o elemento é negativo e o modelo infere como positivo, o que gera um erro do tipo I. O Verdadeiro Negativo (VN) ocorre quando o elemento é negativo e o modelo infere como negativo, o Falso Negativo (FN) ocorre quando o elemento é positivo e o modelo infere como negativo, o que gera um erro do tipo II.

Um erro do tipo I ocorre quando a hipótese nula é verdadeira, mas é rejeitada, o erro do tipo II ocorre quando a hipótese nula é falsa, mas não é rejeitada. Assumindo-se um

classificador binário que infere se um linfócito é saudável ou maligno e que a hipótese nula seja H_0 : *O linfócito é saudável* tem-se:

Erro do tipo I Se o classificador inferir que a célula é maligna, sendo que ela saudável (Falso Positivo - falha em aceitar H_0);

Erro do tipo II Se o classificador inferir como saudável, sendo que ela é maligna (Falso Negativo - falha em rejeitar H_0).

A partir dos registros de VP, FP, VN e FN obtidos pelo classificador são calculadas as principais métricas de desempenho dos classificadores binários que são: acurácia, precisão, especificidade, sensibilidade e *F1-score* (STRALEN et al., 2009; MORADIAMIN et al., 2016; MOURYA et al., 2018).

A acurácia é a taxa de acerto global do classificador definida pela razão entre as inferências corretas e todas as inferências realizadas.

$$\text{Acurácia} = \frac{\text{VP} + \text{VN}}{\text{VP} + \text{VN} + \text{FP} + \text{FN}} \tag{3.129}$$

A precisão ou valor preditivo positivo mede a taxa de acerto dentro da classe verdadeira. É definida pela razão entre as inferências corretas da classe verdadeira e todas as inferências feitas dentro desta classe. Uma precisão ser igual a 1 indica que o classificador consegue inferir corretamente os elementos da classe verdadeira.

$$\text{Precisão} = \frac{\text{VP}}{\text{VP} + \text{FP}} \tag{3.130}$$

A especificidade ou taxa de verdadeiro negativo mede a taxa de acerto dentro da classe falsa, é definida pela razão entre as inferências corretas da classe falsa e todas as inferências feitas dentro desta classe. Uma especificidade igual a 1 indica que o classificador consegue inferir corretamente os elementos da classe falsa.

$$\text{Especificidade} = \frac{\text{VN}}{\text{VN} + \text{FP}} \tag{3.131}$$

A sensibilidade ou *recall* mede a taxa de elementos verdadeiros que foram corretamente inferidos pelo classificador, é definida pela razão entre as inferências corretas da classe verdadeira com a quantidade de elementos verdadeiros existentes. Uma sensibilidade igual a 1 indica que o classificador conseguiu reconhecer todos os elementos verdadeiros existentes.

$$\text{Sensibilidade} = \frac{\text{VP}}{\text{VP} + \text{FN}} \tag{3.132}$$

O *F1-score* é a média harmônica entre a precisão e a sensibilidade. A média harmônica é uma das médias pitagóricas, e é ideal para o cálculo da média entre duas taxas.

$$F_1 = 2 \cdot \frac{\text{Precisão} \cdot \text{Sensibilidade}}{\text{Precisão} + \text{Sensibilidade}} = \frac{2\,\text{VP}}{2\,\text{VP} + \text{FP} + \text{FN}} \tag{3.133}$$

No próximo capitulo será apresentada a metodologia adotada na realização dos experimentos e geração de resultados.

4
Metodologia

Neste capítulo serão apresentados os materiais e recursos utilizados bem como a metodologia adotada na construção dos experimentos e geração dos resultados.

4.1 Recursos utilizados

Os trabalhos foram realizados num Notebook Dell Vostro VT-5470 com processador *Intel Core i5* de 1,60 GHz com 4 núcleos e 8 GB RAM, utilizando uma placa de vídeo *GeForce GT 740M* da *Nvidia*. Outro computador utilizado foi uma máquina virtual da *Google Cloud Platform* com um processador *Intel Core i5* de 2,40 GHz com 4 núcleos e 20 GB RAM, utilizando uma placa de vídeo *Tesla T4* também da *Nvidia*.

Todos os códigos foram escritos na linguagem *Python* devido ao suporte oferecido por meio de bibliotecas de código, APIs e *Frameworks* especializados em Computação Científica, Processamento de Imagens, Aprendizagem de Máquina e Inteligência Artificial.

O conjunto de imagens de linfócitos saudáveis e malignos utilizados para treino, validação e teste dos classificadores foi fornecido por SBILab (2019).

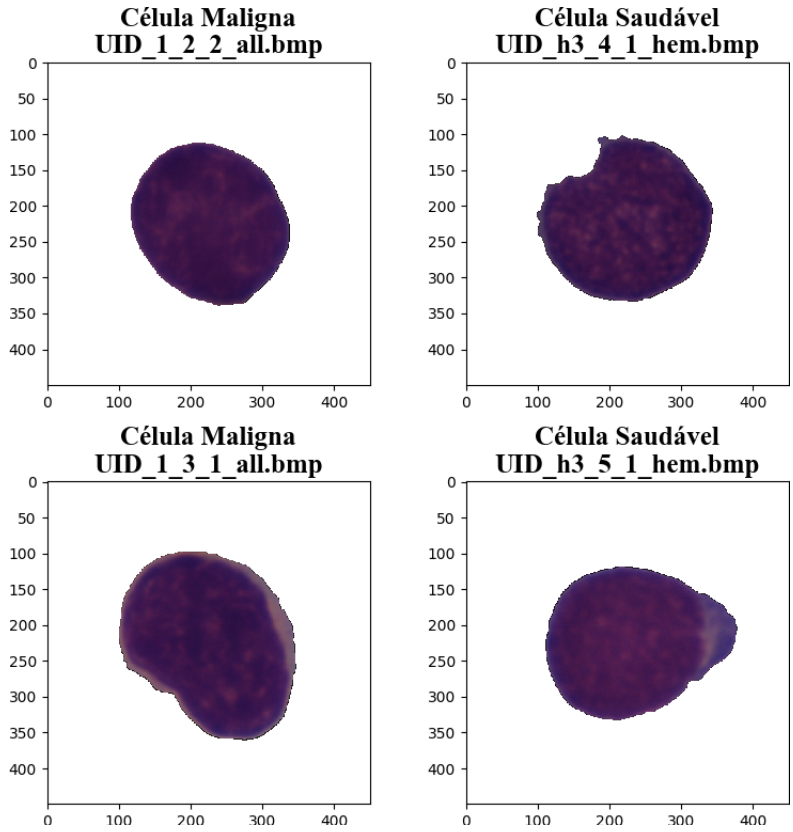
4.2 Descrição das imagens utilizadas

O conjunto de imagens é composto por linfócitos normais e malignos que foram previamente identificados por oncologistas experientes. É formado por 10661 imagens, que provêm de 73 pacientes diferentes, destas: 7272 são de linfócitos malignos de LLA de todos os subtipos (L1 a L3) extraídos de 47 pacientes acometidos pela doença, e 3389 são de linfócitos normais extraídos de 26 pacientes saudáveis.

As células foram segmentadas de imagens *Bitmap* RGB com *color depth* de 24 *bits* e resolução 2560x1920. O comprimento máximo das células varia entre 200 a 400 *pixels*, por

este motivo, após segmentadas, cada célula foi posicionada no centro de uma imagem branca de tamanho 450x450 *pixels*. O ruído introduzido pelo processo de coloração dos linfócitos (*Stain Noise*) foi previamente removido das imagens pela organização que disponibilizou o *dataset*. Na Figura 36 podem ser vistas algumas destas imagens.

Figura 36 – Exemplos de imagens das células do *dataset*



4.3 Criação dos conjuntos de Treinamento, Validação e Teste

As imagens utilizadas para o treinamento, validação e teste dos classificadores foram divididas de duas formas diferentes e recebem os nomes de **divisão aleatória** e **divisão por paciente**.

Na **divisão aleatória** os linfócitos de todos os pacientes foram distribuídos aleatoriamente entre os conjuntos de treino, validação e teste seguindo a proporção de 6:3:1. A divisão nessa proporção foi feita para os linfócitos malignos e saudáveis separadamente. Desta forma a divisão dos 7272 linfócitos malignos deu-se da seguinte forma: 4364 foram para o conjunto de treino, 2181 para o conjunto de validação e 727 para o conjunto de teste. E com relação aos 3389 linfócitos saudáveis: 2034 foram para o conjunto de treino, 1016 para o conjunto de validação e 339 para o conjunto de teste. O código que realiza a divisão aleatória se encontra no apêndice A.

Na **divisão por paciente** os pacientes foram distribuídos aleatoriamente entre os conjuntos de treino, validação e teste de modo que as imagens de seus linfócitos pertençam somente a um

dos conjuntos (treinamento ou validação ou teste). A distribuição dos pacientes entre os três conjuntos seguiu a proporção 6:3:1 e foi feita com os pacientes doentes e saudáveis separadamente. Desta forma dos 47 pacientes acometidos por LLA: 29 foram para o conjunto de treino, 14 para o conjunto de validação e 4 para o conjunto de teste. Com relação aos 26 pacientes saudáveis: 16 foram para o conjunto de treino, 7 para o conjunto de validação e 3 para o conjunto de teste. O código que realiza a divisão por paciente se encontra no apêndice B.

A divisão do *dataset* desta maneira criou 6 conjuntos de imagens, armazenados cada qual em seu próprio diretório. Estes conjuntos foram nomeados da seguinte forma:

rndDiv_train Imagens para treino originárias da divisão aleatória, é composto por 4364 linfócitos malignos e 2034 normais.

rndDiv_valid Imagens para validação originárias da divisão aleatória, é composto por 2181 linfócitos malignos e 1016 normais.

rndDiv_test Imagens para teste originárias da divisão aleatória, é composto por 727 linfócitos malignos e 339 normais.

patLvDiv_train Imagens para treino originárias da divisão por paciente, é composto por 4776 linfócitos provenientes de 29 pacientes com LLA e 2448 linfócitos provenientes de 16 pacientes saudáveis.

patLvDiv_valid Imagens para validação originárias da divisão por paciente, é composto por 2128 linfócitos provenientes de 14 pacientes com LLA e 759 linfócitos provenientes de 7 pacientes saudáveis.

patLvDiv_test Imagens para teste originárias da divisão por paciente, é composto por 368 linfócitos provenientes de 4 pacientes com LLA e 182 linfócitos provenientes de 3 pacientes saudáveis.

Foram criados mais 4 diretórios com a utilização da técnica de *data augmentation* a partir das imagens contidas em *rndDiv_train*, *rndDiv_valid*, *patLvDiv_train* e *patLvDiv_valid* respectivamente. Essa técnica é utilizada para aumentar o número de imagens disponíveis e, ao mesmo tempo, balancear a quantidade de linfócitos malignos e normais, os diretórios de teste não passaram por este processo. As técnicas utilizadas para produzir as imagens adicionais foram: desfoque gaussiano (*Gaussian blur*) com um *kernel* 5x5; espelhamento vertical ou horizontal ou ambos (escolhido de forma aleatória); rotações entre 20° e 340° (ângulo de rotação escolhido de forma aleatória); transformação de cisalhamento (*shear transformation*) com o fator de cisalhamento entre 0,1 e 0,35 (escolhido de forma aleatória); e adição de ruído *Salt-and-Pepper* em 2% dos *pixels* da imagem e com proporção 1:1 de sal e pimenta. O espelhamento e rotação são transformações que ocorrem em todas as novas imagens geradas, no entanto as transformações

de desfoque Gaussiano, cisalhamento e adição de ruído *Salt-and-Pepper* tem uma probabilidade de aproximadamente 25% de ocorrerem em cada nova imagem gerada. O pseudo-código do procedimento de geração das imagens por meio de *data augmentation* pode ser consultado em 2.

Código 2 – Pseudo-código do procedimento de *data augmentation*

```
Entrada: srcImg (Imagem base)
Saída: augImg (Nova imagem gerada a partir da imagem base)
início
    augImg ← espelhamento(srcImg);
    augImg ← rotação(augImg);
    num ← variável aleatória com uma distribuição Uniforme no intervalo [0,1] ;
    se 0 ≤ num < 0,25 então
        retorna augImg;
    se 0,25 ≤ num < 0,50 então
        augImg ← cisalhamento(augImg);
        retorna augImg;
    se 0,50 ≤ num < 0,75 então
        augImg ← ruidoSaltPepper(augImg);
        retorna augImg;
    se num ≥ 0,75 então
        augImg ← desfoqueGaussiano(augImg);
        retorna augImg;
fim
```

O resultado do procedimento de *data augmentation* foram 4 novos diretórios que recebem os seguintes nomes:

augm_rndDiv_train Contém a cópia das imagens do diretório *rndDiv_train* acrescentado das imagens produzidas pelo processo de *data augmentation* utilizando como base as imagens contidas em *rndDiv_train*, é composto por 10000 linfócitos malignos e 10000 normais.

augm_rndDiv_valid Contém a cópia das imagens do diretório *rndDiv_valid* acrescentado das imagens produzidas pelo processo de *data augmentation* utilizando como base as imagens contidas em *rndDiv_valid*, é composto por 5000 linfócitos malignos e 5000 normais.

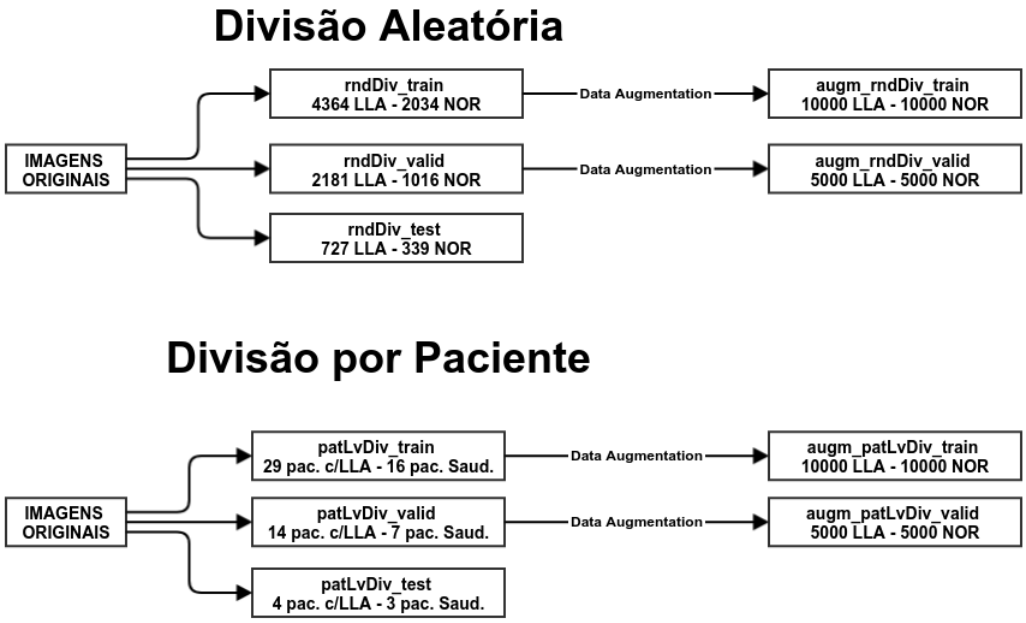
augm_patLvDiv_train Contém a cópia das imagens do diretório *patLvDiv_train* acrescentado das imagens produzidas pelo processo de *data augmentation* utilizando como base as imagens contidas em *patLvDiv_train*, é composto por 10000 linfócitos malignos e 10000 normais.

augm_patLvDiv_valid Contém a cópia das imagens do diretório *patLvDiv_valid* acrescentado das imagens produzidas pelo processo de *data augmentation* utilizando

como base as imagens contidas em *patLvDiv_valid*, é composto por 5000 linfócitos malignos e 5000 normais.

Ao fim deste processo haverá 10 diretórios no total, um resumo de todo o procedimento pode ser visto na Figura 37.

Figura 37 – Processo de repartição das imagens entre os diretórios de Treino, Validação e Teste



O objetivo de dividir os conjuntos de treino, validação e teste destas duas formas foi para avaliar a afirmação feita por Mourya et al. (2018), que diz que as variações celulares particulares de cada paciente podem, de alguma forma, influenciar o desempenho do classificador, bem como avaliar o impacto que o emprego da técnica de *data augmentation* tem sobre o desempenho dos classificadores.

4.4 Extração de características

Para cada linfócito presente em cada imagem foram extraídas as seguintes características: estatísticas de 1ª ordem, de textura, morfológicas, de contorno e da transforma discreta do cosseno. Para tal foi usada a linguagem de programação *Python* v3.6 associada com os pacotes: *NumPy* v1.16.2, *SciPy* v1.2.1, *Scikit-Image* v0.14.2, *Scikit-Learn* v0.20.1, *Pandas* v0.24.1, *SimpleITK* v1.1.0, *Pyradiomics* v2.1.2, *Tensorflow-gpu* v1.12.0, *Keras* v2.2.4 e *Tensorboard* v1.12.0.

A biblioteca *Pandas* foi utilizada para manipular, organizar e realizar a persistência de todas as características extraídas, isso é feito estruturando os dados de uma forma tabular. Essa estrutura de dados recebe o nome de *dataframe*. Todos os *dataframes* sempre serão ajustados para que fiquem balanceados (quantidades iguais de linfócitos malignos e normais) antes de

serem usados em qualquer processo de treino, validação ou teste. Um exemplo de *dataframe* pode ser visto na Tabela 1.

Tabela 1 – Exemplo de um *dataframe* que contém estatísticas de 1ª ordem extraídas dos histogramas das imagens dos linfócitos (cada linha da Tabela representa um linfócito)

índice	ALL=1; HEM=-1	B-10th-percentile	...	H-skewness	H-standard-deviation	...	V-uniformity	V-variance
0	1,0	88,0	...	0,24	3,18	...	0,72	79,938
1	-1,0	61,0	...	0,89	5,12	...	0,61	104,34
2	-1,0	63,0	...	0,06	3,84	...	0,51	180,22
3	1,0	72,0	...	-0,23	4,34	...	0,58	100,84
.	.	.	...	.	.	...	.	.
.	.	.	...	.	.	...	.	.

Utilizando a biblioteca *pyradiomics* (GRIETHUYSEN et al., 2017) foram computadas as estatísticas de 1ª ordem dos histogramas de cada canal de cor das imagens nos espaços de cores RGB e HSV seguindo a metodologia utilizada por MoradiAmin et al. (2016). As estatísticas extraídas de cada histograma foram: média, desvio padrão, variância, mediana, valor mínimo, valor máximo, intervalo de variação, 10º percentil, 90º percentil, variação interquartil, curtose, assimetria, raiz do valor quadrático médio, desvio absoluto da média, desvio absoluto da média robusta, uniformidade, energia e entropia. Ao todo foram extraídas 108 características de cada imagem (18 características para cada um dos seis canais de cor).

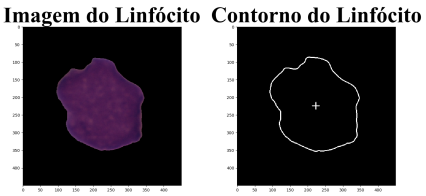
As características morfológicas associadas a cor dos linfócitos foram: *matiz*, *saturação* e *valor*, que são as médias de cada canal de cor da imagem em HSV; *vermelho*, *verde* e *azul* que são as médias de cada canal de cor da imagem em RGB. Àquelas associadas ao contorno e formato do linfócito foram: *área*, *excentricidade*, *elongação* e *perímetro do linfócito*, *área convexa*, *perímetro convexo*, *tamanho do eixo maior*, *tamanho do eixo menor*, *proporção*, *diâmetro equivalente*, *retangularidade*, *compacidade*, *convexidade* e *consistência*. A formulação matemática de cada uma destas características segue a metodologia de Houby (2018), MoradiAmin et al. (2016) e Putzu et al. (2014) e foram apresentadas seção 3.5. Para estes cálculos foi utilizada a biblioteca *opencv* e *numpy*. Ao todo foram extraídas 20 características morfológicas de cada imagem.

As características de textura foram extraídas das imagens dos linfócitos em escala de cinza utilizando a biblioteca *pyradiomics* seguindo a padronização do IBSI, apresentada na seção 3.4. Ao todo foram computadas: 24 características relacionadas a matriz GLCM, 16 características da matriz GLRLM, 16 características da matriz GLSZM, 5 características da matriz NGTDM e 14 características da matriz GLDM, totalizando 75 características para cada imagem (PUTZU et al., 2014; MORADIAMIN et al., 2016; MISHRA et al., 2017).

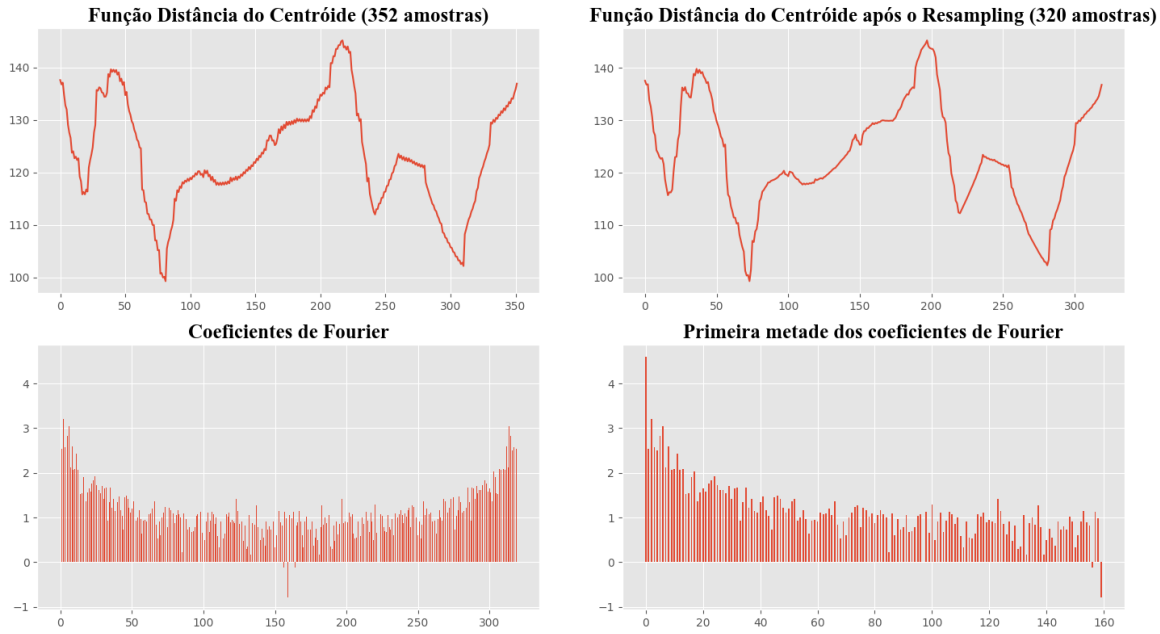
As características de contorno foram os coeficientes da Transformada Discreta de Fourier da função distância do centroide, calculada a partir do contorno dos linfócitos. O contorno dos linfócitos é definido em média por 320 *pixels* com um desvio padrão de 78 *pixels*, que também é a quantidade de amostras discretas da função distância do centroide. Como os algoritmos de classificação exigem como entrada vetores de mesmo tamanho, a função distância passa por

um processo de *resample* para 320 amostras, em seguida é realizada a transformada discreta de *fourier*, o que produz 320 coeficientes. Como o espectro se apresenta espelhado a partir do 160º coeficiente, foram utilizados apenas os primeiros 160 coeficientes de *fourier* como características (*fourier descriptors*). Todo este processo está ilustrado na Figura 38.

Figura 38 – Processo de extração das características de contorno



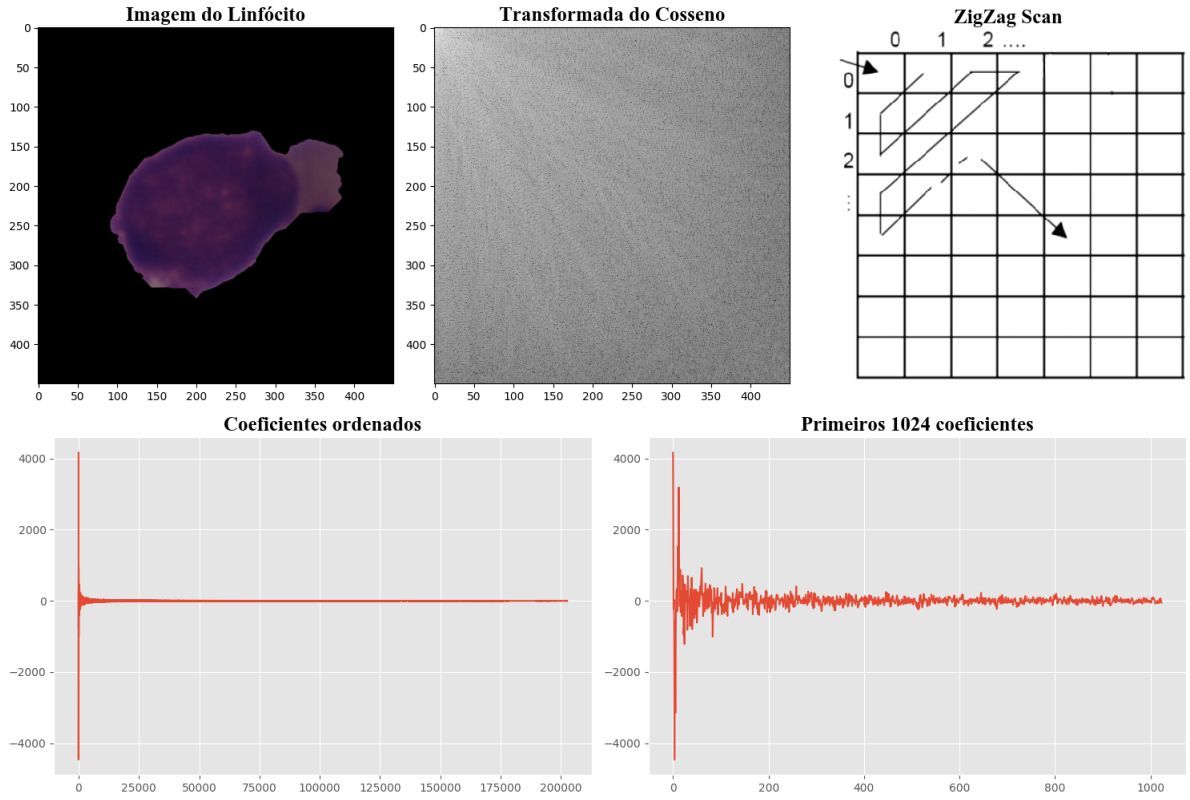
(a) Exemplo de linfócito e seu contorno



(b) Processo de extração das características

Para as características da Transformada Discreta do Cosseno, primeiramente a imagem é transformada para escala de cinza, logo após é efetuada a Transformada Discreta do Cosseno com o auxílio da biblioteca *opencv*. Uma imagem de tamanho 450x450 produz uma matriz também de tamanho 450x450 com os coeficientes da transformada do cosseno, ou seja, 202.500 coeficientes para cada imagem. Entretanto, devido a propriedade de compactação de energia da transformada do cosseno, os coeficientes da frequências mais baixas compactam a maior parte da energia da imagem, por este motivo os coeficientes das frequências mais altas são descartados. Uma forma de ordenar os coeficientes da frequência mais baixa até a mais alta é percorrendo a matriz de coeficientes em zigue-zague (*zig-zag scan*) (WU, 2002; VISHWAKARMA et al., 2007). Após ordenados, os primeiros 1024 coeficientes, dentre os 202.500, foram utilizados como características de cada imagem. Este processo está ilustrado na Figura 39.

Figura 39 – Processo de extração das características da transformada do cosseno



4.5 Desenvolvimento dos classificadores

As características extraídas e armazenadas nos *dataframes* foram utilizadas com os seguintes algoritmos de classificação: SVM com kernel linear, quadrático, polinomial (utilizando um polinômio de grau 3), RBF (*Radial Basis Function*) e Sigmoide, foi padronizado o fator de regularização $C = 1$ para todos os classificadores SVM; *K-Nearest-Neighbors* (KNN), *Random Forest* (RF) e Redes Neurais. Os classificadores SVM, KNN e RF foram desenvolvidos com a biblioteca *Scikit-Learn*, as Redes Neurais foram desenvolvidas utilizando *Tensorflow* em conjunto com a API *Keras*.

Cada linha, de cada *dataframe*, contém o rótulo com a classificação do linfócito e os elementos do seu vetor de características, como exemplificado na Tabela 1. Foram avaliados os desempenho de cada conjunto de características separadamente, desta forma é possível saber como cada um deles se comporta quando utilizados com algoritmos de aprendizagem de máquina. Esta análise individualizada dos conjuntos de características não seria possível caso todos os conjuntos de características fossem concatenados num único *dataframe*.

As 108 estatísticas de 1ª ordem que foram utilizadas para desenvolver os classificadores SVM, KNN e RF passaram pelo processo de redução de dimensionalidade utilizando PCA e foi avaliado o uso de: 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95 e 100 componentes principais e, por fim, foram utilizadas as 108 características sem redução de

dimensionalidade.

As 20 características morfológicas que foram utilizadas para desenvolver os classificadores SVM, KNN e RF passaram pelo processo de redução de dimensionalidade utilizando PCA e foi avaliado o uso de: 5, 10 e 15 componentes principais e, por fim, foram utilizadas as 20 características sem redução de dimensionalidade.

As 75 características relacionadas a textura que foram utilizadas para desenvolver os classificadores SVM, KNN e RF passaram pelo processo de redução de dimensionalidade utilizando PCA e foi avaliado o uso de: 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65 e 70 componentes principais e, por fim, foram utilizadas as 75 características sem redução de dimensionalidade.

As 160 características de contorno foram utilizadas para desenvolver os classificadores SVM, KNN e RF passaram pelo processo de redução de dimensionalidade utilizando PCA e foi avaliado o uso de: 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140 e 150 componentes principais e, por fim, foram utilizadas as 160 características sem redução de dimensionalidade.

As 1024 características da transformada discreta do cosseno que foram utilizadas para desenvolver os classificadores SVM, KNN e RF passaram pelo processo de redução de dimensionalidade utilizando PCA e foi avaliado o uso de: 50, 100, 150, 200, 300, 400, 600 e 800 componentes principais e, por fim, foram utilizadas as 1024 características sem redução de dimensionalidade.

Nos classificadores KNN foi avaliada a influência da quantidade de vizinhos utilizados para classificação, foram testados de 5 a 20 vizinhos em cada treinamento. Nos casos em que há um número par de vizinhos e o número de vizinhos é o mesmo para as duas classes, a biblioteca *Scikit-Learn* tem como padrão retornar a classe do primeiro elemento avaliado. Também segundo a documentação da biblioteca *Scikit-Learn* o classificador RF apresenta desempenho satisfatório na maioria dos problemas utilizando 100 árvores aleatórias, entretanto foi avaliado o seu desempenho com 50, 70, 90, 110, 130 e 150 árvores aleatórias (SCIKIT-LEARN, 2019).

Cada um dos conjuntos de características também foi utilizado pra treinar redes neurais multicamadas, porém não foi utilizado o PCA para redução de dimensionalidade nos dados de entrada destas redes. Esta escolha foi feita por não haver menção de uso do PCA em conjunto com redes neurais multicamadas dentro da bibliografia pesquisada. Entretanto, no que diz respeito a redução de dimensionalidade em redes neurais, vale mencionar que o *Backpropagation* (algoritmo utilizado para treinar as redes neurais) se mostra eficaz para redução da dimensionalidade em conjuntos de dados não-lineares como mostrado por Hinton (2006) em seu artigo sobre *autoencoders*.

As redes neurais foram criadas variando-se 3 parâmetros: quantidade de camadas, 2, 3 ou quatro camadas; função de ativação, ReLU, Sigmoide ou Tangente Hiperbólica; e quantidade de neurônios nas camadas, 32, 128, 512 ou 1024.

Os conjuntos de treino e validação de cada grupo de características foram utilizados para otimizar os parâmetros internos de cada classificador para que obtivessem as maiores acurácias e, ao fim, seus desempenhos foram mensurados com os conjuntos de teste. Todos os conjuntos de treinamento foram balanceados antes de serem utilizados em qualquer processo de treinamento, ou seja, a quantidade de elementos de cada classe (malignos e saudáveis) eram iguais. Cada característica foi normalizada subtraindo-se a sua média e em seguida dividindo-a pelo seu respectivo desvio padrão antes de serem utilizadas em qualquer processo de treino, validação ou teste.

Houve também o treinamento de redes convolucionais com as seguintes arquiteturas: *Xception*, VGG16, VGG19, *ResNet50* e *InceptionV3*. Estas arquiteturas recebem como entrada a imagem em RGB do linfócito pré-processada. O tamanho das imagens foi reduzido realizando-se um corte com tamanho de 100 *pixels* em todas as bordas, reduzindo a área branca ao redor do linfócito (*crop*), isto resultou em imagens de tamanho 250x250. As imagens dos linfócitos apresentam a característica morfológica de Eixo Maior com média de 223 *pixels* e desvio padrão de 43 *pixels*, por isto, ao realizar este corte nas bordas, é possível reduzir o custo computacional do treinamento da rede sem descartar muito a área útil das imagens. O pré-processamento feito nas imagens é realizado conforme o Código 3.

Código 3 – Código responsável pelo pré-processamento das imagens utilizadas utilizadas nas redes convolucionais.

```
import cv2 #opencv
import numpy as np
def processImg(imgPath, imgSize=(250,250)):
  #abrir a imagem
  img = cv2.cvtColor(cv2.imread(str(imgPath)), cv2.COLOR_BGR2RGB)
  #Cortar as bordas
  width, height, _ = img.shape
  w_crop = int(np.floor((width-imgSize[0])/2))
  h_crop = int(np.floor((height-imgSize[1])/2))
  img = img[w_crop:w_crop+imgSize[0], h_crop:h_crop+imgSize[1], :]
  #Aplicar Rescale nos valores 0~255 >> 0~1
  img[:,:,0] = img[:,:,0]/255.0
  img[:,:,1] = img[:,:,1]/255.0
  img[:,:,2] = img[:,:,2]/255.0
  #Subtrair a média
  img[:,:,0] = img[:,:,0] - np.mean(img[:,:,0])
  img[:,:,1] = img[:,:,1] - np.mean(img[:,:,1])
  img[:,:,2] = img[:,:,2] - np.mean(img[:,:,2])
  #Retorna a imagem pre-processada
  return img
```

As camadas convolucionais de cada rede não foram modificadas, sendo somente a camada totalmente conectada, do topo da rede, substituída. A configuração da camada totalmente

conectada foi construída de forma a maximizar a acurácia nos conjuntos de validação.

Devido ao elevado custo computacional, não foram utilizados as imagens dos diretórios *rndDiv_train*, *rndDiv_valid*, *patLvDiv_train* e *patLvDiv_valid*, somente os diretórios com *data augmentation* foram utilizados para treinamento e validação das redes convolucionais. Ao fim, da mesma forma que os outros classificadores, o modelos gerados tiveram seus desempenhos avaliados com os seus respectivos conjuntos de teste.

Todo código-fonte produzido que é referente a metodologia apresentada neste capítulo está disponível em (ELWYSLAN, 2019). No próximo capítulo serão apresentados os resultados obtidos a partir da aplicação desta metodologia.

5
Resultados

Neste capítulo serão apresentados os resultados obtidos aplicando a metodologia proposta no Capítulo 4. Cada um dos processos de extração de características gerou 10 *dataframes* que foram utilizados para treino, validação e teste dos classificadores, e estão divididos da seguinte forma:

- Dois *dataframes*, um para treino e outro para validação com características extraídas das imagens dos diretórios *rndDiv_train* e *rndDiv_valid* respectivamente, referente as imagens da divisão aleatória.
- Dois *dataframes*, um para treino e outro para validação com características extraídas das imagens dos diretórios *patLvDiv_train* e *patLvDiv_valid* respectivamente, referente as imagens da divisão por paciente.
- Dois *dataframes*, um para treino e outro para validação com características extraídas das imagens dos diretórios *augm_rndDiv_train* e *augm_rndDiv_valid* respectivamente, referente as imagens da divisão aleatória com *data augmentation*.
- Dois *dataframes*, um para treino e outro para validação com características extraídas das imagens dos diretórios *augm_patLvDiv_train* e *augm_patLvDiv_valid* respectivamente, referente as imagens da divisão por paciente com *data augmentation*.
- Dois *dataframes* criados a partir das características extraídas das imagens dos diretórios *rndDiv_test*, previamente balanceado e contendo 678 imagens (339 linfócitos malignos e 339 saudáveis), e *patLvDiv_test*, também previamente balanceado e contendo 364 imagens (182 linfócitos malignos e 182 saudáveis). Estes dois *dataframes* foram usados para a avaliação de desempenho final dos classificadores.

Na seção 5.1 são apresentados os resultados obtidos pelos classificadores que utilizaram como características as estatísticas de 1ª ordem, na seção 5.2 são apresentados os resultados

obtidos com as características de textura, na seção 5.3 são apresentados os resultados obtidos com as características morfológicas, na seção 5.4 são apresentados os resultados obtidos com as características de contorno, na seção 5.5 são apresentados os resultados obtidos com as características da transformada discreta do cosseno e na seção 5.6 são apresentados os resultados obtidos com as redes convolucionais. A seção 5.7 contém o resumo dos resultados obtidos.

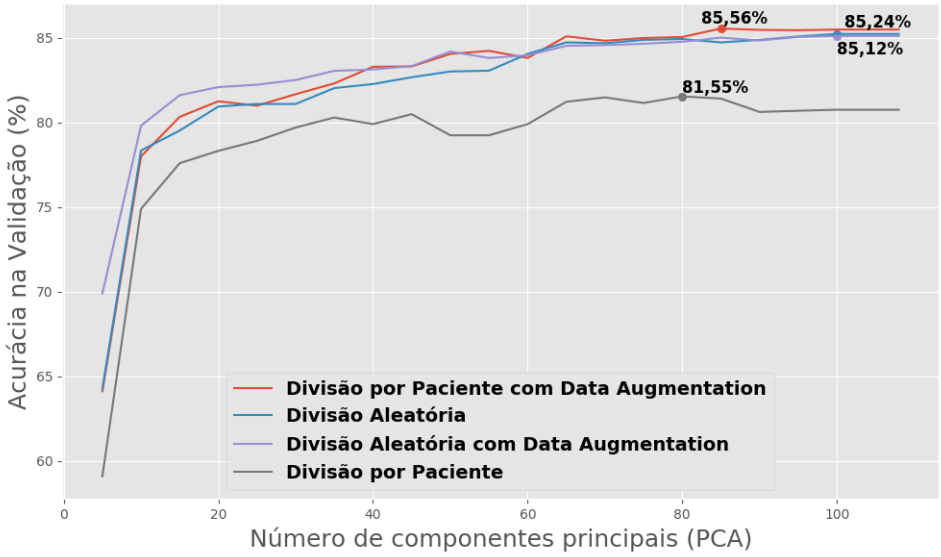
5.1 Classificação dos linfócitos utilizando as estatísticas de 1ª ordem

Os resultados apresentados nesta seção são referentes a classificação dos linfócitos utilizando as estatísticas de 1ª ordem extraídas de suas imagens, as suas subseções estão organizadas da seguinte forma: na subseção 5.1.1 são apresentados os resultados obtidos com o algoritmo do SVM, na subseção 5.1.2 os resultados obtidos com o algoritmo KNN, na subseção 5.1.3 os resultados obtidos com algoritmo RF e, por fim, na subseção 5.1.4 os resultados obtidos com as Redes Neurais.

5.1.1 Classificação utilizando SVM

Para o classificador SVM com *kernel* linear (L-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 40.

Figura 40 – Acurácia do L-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



As maiores acurácias foram: 85,56% com 85 componentes principais utilizando as imagens da divisão por paciente com *data augmentation*, 85,24% com 100 componentes principais utilizando as imagens da divisão aleatória, 85,12% com 100 componentes principais

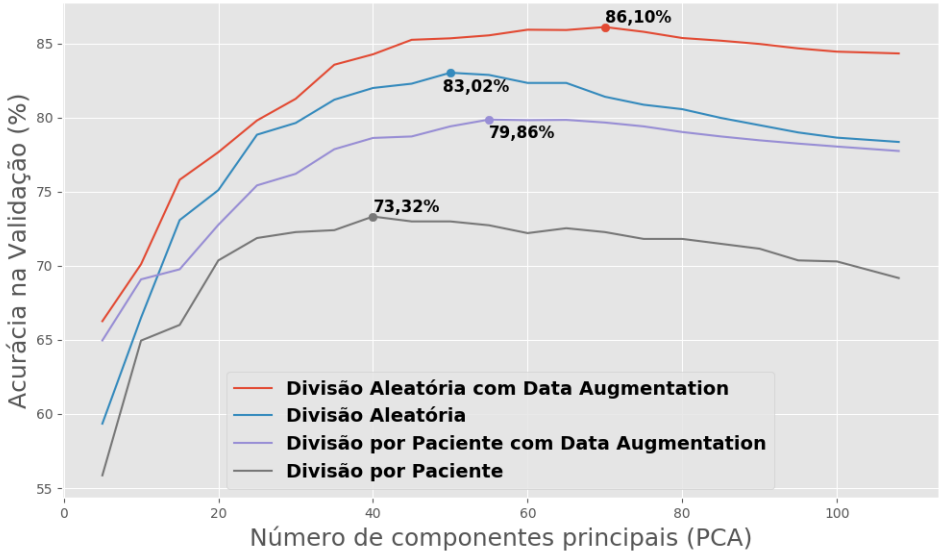
utilizando as imagens da divisão aleatória com *data augmentation* e 81,55% com 80 componentes utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 2.

Tabela 2 – Desempenho dos classificadores L-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	71,98%	66,39%	89,01%	54,95%	76,05%
Divisão por paciente (sem augm.)		70,88%	67,92%	79,12%	62,64%	73,09%
Divisão aleatória (com augm.)	*rndDiv_test*	83,63%	78,50%	92,63%	74,63%	84,98%
Divisão aleatória (sem augm.)		84,22%	82,04%	87,61%	80,83%	84,73%

Para o classificador SVM com *kernel* quadrático (Q-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 41.

Figura 41 – Acurácia do Q-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



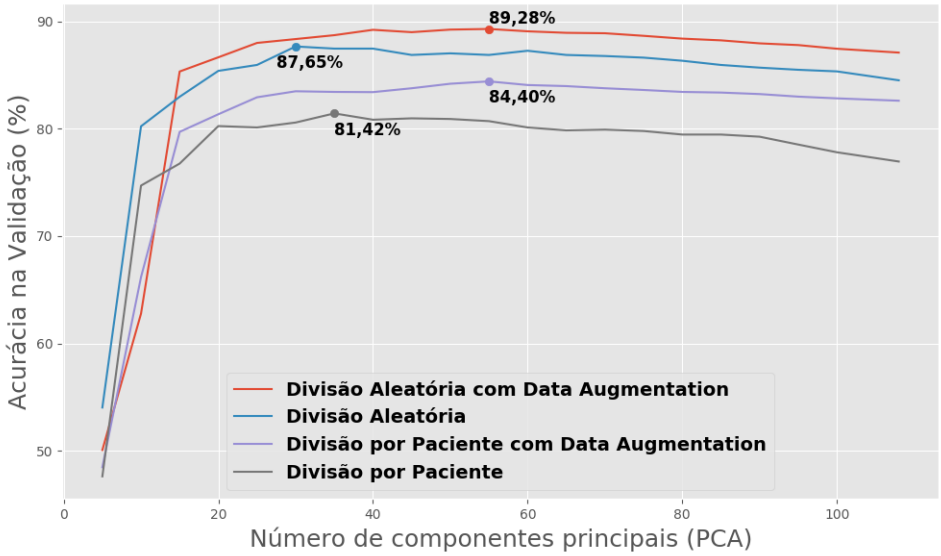
As maiores acurácias foram: 86,10% com 70 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 83,02% com 50 componentes principais utilizando as imagens da divisão aleatória, 79,86% com 55 componentes principais utilizando as imagens da divisão por paciente com *data augmentation* e 73,32% com 40 componentes principais utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 3.

Tabela 3 – Desempenho dos classificadores Q-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	72,25%	69,19%	80,22%	64,29%	74,30%
Divisão por paciente (sem augm.)		66,76%	68,94%	60,99%	72,53%	64,72%
Divisão aleatória (com augm.)	*rndDiv_test*	83,19%	77,64%	93,22%	73,16%	84,72%
Divisão aleatória (sem augm.)		80,68%	78,89%	83,78%	77,58%	81,26%

Para o classificador SVM com *kernel* polinomial (P-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 42.

Figura 42 – Acurácia do P-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



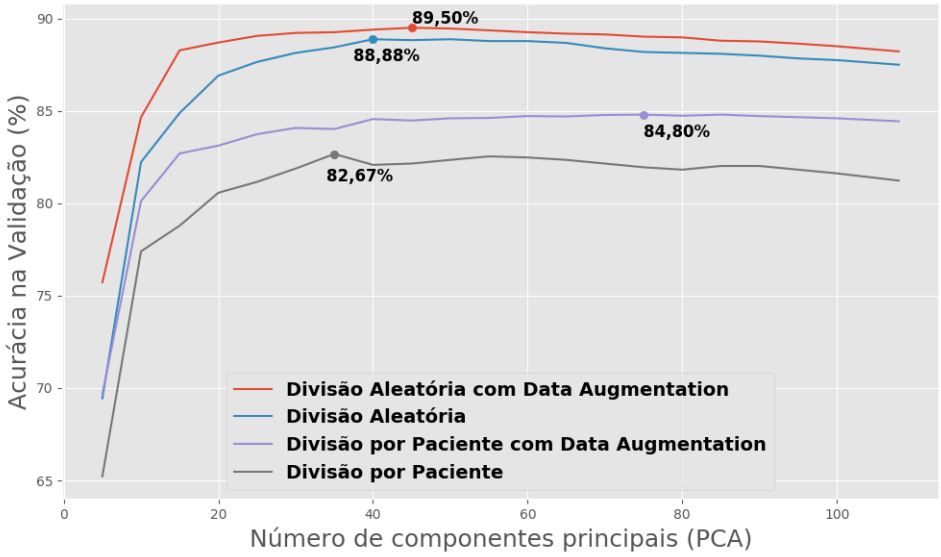
As maiores acurácias foram: 89,28% com 55 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 87,65% com 30 componentes principais utilizando as imagens da divisão aleatória, 84,40% com 55 componentes principais utilizando as imagens da divisão por paciente com *data augmentation* e 81,42% com 35 componentes principais utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 4.

Tabela 4 – Desempenho dos classificadores P-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	75,00%	71,98%	81,87%	68,13%	76,61%
Divisão por paciente (sem augm.)		73,08%	74,14%	70,88%	75,27%	72,47%
Divisão aleatória (com augm.)	*rndDiv_test*	84,51%	79,40%	93,22%	75,81%	85,76%
Divisão aleatória (sem augm.)		83,78%	84,38%	82,89%	84,66%	83,63%

Para o classificador SVM com *kernel* RBF (R-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 43.

Figura 43 – Acurácia do R-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



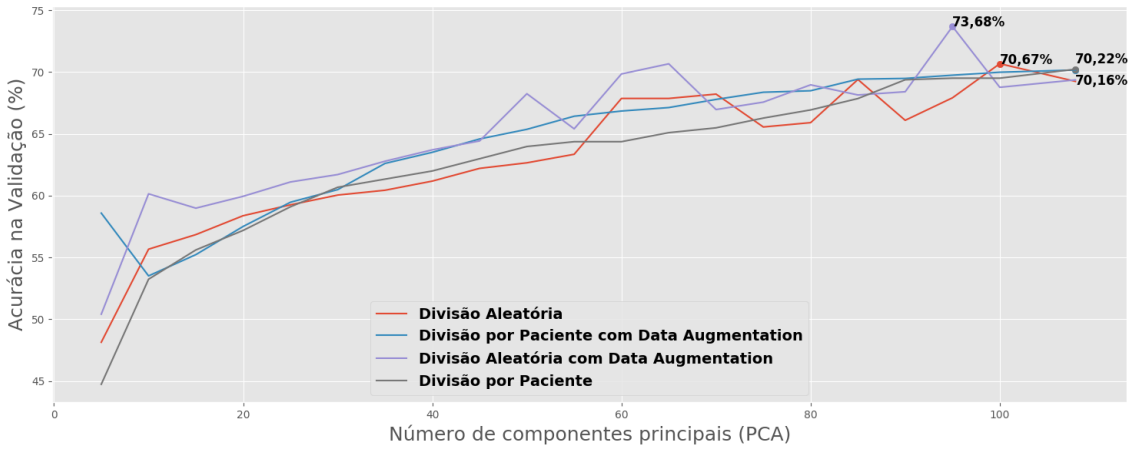
As maiores acurácias foram: 89,50% com 45 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 88,88% com 40 componentes principais utilizando as imagens da divisão aleatória, 84,80% com 75 componentes principais utilizando as imagens da divisão por paciente com *data augmentation* e 82,67% com 35 componentes principais utilizando as imagen da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 5.

Tabela 5 – Desempenho dos classificadores R-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	76,37%	71,05%	89,01%	63,74%	79,02%
Divisão por paciente (sem augm.)		68,41%	69,14%	66,48%	70,33%	67,78%
Divisão aleatória (com augm.)	*rndDiv_test*	87,02%	81,30%	96,17%	77,88%	88,11%
Divisão aleatória (sem augm.)		85,25%	82,92%	88,79%	81,71%	85,75%

Para o classificador SVM com *kernel* sigmoide (Sig-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 44.

Figura 44 – Acurácia do Sig-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



As maiores acurácias foram: 73,68% com 95 componentes principais utilizando as imagem da divisão aleatória com *data augmentation*, 70,67% com 100 componentes principais com as imagens da divisão aleatória, 70,22% sem redução de dimensionalidade com as imagens da divisão por paciente e 70,16% sem redução de dimensionalidade com as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 6.

Tabela 6 – Desempenho dos classificadores Sig-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	65,38%	63,08%	74,18%	56,59%	68,18%
Divisão por paciente (sem augm.)		64,29%	65,29%	60,99%	67,58%	63,07%
Divisão aleatória (com augm.)	*rndDiv_test*	72,57%	70,18%	78,47%	66,67%	74,09%
Divisão aleatória (sem augm.)		69,62%	69,16%	70,80%	68,44%	69,97%

5.1.2 Classificação utilizando KNN

Para o classificador *K-Nearest Neighbors* (KNN) a relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação, utilizando as imagens da divisão por paciente com *data augmentation*, pode ser consultada na Tabela 7.

Tabela 7 – Acurácia do KNN (em %) com as imagens da divisão por paciente com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	68,24	67,56	68,72	67,92	68,86	68,14	69,24	68,62	69,28	68,62	69,06	68,80	69,50	68,86	69,42	69,06
10	77,50	77,06	78,22	76,70	78,26	77,16	78,10	77,14	77,58	76,96	77,54	77,26	77,64	77,06	77,72	77,26
15	78,66	77,14	78,70	**78,20**	78,68	78,16	78,70	77,88	78,58	**78,22**	**78,60**	**78,16**	**78,58**	**78,24**	78,32	**78,10**
20	78,44	77,44	**79,36**	78,12	**79,06**	**78,40**	**78,78**	77,98	**78,82**	77,84	78,18	77,88	78,22	77,72	78,14	77,72
25	78,40	76,92	78,82	77,58	78,90	78,12	78,74	78,00	78,66	77,80	78,38	77,50	78,08	78,04	78,42	77,78
30	78,64	77,50	78,96	77,76	78,92	78,18	78,60	**78,04**	78,44	77,90	78,20	77,94	78,38	78,06	78,40	77,66
35	78,68	77,52	78,88	77,62	78,70	77,88	78,48	77,94	78,52	77,90	78,34	77,88	78,22	77,94	78,50	77,92
40	78,78	77,56	79,02	77,64	78,66	78,04	78,52	77,88	78,52	77,96	78,36	77,86	78,16	78,02	78,50	78,06
45	**78,82**	77,52	79,08	77,80	78,52	77,96	78,52	77,96	78,52	78,00	78,60	78,00	78,08	77,96	78,44	77,94
50	78,78	77,60	79,08	77,84	78,54	78,00	78,46	78,04	78,52	78,00	78,56	78,04	78,16	78,00	78,32	78,02
55	78,74	77,56	79,14	77,76	78,54	78,04	78,54	77,98	78,56	78,04	78,56	78,00	78,24	77,98	78,46	77,96
60	78,80	77,56	79,14	77,80	78,54	78,06	78,58	78,02	78,54	78,04	78,52	77,98	78,26	78,02	**78,52**	78,00
65	78,80	77,58	79,06	77,78	78,60	78,08	78,60	78,00	78,54	78,06	78,54	77,96	78,26	78,06	78,50	78,02
70	78,82	**77,64**	79,08	77,74	78,58	78,06	78,64	77,96	78,54	78,08	78,58	78,02	78,26	78,06	78,46	78,00
75	78,82	77,60	79,10	77,74	78,58	78,08	78,62	77,98	78,54	78,06	78,58	78,00	78,28	78,06	78,50	78,02
80	78,80	77,62	79,10	77,72	78,54	78,08	78,60	77,98	78,56	78,02	78,52	78,06	78,28	78,04	78,50	78,00
85	78,82	77,62	79,08	77,76	78,54	78,08	78,60	77,98	78,54	78,00	78,50	78,06	78,26	78,04	78,50	78,02
90	78,82	77,64	79,08	77,76	78,56	78,08	78,60	77,98	78,54	78,02	78,50	78,06	78,26	78,02	78,50	78,02
95	78,82	77,64	79,08	77,76	78,54	78,08	78,58	77,98	78,54	78,02	78,52	78,06	78,26	78,02	78,50	78,02
100	78,82	77,64	79,08	77,76	78,54	78,08	78,58	77,98	78,54	78,02	78,52	78,06	78,26	78,02	78,50	78,02
108	78,82	77,64	79,08	77,76	78,54	78,08	78,58	77,98	78,54	78,02	78,52	78,06	78,26	78,02	78,50	78,02

A maior acurácia foi 79,36% e ocorreu com 20 componentes principais e $k = 7$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão da divisão aleatória com *data augmentation* pode ser consultada na Tabela 8.

Tabela 8 – Acurácia do KNN (em %) com as imagens da divisão aleatória com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	71,66	72,30	72,16	72,70	72,60	73,08	72,98	73,40	73,34	73,60	73,58	74,28	73,96	74,34	74,32	74,72
10	82,36	82,62	83,00	83,08	82,90	82,70	82,92	82,80	83,08	82,94	83,10	83,26	83,00	83,34	83,38	83,38
15	85,66	85,50	85,24	85,72	85,12	85,58	85,74	86,04	85,72	85,80	85,48	85,68	85,36	85,32	85,32	85,38
20	86,14	85,90	86,04	86,32	85,82	86,40	86,32	**86,68**	86,08	86,12	85,80	85,94	85,64	85,84	85,66	85,84
25	86,02	86,16	86,10	86,46	85,98	86,50	86,24	86,38	86,22	86,28	85,96	86,20	86,00	86,14	85,60	86,08
30	86,22	86,52	86,10	86,48	85,98	86,54	86,06	86,16	86,14	86,26	86,10	**86,46**	85,98	86,20	85,78	**86,10**
35	86,32	86,46	86,06	86,50	86,12	86,48	86,18	86,26	86,24	86,20	86,08	86,30	85,92	86,10	85,80	86,02
40	86,42	86,58	86,18	86,46	86,22	**86,62**	**86,36**	86,24	86,28	**86,32**	86,10	86,40	85,92	86,16	85,88	86,02
45	86,42	86,56	86,30	86,48	86,26	86,58	86,30	86,32	86,18	86,28	86,08	86,46	**86,08**	86,22	85,94	86,10
50	86,46	86,54	**86,40**	86,60	**86,28**	86,56	86,28	86,32	86,28	86,32	86,18	86,44	86,00	**86,26**	**85,98**	86,06
55	**86,48**	86,60	86,40	86,62	86,26	86,58	86,24	86,38	**86,30**	86,28	86,18	86,44	85,98	86,24	85,92	86,00
60	86,40	86,68	86,40	**86,64**	86,24	86,58	86,24	86,42	86,20	86,30	86,24	86,46	86,04	86,18	85,98	85,98
65	86,42	86,64	86,38	86,64	86,24	86,58	86,28	86,46	86,28	86,30	86,24	86,46	85,98	86,20	85,96	86,02
70	86,40	86,64	86,38	86,64	86,24	86,62	86,30	86,44	86,28	86,32	86,24	86,40	86,00	86,22	85,94	85,98
75	86,44	86,68	86,40	86,64	86,26	86,58	86,26	86,46	86,20	86,30	86,24	86,42	86,00	86,22	85,98	86,00
80	86,44	86,68	86,40	86,62	86,26	86,62	86,26	86,46	86,24	86,32	86,24	86,40	85,98	86,22	85,94	85,98
85	86,46	**86,70**	86,40	86,62	86,28	86,62	86,26	86,46	86,24	86,30	**86,26**	86,36	86,00	86,22	85,96	85,98
90	86,46	86,70	86,38	86,62	86,28	86,62	86,28	86,46	86,24	86,30	86,26	86,32	85,98	86,22	85,96	85,98
95	86,46	86,70	86,38	86,62	86,28	86,62	86,28	86,46	86,26	86,30	86,26	86,34	86,00	86,22	85,96	85,98
100	86,46	86,70	86,38	86,62	86,28	86,62	86,28	86,46	86,26	86,30	86,26	86,34	86,00	86,22	85,96	85,98
108	86,46	86,70	86,38	86,62	86,28	86,62	86,28	86,46	86,26	86,30	86,26	86,34	86,00	86,22	85,96	85,98

A maior acurácia foi 86,70% e ocorreu com 85 componentes principais e $k = 6$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão por paciente pode ser consultada na Tabela 9.

Tabela 9 – Acurácia do KNN (em %) com as imagens da divisão por paciente

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	64,10	61,79	63,83	62,45	64,03	63,18	63,64	62,45	63,64	63,11	64,49	62,65	63,57	62,98	63,77	62,32
10	72,86	70,36	72,86	71,01	**72,79**	71,87	72,60	71,61	71,94	71,67	71,87	71,48	**72,60**	71,54	72,07	71,34
15	72,60	70,49	72,66	71,08	72,53	71,01	72,60	71,81	72,66	71,94	72,60	71,48	71,67	71,01	71,74	71,61
20	**73,19**	**70,75**	72,40	**71,34**	72,53	71,28	72,60	71,21	73,06	72,60	72,60	71,61	72,20	71,54	72,33	**71,94**
25	72,99	70,75	**72,92**	71,34	72,60	**72,00**	**73,32**	72,20	**73,72**	**72,66**	72,66	72,07	72,33	71,74	**72,66**	71,94
30	72,92	70,09	72,40	70,62	72,40	71,94	73,06	71,74	73,32	72,66	72,73	**72,13**	72,40	71,94	72,33	71,67
35	72,92	70,16	72,27	70,42	72,60	71,87	73,12	71,81	72,99	72,66	72,79	72,07	72,40	**72,07**	72,60	71,81
40	73,06	70,03	72,00	70,69	72,60	71,61	73,12	72,00	72,86	72,20	**72,92**	72,13	72,20	72,00	72,40	71,81
45	72,86	70,03	72,20	70,42	72,33	71,81	72,99	**72,33**	72,92	72,07	72,66	72,07	72,40	72,00	72,46	71,67
50	72,79	70,09	72,20	70,49	72,66	71,74	72,99	72,27	72,92	72,20	72,79	72,00	72,40	71,87	72,40	71,87
55	72,79	70,22	72,20	70,49	72,40	71,81	72,79	72,20	72,92	72,13	72,86	71,87	72,33	71,87	72,33	71,81
60	72,73	70,09	72,00	70,62	72,46	71,61	72,86	72,20	72,92	72,20	72,79	71,81	72,27	71,94	72,33	71,87
65	72,86	70,16	71,94	70,62	72,46	71,67	72,92	72,20	72,92	72,20	72,86	71,94	72,27	71,94	72,33	71,87
70	72,86	70,22	72,07	70,62	72,46	71,67	72,92	72,20	72,99	72,27	72,86	71,94	72,33	71,94	72,33	71,87
75	72,86	70,29	72,07	70,62	72,33	71,67	72,92	72,13	72,99	72,27	72,86	71,87	72,27	71,87	72,33	71,87
80	72,86	70,29	72,07	70,62	72,27	71,67	72,92	72,20	72,92	72,27	72,86	71,94	72,33	71,87	72,33	71,94
85	72,86	70,29	72,13	70,55	72,33	71,67	72,92	72,20	72,92	72,27	72,86	71,94	72,33	71,87	72,33	71,94
90	72,86	70,36	72,13	70,55	72,33	71,67	72,92	72,20	72,92	72,27	72,86	71,94	72,33	71,87	72,33	71,94
95	72,86	70,36	72,07	70,55	72,40	71,67	72,92	72,20	72,92	72,27	72,86	71,94	72,33	71,87	72,33	71,94
100	72,86	70,36	72,07	70,55	72,40	71,67	72,92	72,20	72,92	72,27	72,86	71,94	72,33	71,87	72,33	71,94
108	72,86	70,36	72,07	70,55	72,40	71,67	72,92	72,20	72,92	72,27	72,86	71,94	72,33	71,87	72,33	71,94

A maior acurácia foi 73,72% e ocorreu com 25 componentes principais e $k = 13$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão aleatória pode ser consultada na Tabela 10.

Tabela 10 – Acurácia do KNN (em %) com as imagens da divisão aleatória

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	67,22	66,44	68,11	67,37	68,55	68,55	69,73	68,90	68,95	69,14	69,24	69,05	69,83	68,45	69,34	69,49
10	78,79	78,69	78,94	78,79	79,23	78,94	79,43	79,08	79,53	79,38	80,07	78,84	79,53	79,28	79,92	79,63
15	80,36	80,76	82,14	81,30	82,23	81,10	81,35	80,86	81,55	81,00	81,59	80,71	81,25	80,22	80,81	80,71
20	82,58	81,64	82,33	81,25	82,04	81,84	82,73	82,04	82,53	81,69	82,48	81,74	82,38	81,59	82,28	**82,04**
25	82,58	81,30	82,48	81,55	**82,68**	81,74	82,87	82,33	**82,87**	81,99	82,97	82,23	82,53	81,84	82,38	81,79
30	82,63	81,50	83,02	81,40	82,33	81,55	82,82	82,38	82,87	82,19	**83,02**	**82,58**	82,58	**82,58**	82,28	81,64
35	**82,78**	81,79	82,92	**81,69**	82,43	81,50	82,87	82,23	82,78	81,94	82,78	82,09	**82,78**	82,43	82,43	81,59
40	82,58	81,79	82,97	81,59	82,48	81,64	82,87	82,38	82,73	82,09	82,87	82,14	82,43	82,28	82,38	81,74
45	82,68	81,69	83,07	81,50	82,53	81,84	83,02	**82,53**	82,63	82,14	82,92	82,23	82,53	82,33	**82,48**	81,84
50	82,73	81,84	83,02	81,35	82,48	81,74	83,07	82,43	82,68	82,23	82,97	82,33	82,53	82,23	82,48	81,89
55	82,63	81,79	83,07	81,55	82,48	81,69	83,07	82,38	82,68	**82,28**	82,87	82,33	82,73	82,33	82,48	81,84
60	82,58	**81,89**	83,12	81,40	82,48	81,74	83,07	82,48	82,78	82,28	82,92	82,33	82,68	82,28	82,48	81,84
65	82,68	81,84	83,17	81,40	82,38	81,79	**83,12**	82,43	82,78	82,23	82,97	82,33	82,58	82,33	82,48	81,84
70	82,68	81,84	**83,22**	81,45	82,38	81,84	83,07	82,38	82,78	82,23	82,97	82,28	82,58	82,28	82,38	81,84
75	82,73	81,84	83,22	81,45	82,43	81,84	83,07	82,33	82,78	82,23	82,97	82,38	82,63	82,33	82,48	81,94
80	82,73	81,89	83,22	81,45	82,43	**81,89**	83,02	82,38	82,78	82,23	82,97	82,38	82,63	82,33	82,48	81,89
85	82,73	81,89	83,22	81,45	82,43	81,89	83,02	82,33	82,78	82,23	82,97	82,38	82,63	82,33	82,48	81,89
90	82,73	81,89	83,22	81,45	82,43	81,89	83,02	82,33	82,78	82,23	82,97	82,38	82,63	82,33	82,48	81,89
95	82,73	81,89	83,22	81,45	82,43	81,89	83,02	82,33	82,78	82,23	82,97	82,38	82,63	82,33	82,48	81,89
100	82,73	81,89	83,22	81,45	82,43	81,89	83,02	82,33	82,78	82,23	82,97	82,38	82,63	82,33	82,48	81,89
108	82,73	81,89	83,22	81,45	82,43	81,89	83,02	82,33	82,78	82,23	82,97	82,38	82,63	82,33	82,48	81,89

A maior acurácia foi 83,22% e ocorreu com 70 componentes principais e $k = 7$.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens do conjuntos de teste, obtêm-se os resultados da Tabela 11.

Tabela 11 – Desempenho dos classificadores KNN nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	69,78%	66,67%	79,12%	60,44%	72,36%
Divisão por paciente (sem augm.)		62,36%	65,31%	52,75%	71,98%	58,36%
Divisão aleatória (com augm.)	*rndDiv_test*	82,74%	78,32%	90,56%	74,93%	84,00%
Divisão aleatória (sem augm.)		78,76%	77,94%	80,24%	77,29%	79,07%

5.1.3 Classificação utilizando Floresta Aleatória

Para o classificador Floresta Aleatória (RF) a relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente com *data augmentation* pode ser consultada na Tabela 12.

Tabela 12 – Acurácia do RF com as imagens da divisão por paciente com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	69,60%	69,80%	69,50%	70,34%	69,84%	69,92%
10	80,22%	80,66%	80,52%	80,46%	80,38%	80,18%
15	81,08%	81,20%	81,40%	81,36%	81,54%	81,46%
20	81,64%	81,72%	81,48%	81,34%	82,00%	81,86%
25	80,76%	81,52%	81,38%	81,72%	81,16%	81,64%
30	81,56%	81,38%	81,72%	81,46%	81,80%	80,86%
35	80,84%	81,56%	81,00%	81,32%	81,42%	81,46%
40	81,66%	81,48%	82,04%	81,74%	81,66%	81,76%
45	81,32%	81,32%	81,82%	81,24%	81,80%	81,50%
50	81,66%	81,70%	81,82%	82,54%	81,88%	82,22%
55	82,20%	82,40%	82,78%	83,00%	82,24%	82,54%
60	81,74%	82,20%	82,08%	82,34%	82,96%	82,74%
65	82,32%	82,32%	82,56%	82,72%	82,52%	82,86%
70	81,46%	82,06%	83,02%	82,28%	82,68%	82,98%
75	82,04%	82,52%	**83,30**%	83,00%	83,22%	**83,60**%
80	**83,08**%	82,02%	82,88%	**83,20**%	**83,28**%	83,26%
85	82,36%	82,44%	83,10%	82,70%	82,24%	83,44%
90	82,00%	**82,76**%	82,26%	82,60%	83,00%	83,10%
95	81,94%	82,48%	82,36%	82,82%	82,56%	83,06%
100	82,78%	82,30%	82,44%	82,60%	82,74%	83,36%
108	81,54%	82,76%	82,28%	82,68%	82,90%	82,82%

A maior acurácia foi 83,60% e ocorreu com 75 componentes principais e 150 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória com *data augmentation* pode ser consultada na Tabela 13.

Tabela 13 – Acurácia do RF com as imagens da divisão aleatória com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	74,48%	74,46%	74,08%	74,62%	74,34%	74,36%
10	83,76%	84,24%	84,54%	84,32%	84,36%	84,34%
15	86,02%	86,84%	86,42%	86,10%	86,56%	86,10%
20	86,80%	86,72%	86,60%	86,84%	86,32%	86,82%
25	86,36%	86,48%	86,90%	86,60%	87,12%	86,76%
30	86,76%	86,60%	86,72%	86,68%	86,96%	86,86%
35	86,46%	86,98%	86,70%	86,88%	86,94%	86,98%
40	86,94%	87,42%	87,42%	87,44%	87,16%	87,46%
45	86,86%	87,08%	87,20%	87,66%	87,64%	87,60%
50	87,22%	87,28%	**87,78**%	**87,86**%	**87,86**%	87,16%
55	87,42%	87,44%	87,12%	87,38%	87,74%	87,58%
60	87,62%	86,72%	87,12%	87,08%	87,58%	87,48%
65	86,90%	86,86%	87,00%	87,72%	87,38%	87,32%
70	87,28%	86,96%	87,30%	87,34%	87,72%	87,44%
75	**87,76**%	87,02%	86,86%	87,42%	87,50%	87,68%
80	86,50%	86,96%	87,56%	87,16%	87,26%	87,72%
85	87,12%	86,72%	87,58%	87,64%	87,48%	87,62%
90	86,56%	86,76%	87,44%	87,44%	87,26%	87,46%
95	87,68%	86,98%	87,52%	87,82%	87,44%	87,26%
100	86,78%	**87,60**%	87,08%	87,42%	87,52%	**87,88**%
108	86,90%	87,14%	87,28%	87,62%	87,66%	87,54%

A maior acurácia foi 87,88% e ocorreu com 100 componentes principais e 150 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente pode ser consultada na Tabela 14.

Tabela 14 – Acurácia do RF com as imagens da divisão por paciente

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	66,47%	67,19%	66,47%	66,14%	65,61%	66,60%
10	77,67%	78,72%	78,33%	78,52%	78,92%	78,26%
15	77,54%	78,13%	78,19%	78,39%	79,05%	78,66%
20	79,31%	78,72%	79,38%	79,25%	78,85%	79,78%
25	78,19%	78,26%	**80,30**%	**80,63**%	80,24%	79,51%
30	**79,71**%	**79,71**%	80,11%	79,58%	79,12%	79,84%
35	78,99%	78,33%	80,24%	79,45%	79,58%	**80,30**%
40	78,06%	78,00%	79,58%	78,92%	80,24%	80,30%
45	79,31%	78,13%	77,80%	78,85%	79,45%	79,91%
50	76,42%	78,39%	79,25%	77,87%	79,31%	79,25%
55	78,06%	78,59%	79,51%	78,92%	79,38%	79,38%
60	78,13%	79,25%	79,84%	77,80%	79,18%	78,72%
65	79,64%	79,25%	79,64%	79,71%	79,25%	79,25%
70	78,52%	79,31%	79,05%	80,50%	79,64%	79,25%
75	77,93%	78,00%	79,38%	78,99%	78,06%	79,18%
80	78,99%	77,93%	79,84%	78,72%	79,64%	80,11%
85	77,87%	78,85%	78,99%	78,79%	**80,43**%	79,38%
90	76,68%	77,27%	79,64%	79,25%	78,06%	79,91%
95	78,06%	79,05%	78,00%	79,18%	79,05%	78,79%
100	78,46%	79,31%	79,38%	79,91%	79,64%	79,78%
108	77,47%	77,54%	78,72%	78,46%	79,84%	79,05%

A maior acurácia foi 80,63% e ocorreu com 25 componentes principais e 110 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória pode ser consultada na Tabela 15.

Tabela 15 – Acurácia do RF com as imagens da divisão aleatória

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	68,95%	69,00%	70,13%	69,78%	69,64%	69,78%
10	83,86%	83,71%	83,86%	83,81%	83,91%	83,61%
15	85,97%	86,07%	85,73%	85,63%	86,42%	85,48%
20	87,06%	87,01%	86,96%	87,20%	87,25%	87,06%
25	86,96%	**87,75**%	**88,19**%	**87,89**%	**88,09**%	87,75%
30	87,25%	87,65%	87,35%	87,55%	87,50%	87,75%
35	**87,55**%	86,96%	87,70%	87,06%	87,94%	87,89%
40	87,01%	87,06%	86,96%	87,55%	87,25%	87,65%
45	86,66%	87,20%	86,96%	87,65%	87,30%	86,52%
50	86,22%	86,76%	87,84%	87,16%	87,06%	87,40%
55	86,12%	86,91%	87,35%	87,16%	87,65%	87,50%
60	86,61%	86,96%	86,61%	87,06%	87,65%	87,50%
65	86,07%	86,47%	86,61%	87,70%	87,60%	86,96%
70	86,56%	86,76%	86,32%	87,89%	87,16%	87,40%
75	87,20%	87,30%	86,61%	87,65%	87,16%	**88,14**%
80	84,84%	86,71%	86,47%	87,70%	87,45%	87,01%
85	84,40%	87,25%	86,96%	87,70%	86,66%	86,66%
90	86,07%	86,32%	86,96%	87,89%	86,76%	86,81%
95	85,29%	86,91%	87,30%	87,11%	87,16%	87,60%
100	86,32%	86,02%	87,16%	86,61%	87,30%	87,55%
108	86,42%	86,42%	87,25%	87,11%	87,11%	87,55%

A maior acurácia foi 88,19% e ocorreu com 25 componentes principais e 90 árvores aleatórias.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 16.

Tabela 16 – Desempenho dos classificadores RF nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	72,80%	70,65%	78,02%	67,58%	74,15%
Divisão por paciente (sem augm.)		71,98%	74,10%	67,58%	76,37%	70,69%
Divisão aleatória (com augm.)	*rndDiv_test*	84,66%	78,31%	95,87%	73,45%	86,20%
Divisão aleatória (sem augm.)		84,22%	81,69%	88,20%	80,24%	84,82%

5.1.4 Classificação utilizando Redes Neurais

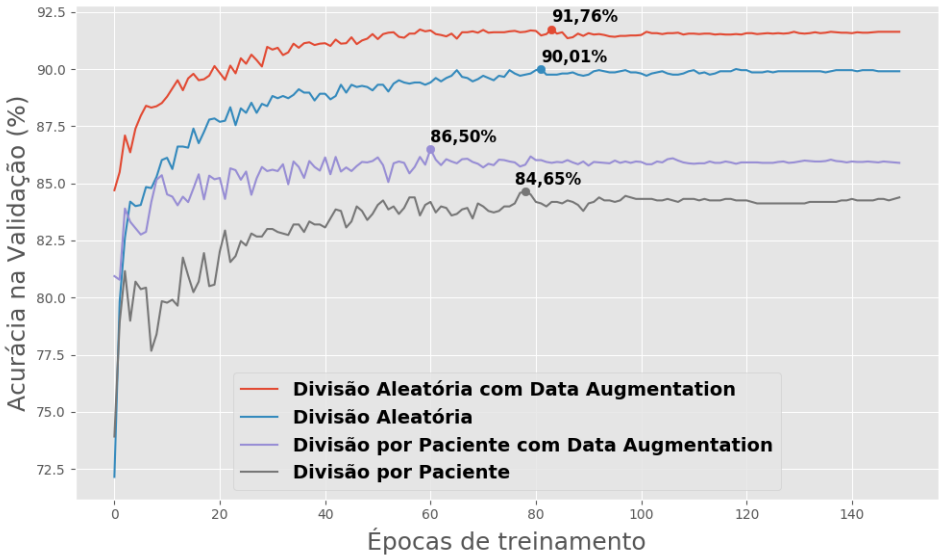
Das arquiteturas de Redes Neurais avaliadas, as que obtiveram as maiores acurácias foram:

- 2 camadas, cada qual com 1024 neurônios, que utilizam tangente hiperbólica como função de ativação, para as imagens da divisão por Paciente com *data augmentation*;
- 3 camadas, cada qual com 1024, neurônios que utilizam ReLU como função de ativação, Para as imagens da divisão aleatória com *data augmentation*;

- 3 camadas, cada qual com 1024, neurônios que utilizam tangente hiperbólica como função de ativação, para as imagens da divisão por paciente;
- 2 camadas, cada qual com 1024, neurônios que utilizam ReLU como função de ativação, para as imagens da divisão aleatória;

A acurácia na validação em função do número de épocas de treinamento de cada uma destas redes pode ser vista na Figura 45.

Figura 45 – Acurácia das Redes Neurais dos *dataframes* de validação em função das épocas de treinamento



As maiores acurácias foram: 91,76% com as imagens da divisão aleatória com *data augmentation*, 90,01% com as imagens da divisão aleatória, 86,50% com as imagens da divisão por paciente com *data augmentation* e 84,65% com as imagens da divisão por paciente.

Aplicando estas redes neurais, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 17.

Tabela 17 – Desempenho das Redes Neurais nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	81,04%	76,78%	89,01%	73,08%	82,44%
Divisão por paciente (sem augm.)		77,47%	77,78%	76,92%	78,02%	77,35%
Divisão aleatória (com augm.)	*rndDiv_test*	89,53%	85,83%	94,69%	84,37%	90,04%
Divisão aleatória (sem augm.)		89,09%	87,54%	91,15%	87,02%	89,31%

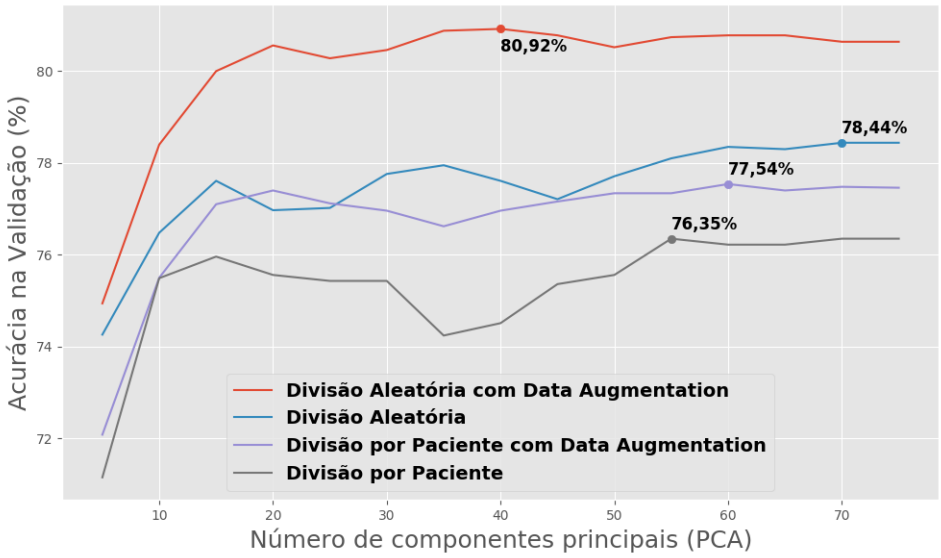
5.2 Classificação dos linfócitos utilizando as características de textura

Os resultados apresentados nesta seção são referentes a classificação dos linfócitos utilizando as características de textura extraídas de suas imagens, as suas subseções estão organizadas da seguinte forma: na subseção 5.2.1 são apresentados os resultados obtidos com o algoritmo SVM, na subseção 5.2.2 os resultados obtidos com o algoritmo KNN, na subseção 5.2.3 os resultados obtidos com algoritmo RF e, por fim, na subseção 5.2.4 os resultados obtidos com as Redes Neurais.

5.2.1 Classificação utilizando SVM

Para o classificador SVM com *kernel* linear (L-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 46.

Figura 46 – Acurácia do L-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



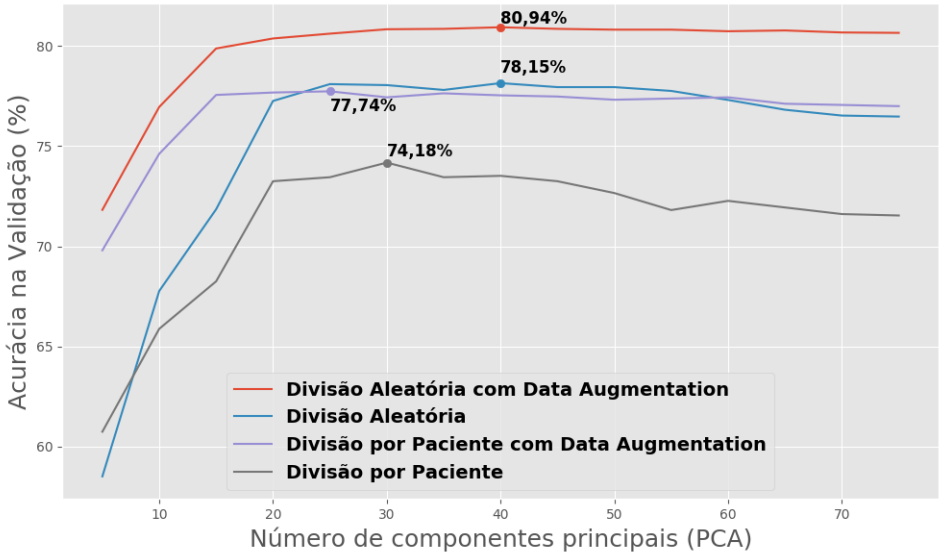
As maiores acurácias foram: 80,92% com 40 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 78,44% com 70 componentes principais utilizando as imagens da divisão aleatória, 77,54% com 60 componentes principais utilizando as imagens da divisão por paciente com *data augmentation* e 76,35% com 55 componentes principais utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 18.

Tabela 18 – Desempenho dos classificadores L-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	64,01%	60,49%	80,77%	47,25%	69,17%
Divisão por paciente (sem augm.)		61,81%	60,70%	67,03%	56,59%	63,71%
Divisão aleatória (com augm.)	*rndDiv_test*	76,11%	71,53%	86,73%	65,49%	78,40%
Divisão aleatória (sem augm.)		75,81%	75,07%	77,29%	74,34%	76,16%

Para o classificador SVM com *kernel* quadrático (Q-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 47.

Figura 47 – Acurácia do Q-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



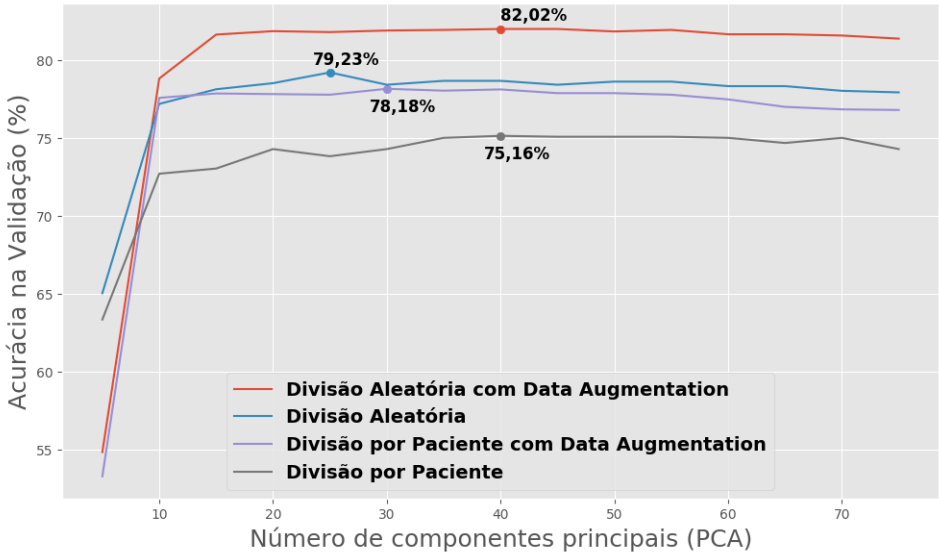
As maiores acurácias foram: 80,94% com 40 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 78,15% com 40 componentes principais utilizando as imagens da divisão aleatória, 77,74% com 25 componentes principais utilizando as imagens da divisão por paciente com *data augmentation* e 74,18% com 40 componentes principais utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 19.

Tabela 19 – Desempenho dos classificadores Q-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	62,91%	59,51%	80,77%	45,05%	68,53%
Divisão por paciente (sem augm.)		62,64%	61,06%	69,78%	55,49%	65,13%
Divisão aleatória (com augm.)	*rndDiv_test*	75,81%	70,30%	89,38%	62,24%	78,70%
Divisão aleatória (sem augm.)		75,66%	74,72%	77,58%	73,75%	76,12%

Para o classificador SVM com *kernel* polinomial (P-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 48.

Figura 48 – Acurácia do P-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



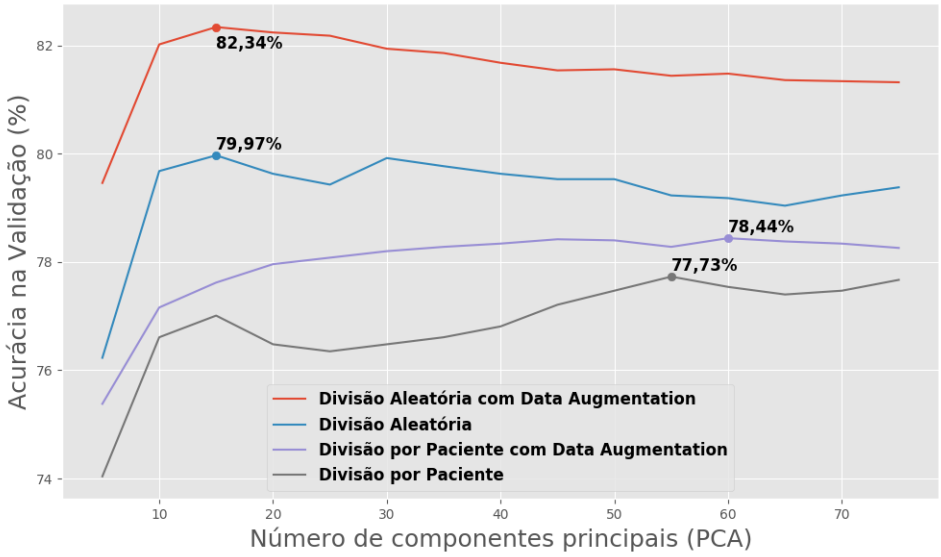
As maiores acurácias foram: 82,02% com 40 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 79,23% com 25 componentes principais utilizando as imagens da divisão aleatória, 78,18% com 30 componentes principais utilizando as imagens da divisão por paciente com *data augmentation* e 75,16% com 40 componentes principais utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de testee, obtêm-se os resultados da Tabela 20.

Tabela 20 – Desempenho dos classificadores P-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	65,11%	61,22%	82,42%	47,80%	70,26%
Divisão por paciente (sem augm.)		60,99%	60,42%	63,74%	58,24%	62,04%
Divisão aleatória (com augm.)	*rndDiv_test*	76,70%	71,19%	89,68%	63,72%	79,37%
Divisão aleatória (sem augm.)		75,22%	74,08%	77,58%	72,86%	75,79%

Para o classificador SVM com *kernel* RBF (R-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 49.

Figura 49 – Acurácia do R-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



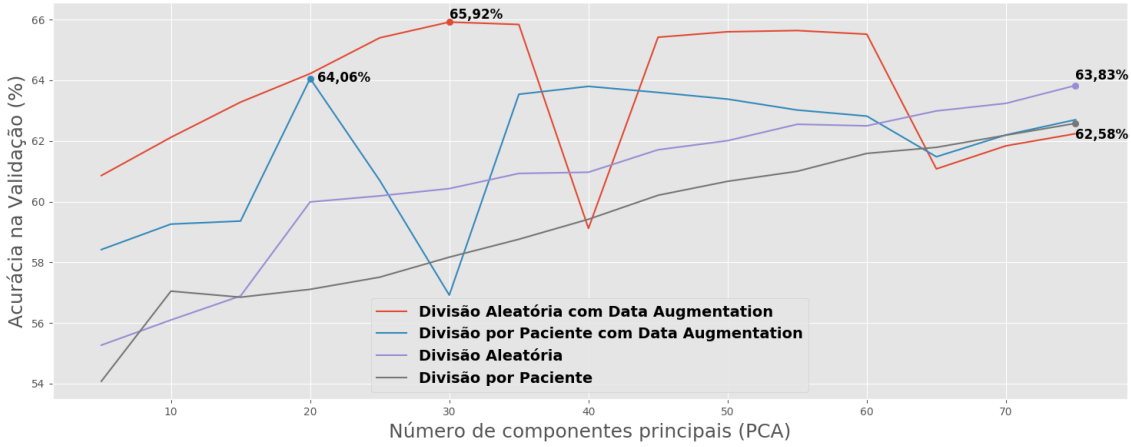
As maiores acurácias foram: 82,34% com 15 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 79,97% com 15 componentes principais utilizando as imagens da divisão aleatória, 78,44% com 60 componentes principais utilizando as imagens da divisão por paciente com *data augmentation* e 77,73% com 55 componentes principais utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 21.

Tabela 21 – Desempenho dos classificadores R-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	62,09%	58,80%	80,77%	43,41%	68,06%
Divisão por paciente (sem augm.)		60,71%	59,90%	64,84%	56,59%	62,27%
Divisão aleatória (com augm.)	*rndDiv_test*	75,96%	70,28%	89,97%	61,95%	78,92%
Divisão aleatória (sem augm.)		76,55%	73,94%	82,01%	71,09%	77,77%

Para o classificador SVM com *kernel* sigmoide (Sig-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 50.

Figura 50 – Acurácia do Sig-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



As maiores acurácias foram: 65,92% com 30 componentes principais utilizando as imagem da divisão aleatória com *data augmentation*, 64,06% com 20 componentes principais utilizando as imagens da divisão por paciente com *data augmentation*, 63,83% sem redução de dimensionalidade utilizando as imagens da divisão aleatória e 62,58% sem redução de dimensionalidade com as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 22.

Tabela 22 – Desempenho dos classificadores Sig-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	55,22%	54,87%	58,79%	51,65%	56,76%
Divisão por paciente (sem augm.)		56,04%	56,63%	51,65%	60,44%	54,03%
Divisão aleatória (com augm.)	*rndDiv_test*	63,86%	61,19%	75,81%	51,92%	67,72%
Divisão aleatória (sem augm.)		64,16%	65,09%	61,06%	67,26%	63,01%

5.2.2 Classificação utilizando KNN

Para o classificador *K-Nearest Neighbors* (KNN) a relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação, utilizando as imagens da divisão por paciente com *data augmentation*, pode ser consultada na Tabela 23.

Tabela 23 – Acurácia do KNN (em %) com as imagens da divisão por paciente com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	73,80	72,50	74,42	73,78	74,86	74,00	74,84	74,68	75,16	74,76	74,94	74,74	74,80	74,90	75,00	75,04
10	75,68	74,44	75,64	75,00	75,88	**75,52**	75,98	75,42	76,04	75,78	76,14	75,82	76,02	75,78	75,98	75,92
15	76,32	74,96	76,24	75,46	76,08	75,38	**76,54**	**75,76**	76,04	75,90	76,28	76,22	76,38	76,50	76,44	76,34
20	76,22	74,88	76,54	75,60	76,30	75,48	76,18	75,56	76,56	75,96	**76,74**	76,26	76,30	76,32	76,40	76,48
25	76,08	75,14	76,78	75,66	76,32	75,44	76,24	75,58	76,62	76,08	76,60	76,38	76,74	76,62	76,30	76,38
30	76,20	75,16	76,98	75,68	76,30	75,34	76,30	75,64	76,64	76,08	76,64	**76,56**	**76,78**	**76,70**	**76,52**	76,54
35	76,30	75,14	**77,02**	75,78	76,38	75,36	76,34	75,68	**76,72**	76,08	76,50	76,32	76,54	76,58	76,52	**76,62**
40	76,36	75,28	76,94	75,84	76,34	75,32	76,38	75,68	76,62	76,10	76,62	76,32	76,48	76,50	76,46	76,58
45	76,36	**75,30**	76,94	75,84	76,34	75,32	76,40	75,62	76,66	76,14	76,60	76,34	76,50	76,46	76,48	76,48
50	76,36	75,20	76,88	**75,86**	**76,40**	75,28	76,44	75,62	76,72	76,14	76,62	76,30	76,52	76,46	76,46	76,50
55	76,36	75,22	76,92	75,86	76,38	75,28	76,42	75,66	76,68	76,14	76,60	76,30	76,54	76,46	76,48	76,48
60	**76,38**	75,22	76,88	75,84	76,38	75,28	76,42	75,66	76,68	76,14	76,60	76,32	76,52	76,46	76,50	76,48
65	76,38	75,20	76,88	75,84	76,38	75,28	76,42	75,68	76,68	**76,16**	76,60	76,30	76,52	76,46	76,50	76,48
70	76,38	75,20	76,88	75,84	76,38	75,28	76,42	75,68	76,68	76,16	76,60	76,30	76,52	76,46	76,50	76,48
75	76,38	75,20	76,88	75,84	76,38	75,28	76,42	75,68	76,68	76,16	76,60	76,30	76,52	76,46	76,50	76,48

A maior acurácia foi 77,02% e ocorreu com 35 componentes principais e $k = 7$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão da divisão aleatória com *data augmentation* pode ser consultada na Tabela 24.

Tabela 24 – Acurácia do KNN (em %) com as imagens da divisão aleatória com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	77,46	76,98	77,94	77,42	77,98	78,46	78,26	78,44	78,42	78,46	78,64	78,50	78,68	78,76	78,82	78,78
10	79,46	80,16	80,76	**80,98**	80,62	**81,12**	80,62	80,94	80,68	80,96	80,78	80,76	80,96	81,12	81,04	81,02
15	80,08	79,80	80,36	80,92	**81,16**	80,92	81,22	81,18	81,08	81,44	81,00	81,08	81,18	81,26	**81,58**	81,32
20	79,96	80,32	80,78	80,64	80,86	81,08	81,24	81,24	81,14	81,30	81,20	81,42	81,38	81,36	81,34	81,48
25	79,98	**80,36**	80,80	80,82	80,84	80,82	81,16	81,36	81,48	81,26	81,38	**81,62**	81,46	81,78	81,38	81,74
30	**80,10**	80,30	**80,96**	80,92	80,82	80,80	**81,44**	81,44	81,44	81,40	81,40	81,46	81,66	**81,86**	81,46	**81,76**
35	79,96	80,16	80,94	80,90	80,78	80,86	81,38	81,48	81,54	81,46	81,40	81,42	81,64	81,84	81,30	81,70
40	79,94	80,18	80,92	80,86	80,76	80,88	81,32	**81,50**	81,60	**81,52**	**81,42**	81,50	81,66	81,82	81,28	81,62
45	80,08	80,22	80,88	80,90	80,70	80,90	81,28	81,38	81,62	81,50	81,40	81,44	81,74	81,76	81,40	81,68
50	80,02	80,14	80,84	80,84	80,82	80,94	81,32	81,42	81,62	81,48	81,38	81,40	81,72	81,82	81,44	81,68
55	80,02	80,18	80,88	80,82	80,80	80,94	81,30	81,42	**81,64**	81,50	81,38	81,40	**81,78**	81,86	81,40	81,68
60	80,04	80,16	80,88	80,80	80,84	80,90	81,30	81,42	81,64	81,52	81,38	81,42	81,78	81,86	81,40	81,66
65	80,02	80,16	80,88	80,80	80,84	80,92	81,30	81,42	81,64	81,52	81,38	81,40	81,78	81,86	81,40	81,68
70	80,02	80,16	80,86	80,80	80,84	80,92	81,30	81,42	81,64	81,52	81,38	81,40	81,78	81,86	81,40	81,68
75	80,02	80,16	80,86	80,80	80,84	80,92	81,30	81,42	81,64	81,52	81,38	81,40	81,78	81,86	81,40	81,68

A maior acurácia foi 81,86% e ocorreu com 30 componentes principais e $k = 18$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão por paciente pode ser consultada na Tabela 25.

Tabela 25 – Acurácia do KNN (em %) com as imagens da divisão por paciente

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	72,07	69,63	71,74	70,55	72,66	70,16	73,19	71,48	73,32	71,81	73,58	71,67	72,86	72,27	72,40	72,33
10	72,40	**71,21**	**73,32**	**72,13**	**74,11**	**73,06**	73,52	72,79	73,91	72,79	73,98	72,99	73,85	73,06	74,37	**73,58**
15	72,33	69,76	72,79	71,54	73,12	72,66	73,45	73,52	**74,57**	73,25	**74,77**	**73,98**	**74,64**	**73,98**	**74,97**	73,52
20	72,53	69,89	72,99	71,54	73,25	72,27	73,72	72,60	73,78	73,25	74,11	73,25	74,24	73,45	74,31	73,39
25	**72,66**	70,09	72,86	71,48	72,79	72,13	**73,85**	**73,58**	73,91	73,19	73,78	73,52	74,31	73,25	74,04	72,60
30	72,33	69,63	72,86	71,54	72,92	72,07	73,72	73,58	74,04	73,12	74,11	73,65	74,64	73,19	74,04	73,25
35	72,00	69,96	72,66	71,28	72,73	71,94	73,72	73,32	74,37	73,25	73,85	73,65	74,57	73,32	74,31	73,25
40	72,00	70,03	72,79	71,01	72,79	72,20	73,65	73,32	74,37	73,39	73,78	73,85	74,24	73,25	74,18	73,12
45	72,13	69,96	72,86	71,21	72,86	72,27	73,65	73,39	74,24	73,25	73,65	73,65	74,44	73,32	74,11	73,19
50	72,20	69,96	72,73	71,21	72,73	72,33	73,65	73,32	74,31	**73,45**	73,65	73,65	74,31	73,32	73,98	73,12
55	72,13	69,96	72,73	71,21	72,79	72,40	73,65	73,39	74,24	73,39	73,65	73,65	74,44	73,32	74,04	73,12
60	72,13	69,96	72,73	71,28	72,86	72,33	73,58	73,39	74,18	73,32	73,65	73,65	74,44	73,32	74,04	73,12
65	72,13	69,96	72,73	71,28	72,86	72,33	73,58	73,39	74,18	73,39	73,65	73,65	74,44	73,32	74,04	73,12
70	72,13	69,96	72,73	71,28	72,86	72,33	73,58	73,39	74,18	73,32	73,65	73,65	74,44	73,32	74,04	73,12
75	72,13	69,96	72,73	71,28	72,86	72,33	73,58	73,39	74,18	73,32	73,65	73,65	74,44	73,32	74,04	73,12

A maior acurácia foi 74,97% e ocorreu com 15 componentes principais e $k = 19$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão aleatória pode ser consultada na Tabela 26.

Tabela 26 – Acurácia do KNN (em %) com as imagens da divisão aleatória

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	73,72	72,59	74,21	73,43	74,46	74,51	74,75	74,61	74,95	74,41	75,34	74,36	75,34	74,85	75,05	75,15
10	**77,07**	76,08	**77,17**	**77,36**	**77,71**	**77,56**	**78,05**	**77,81**	77,66	**77,71**	**77,76**	76,82	77,41	77,17	77,12	76,92
15	76,62	**76,28**	77,02	76,33	77,12	77,36	78,05	77,41	**77,81**	77,51	77,66	**77,61**	**77,71**	77,41	77,17	76,97
20	76,67	75,74	76,67	76,67	77,41	76,48	77,46	76,92	76,77	77,17	77,46	77,26	77,26	77,07	77,07	76,87
25	76,53	75,94	76,92	77,02	77,36	76,53	77,17	76,48	77,21	76,92	77,61	77,17	77,36	77,07	77,26	**77,07**
30	76,53	75,89	76,57	77,02	77,46	76,67	77,41	76,77	77,21	76,97	77,51	77,51	77,36	**77,46**	77,31	76,87
35	76,53	75,89	76,72	76,87	77,41	76,92	77,46	76,67	77,41	77,07	77,66	77,51	77,41	77,26	77,31	76,97
40	76,33	75,89	76,72	76,92	77,51	77,12	77,31	76,77	77,36	77,02	77,66	77,51	77,21	77,26	77,41	76,92
45	76,43	75,84	76,82	76,77	77,61	76,97	77,36	76,62	77,41	77,07	77,66	77,41	77,26	77,21	**77,46**	76,97
50	76,53	75,89	76,82	76,82	77,41	76,92	77,41	76,67	77,36	77,12	77,71	77,51	77,26	77,26	77,41	76,82
55	76,53	75,89	76,87	76,77	77,31	76,97	77,36	76,72	77,46	77,07	77,71	77,51	77,26	77,26	77,46	76,77
60	76,57	75,89	76,87	76,77	77,31	76,97	77,36	76,72	77,46	77,07	77,71	77,51	77,26	77,26	77,46	76,77
65	76,53	75,89	76,87	76,77	77,31	76,97	77,36	76,72	77,46	77,07	77,71	77,51	77,26	77,26	77,46	76,77
70	76,53	75,89	76,87	76,77	77,31	76,97	77,36	76,72	77,46	77,07	77,71	77,51	77,26	77,26	77,46	76,77
75	76,53	75,89	76,87	76,77	77,31	76,97	77,36	76,72	77,46	77,07	77,71	77,51	77,26	77,26	77,46	76,77

A maior acurácia foi 78,05% e ocorreu com 10 componentes principais e $k = 11$.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens do conjuntos de teste, obtêm-se os resultados da Tabela 27.

Tabela 27 – Desempenho dos classificadores KNN nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	65,93%	62,39%	80,22%	51,65%	70,19%
Divisão por paciente (sem augm.)		58,52%	57,95%	62,09%	54,95%	59,95%
Divisão aleatória (com augm.)	*rndDiv_test*	74,78%	70,19%	86,14%	63,42%	77,35%
Divisão aleatória (sem augm.)		74,04%	71,62%	79,65%	68,44%	75,42%

5.2.3 Classificação utilizando Floresta Aleatória

Para o classificador Floresta Aleatória (RF) a relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente com *data augmentation* pode ser consultada na Tabela 28.

Tabela 28 – Acurácia do RF com as imagens da divisão por paciente com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	75,56%	75,30%	75,46%	75,76%	75,88%	75,78%
10	77,34%	77,90%	78,08%	78,20%	77,92%	78,14%
15	78,04%	78,32%	77,88%	78,06%	78,10%	78,00%
20	78,48%	78,36%	78,50%	78,16%	78,50%	78,50%
25	**78,74**%	77,88%	78,44%	78,36%	78,52%	78,36%
30	78,16%	78,46%	78,24%	78,42%	78,46%	78,20%
35	78,54%	78,56%	78,28%	78,16%	**78,82**%	78,56%
40	77,98%	78,32%	78,04%	78,48%	78,48%	**78,68**%
45	77,78%	77,90%	78,54%	78,62%	78,28%	78,60%
50	78,24%	78,30%	78,50%	**78,70**%	78,02%	78,58%
55	78,04%	78,22%	78,16%	78,36%	78,56%	78,64%
60	77,74%	78,04%	78,32%	78,36%	78,42%	78,34%
65	78,36%	78,34%	78,48%	78,66%	78,38%	78,52%
70	78,36%	78,44%	**78,78**%	78,44%	78,46%	78,40%
75	78,28%	**79,04**%	78,64%	78,28%	78,42%	78,26%

A maior acurácia foi 79,04% e ocorreu sem redução de dimensionalidade e 70 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória com *data augmentation* pode ser consultada na Tabela 29.

Tabela 29 – Acurácia do RF com as imagens da divisão aleatória com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	79,56%	79,26%	79,24%	79,74%	79,62%	80,14%
10	81,34%	81,76%	81,70%	81,58%	81,56%	82,02%
15	81,94%	82,14%	82,10%	82,20%	82,32%	82,26%
20	82,02%	81,86%	82,40%	82,28%	82,28%	82,28%
25	81,92%	82,12%	82,20%	82,50%	82,06%	82,56%
30	82,20%	82,62%	82,40%	82,32%	82,54%	82,40%
35	82,52%	82,38%	82,72%	82,54%	82,46%	82,20%
40	82,30%	82,24%	82,00%	82,64%	82,14%	82,26%
45	82,30%	**82,74**%	82,70%	82,58%	82,68%	82,40%
50	82,38%	82,36%	82,72%	82,32%	82,60%	82,88%
55	81,88%	82,60%	82,52%	82,48%	82,30%	82,64%
60	82,06%	82,40%	**83,02**%	82,46%	82,74%	82,64%
65	**82,56**%	82,32%	82,36%	82,66%	**82,90**%	82,58%
70	82,42%	82,30%	82,92%	**82,72**%	82,54%	**83,04**%
75	82,10%	82,50%	82,78%	82,60%	82,44%	82,66%

A maior acurácia foi 83,04% e ocorreu com 70 componentes principais e 150 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente pode ser consultada na Tabela 30.

Tabela 30 – Acurácia do RF com as imagens da divisão por paciente

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	73,72%	73,45%	73,06%	73,78%	73,25%	73,85%
10	76,28%	76,22%	76,48%	76,28%	76,28%	76,55%
15	**76,61**%	75,43%	**76,61**%	77,08%	**76,94**%	76,42%
20	75,82%	75,89%	76,02%	76,75%	76,55%	77,21%
25	75,56%	75,03%	76,09%	76,28%	76,55%	76,42%
30	76,35%	75,49%	75,63%	77,01%	76,75%	77,01%
35	74,90%	75,96%	76,02%	74,90%	75,76%	76,09%
40	74,44%	76,15%	76,48%	76,09%	75,63%	75,63%
45	75,43%	75,16%	76,35%	76,35%	76,68%	77,01%
50	75,89%	**76,28**%	76,22%	76,55%	76,68%	76,48%
55	75,69%	75,43%	76,28%	76,35%	76,88%	76,75%
60	75,63%	75,82%	76,22%	76,22%	76,61%	76,22%
65	76,61%	75,56%	74,84%	**77,21**%	76,88%	76,42%
70	75,16%	76,09%	75,63%	76,42%	76,42%	**77,34**%
75	75,82%	74,64%	76,35%	75,36%	75,63%	76,22%

A maior acurácia foi 77,34% e ocorreu com 70 componentes principais e 150 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória pode ser consultada na Tabela 31.

Tabela 31 – Acurácia do RF com as imagens da divisão aleatória

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	74,85%	75,10%	75,59%	75,44%	75,20%	75,89%
10	78,89%	78,89%	79,04%	78,54%	78,25%	78,35%
15	78,99%	78,15%	78,74%	79,04%	78,84%	78,64%
20	79,13%	78,54%	78,74%	78,89%	79,08%	79,08%
25	78,94%	78,49%	79,28%	78,74%	79,38%	79,33%
30	78,44%	77,95%	79,08%	79,18%	79,18%	79,68%
35	77,76%	79,68%	78,64%	79,08%	78,99%	79,08%
40	78,49%	78,54%	79,08%	78,99%	79,04%	79,08%
45	78,20%	78,20%	78,89%	78,79%	79,18%	79,38%
50	78,35%	79,63%	79,08%	79,28%	79,28%	78,99%
55	78,89%	79,38%	78,84%	79,38%	79,58%	78,89%
60	78,99%	79,18%	79,38%	79,38%	79,68%	79,87%
65	78,89%	78,94%	79,48%	**80,12**%	79,38%	79,92%
70	**79,43**%	78,94%	**80,12**%	79,97%	**79,72**%	**80,46**%
75	78,59%	**79,82**%	79,68%	79,63%	78,74%	78,99%

A maior acurácia foi 80,46% e ocorreu com 70 componentes principais e 150 árvores aleatórias.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 32.

Tabela 32 – Desempenho dos classificadores RF nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	67,86%	64,32%	80,22%	55,49%	71,40%
Divisão por paciente (sem augm.)		65,11%	64,25%	68,13%	62,09%	66,13%
Divisão aleatória (com augm.)	*rndDiv_test*	78,47%	72,29%	92,33%	64,60%	81,09%
Divisão aleatória (sem augm.)		77,14%	75,84%	79,65%	74,63%	77,70%

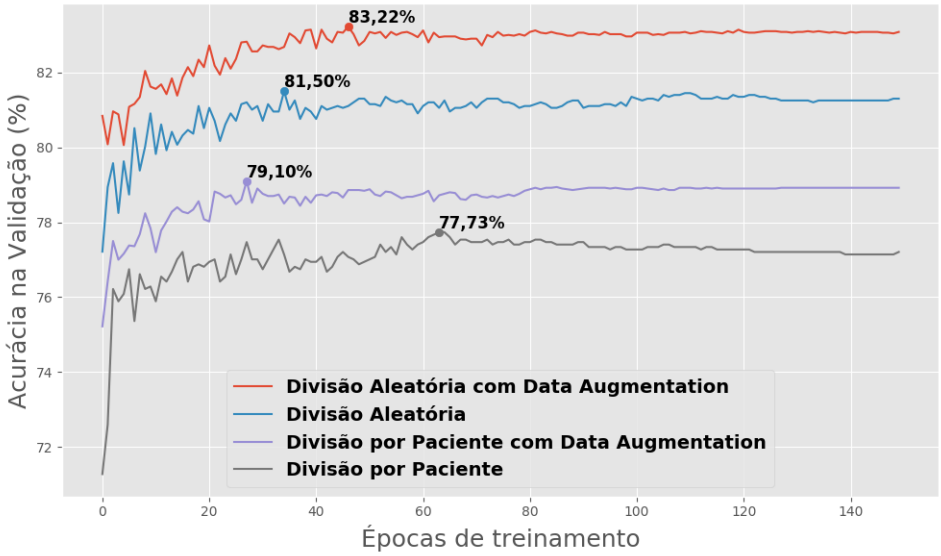
5.2.4 Classificação utilizando Redes Neurais

Das arquiteturas de Redes Neurais avaliadas, as que obtiveram as maiores acurácias foram:

- 3 camadas, cada qual com 128 neurônios, que utilizam ReLU como função de ativação, para as imagens da divisão por Paciente com *data augmentation*;
- 2 camadas, cada qual com 1024 neurônios, que utilizam ReLU como função de ativação, para as imagens da divisão aleatória com *data augmentation*;
- 4 camadas, cada qual com 512 neurônios, que utilizam ReLU como função de ativação, para as imagens da divisão por paciente;
- 2 camadas, cada qual com 1024 neurônios, que utilizam ReLU como função de ativação, para as imagens da divisão aleatória;

A acurácia na validação em função do número de épocas de treinamento de cada uma destas redes pode ser vista na Figura 51.

Figura 51 – Acurácia das Redes Neurais dos *dataframes* de validação em função das épocas de treinamento



As maiores acurácias foram: 83,22% com as imagens da divisão aleatória com *data augmentation*, 81,50% com as imagens da divisão aleatória, 79,10% com as imagens da divisão por paciente com *data augmentation* e 77,73% com as imagens da divisão por paciente.

Aplicando estás redes neurais, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 33.

Tabela 33 – Desempenho das Redes Neurais nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	64,56%	60,31%	85,16%	43,96%	70,61%
Divisão por paciente (sem augm.)		61,54%	59,37%	73,08%	50,00%	65,52%
Divisão aleatória (com augm.)	*rndDiv_test*	76,40%	71,16%	88,79%	64,01%	79,00%
Divisão aleatória (sem augm.)		78,02%	74,48%	85,25%	70,80%	79,50%

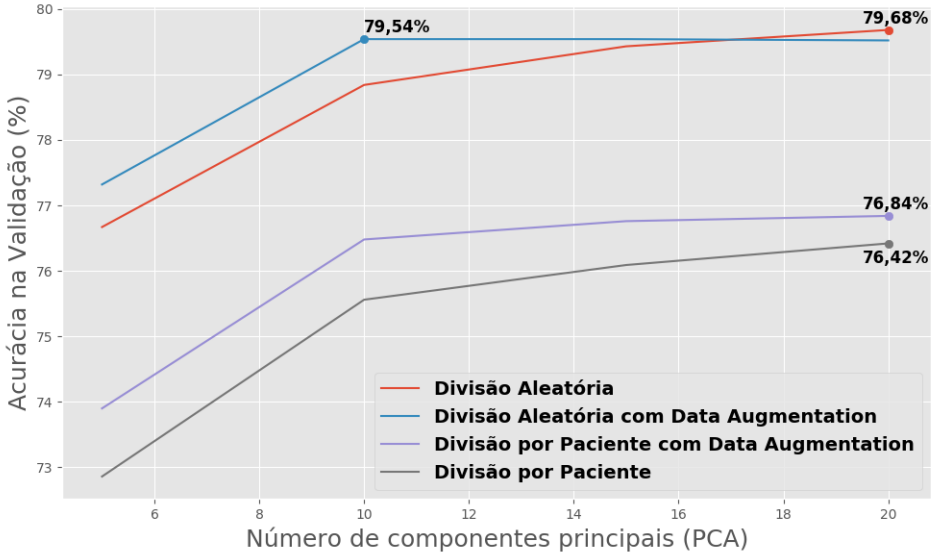
5.3 Classificação dos linfócitos utilizando as características morfológicas

Os resultados apresentados nesta seção são referentes a classificação dos linfócitos utilizando as características morfológicas extraídas de suas imagens, as suas subseções estão organizadas da seguinte forma: na subseção 5.3.1 são apresentados os resultados obtidos com o algoritmo do SVM, na subseção 5.3.2 os resultados obtidos com o algoritmo KNN, na subseção 5.3.3 os resultados obtidos com algoritmo RF e, por fim, na subseção 5.3.4 os resultados obtidos com as Redes Neurais.

5.3.1 Classificação utilizando SVM

Para o classificador SVM com *kernel* linear (L-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 52.

Figura 52 – Acurácia do L-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



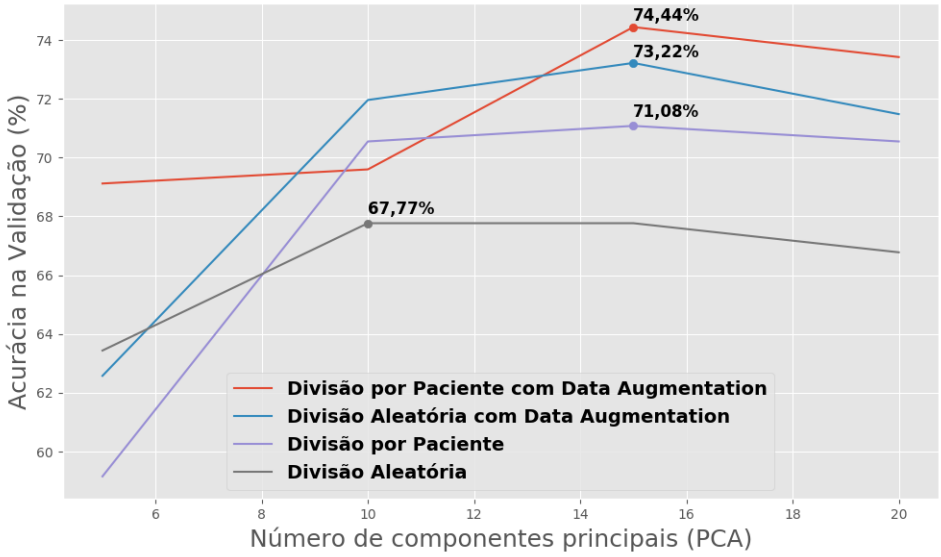
As maiores acurácias foram: 79,68% sem redução de dimensionalidade utilizando as imagens da divisão aleatória, 79,54% com 10 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 76,84% sem redução de dimensionalidade utilizando as imagens da divisão por paciente com *data augmentation* e 76,42% sem redução de dimensionalidade utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 34.

Tabela 34 – Desempenho dos classificadores L-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	69,78%	65,52%	83,52%	56,04%	73,43%
Divisão por paciente (sem augm.)		67,58%	65,09%	75,82%	59,34%	70,05%
Divisão aleatória (com augm.)	*rndDiv_test*	75,37%	74,29%	77,58%	73,16%	75,90%
Divisão aleatória (sem augm.)		74,48%	73,18%	77,29%	71,68%	75,18%

Para o classificador SVM com *kernel* quadrático (Q-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 53.

Figura 53 – Acurácia do Q-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



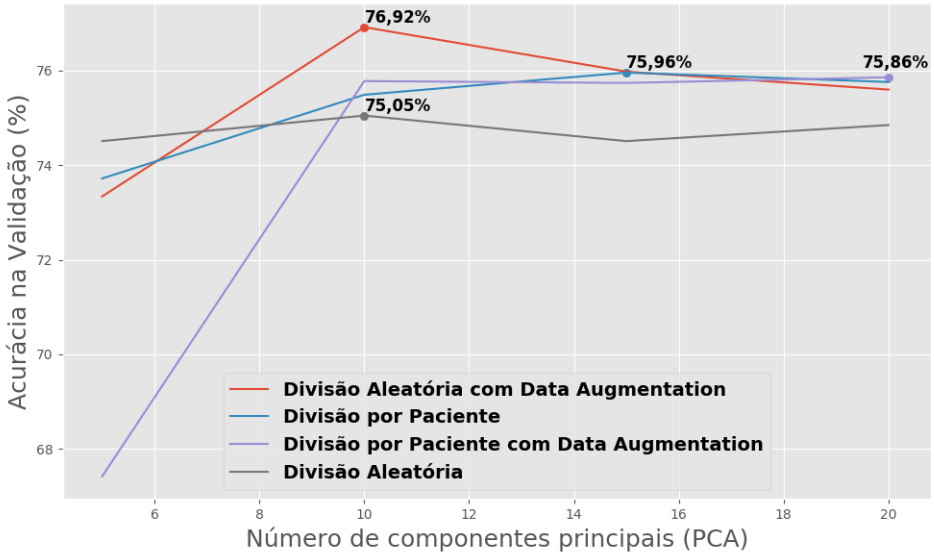
As maiores acurácias foram: 74,44% com 15 componentes principais utilizando as imagens da divisão por paciente com *data augmentation*, 73,22% com 15 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 71,08% com 15 componentes principais utilizando as imagens da divisão por paciente e 67,77% com 10 componentes principais utilizando as imagens da divisão aleatória. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 35.

Tabela 35 – Desempenho dos classificadores Q-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	65,38%	60,14%	91,21%	39,56%	72,49%
Divisão por paciente (sem augm.)		62,64%	58,10%	90,66%	34,62%	70,82%
Divisão aleatória (com augm.)	*rndDiv_test*	71,24%	65,25%	90,86%	51,62%	75,95%
Divisão aleatória (sem augm.)		65,63%	60,91%	87,32%	43,95%	71,76%

Para o classificador SVM com *kernel* polinomial (P-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 54.

Figura 54 – Acurácia do P-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



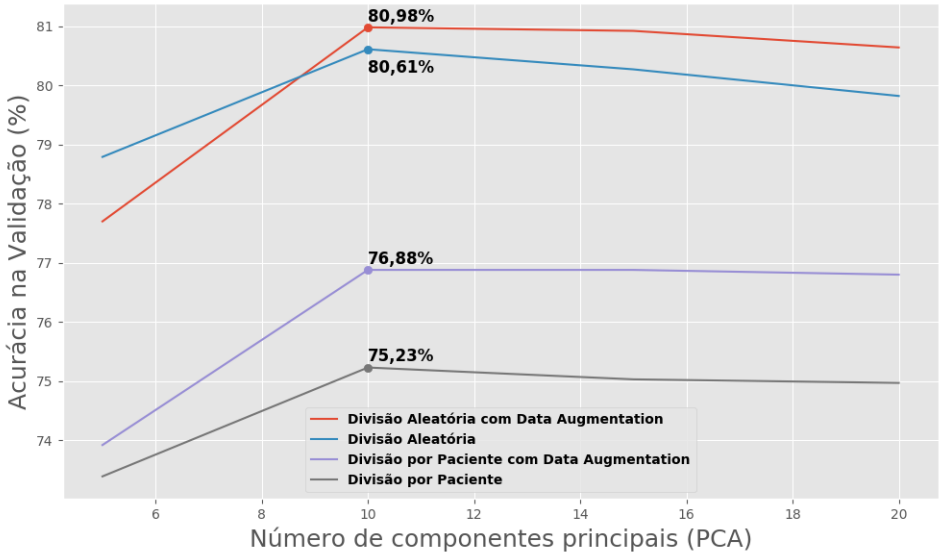
As maiores acurácias foram: 76,92% com 10 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 75,96% com 15 componentes principais utilizando as imagens da divisão por paciente, 75,86% sem redução de dimensionalidade utilizando as imagens da divisão por paciente com *data augmentation* e 75,05% com 10 componentes principais utilizando as imagens da divisão aleatória. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 36.

Tabela 36 – Desempenho dos classificadores P-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	66,21%	61,05%	89,56%	42,86%	72,61%
Divisão por paciente (sem augm.)		67,03%	61,23%	92,86%	41,21%	73,80%
Divisão aleatória (com augm.)	*rndDiv_test*	73,60%	69,70%	83,48%	63,72%	75,97%
Divisão aleatória (sem augm.)		74,34%	68,54%	89,97%	58,70%	77,81%

Para o classificador SVM com *kernel* RBF (R-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 55.

Figura 55 – Acurácia do R-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



As maiores acurácias foram: 80,98% com 10 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 80,61% com 10 componentes principais utilizando as imagens da divisão aleatória, 76,88% com 10 componentes principais utilizando as imagens da divisão por paciente com *data augmentation* e 75,23% com 10 componentes principais utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 37.

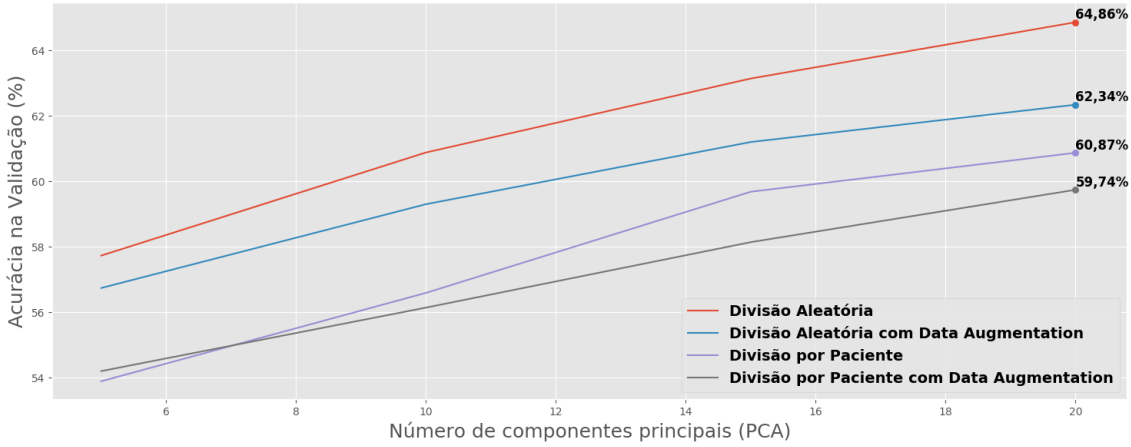
[Características morfológicas: Desempenho do Sig-SVM nos conjuntos de teste]

Tabela 37 – Desempenho dos classificadores R-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	66,21%	63,11%	78,02%	54,40%	69,78%
Divisão por paciente (sem augm.)		67,31%	64,38%	77,47%	57,14%	70,32%
Divisão aleatória (com augm.)	*rndDiv_test*	77,14%	74,34%	82,89%	71,39%	78,38%
Divisão aleatória (sem augm.)		76,11%	74,79%	78,76%	73,45%	76,72%

Para o classificador SVM com *kernel* sigmoide (Sig-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 56.

Figura 56 – Acurácia do Sig-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



As maiores acurácias foram: 64,86% sem redução de dimensionalidade utilizando as imagem da divisão aleatória, 62,34% sem redução de dimensionalidade utilizando as imagens da divisão aleatória com *data augmentation*, 60,87% sem redução de dimensionalidade utilizando as imagens da divisão por paciente e 59,74% sem redução de dimensionalidade com as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 38.

Tabela 38 – Desempenho dos classificadores Sig-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	48,08%	48,13%	49,45%	46,70%	48,78%
Divisão por paciente (sem augm.)		53,30%	53,19%	54,95%	51,65%	54,06%
Divisão aleatória (com augm.)	*rndDiv_test*	59,44%	58,65%	64,01%	54,87%	61,21%
Divisão aleatória (sem augm.)		62,68%	62,50%	63,42%	61,95%	62,96%

5.3.2 Classificação utilizando KNN

Para o classificador *K-Nearest Neighbors* (KNN) a relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação, utilizando as imagens da divisão por paciente com *data augmentation*, pode ser consultada na Tabela 39.

Tabela 39 – Acurácia do KNN (em %) com as imagens da divisão por paciente com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	73,38	72,62	73,82	73,14	73,84	73,14	73,70	73,34	74,06	73,46	74,04	73,78	74,06	73,66	73,86	73,78
10	**74,90**	73,26	**74,22**	**73,98**	**74,34**	**73,96**	74,38	74,12	74,54	74,08	**74,48**	74,04	**74,68**	74,34	**74,80**	**74,50**
15	74,70	73,34	74,04	73,96	74,28	73,86	74,50	**74,30**	**74,60**	**74,10**	74,38	**74,14**	74,56	74,34	74,72	74,40
20	74,70	**73,38**	74,08	73,96	74,30	73,88	**74,54**	74,30	74,58	74,10	74,38	74,10	74,54	**74,36**	74,72	74,42

A maior acurácia foi 74,90% e ocorreu com 10 componentes principais e $k = 5$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão da divisão aleatória com *data augmentation* pode ser consultada na Tabela 40.

Tabela 40 – Acurácia do KNN (em %) com as imagens da divisão aleatória com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	76,18	75,30	76,62	75,92	77,02	76,44	77,26	76,98	77,08	77,04	77,12	76,92	77,12	77,24	77,06	77,08
10	**78,40**	77,38	**78,74**	77,88	78,78	78,24	78,98	**78,86**	**78,84**	78,58	**78,84**	78,44	78,40	78,40	78,46	78,38
15	78,40	77,52	78,66	**78,30**	**78,82**	**78,52**	**79,38**	78,74	78,74	**78,70**	78,84	**78,70**	**78,58**	**78,48**	**78,70**	78,38
20	78,40	**77,54**	78,62	78,30	78,82	78,52	79,36	78,78	78,72	78,70	78,84	78,70	78,56	78,44	78,70	**78,40**

A maior acurácia foi 79,38% e ocorreu com 15 componentes principais e $k = 11$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão por paciente pode ser consultada na Tabela 41.

Tabela 41 – Acurácia do KNN (em %) com as imagens da divisão por paciente

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	71,61	70,88	72,00	**71,67**	72,66	72,79	73,45	72,79	**73,52**	**73,25**	**73,45**	**73,19**	**73,52**	**73,06**	73,39	72,86
10	**72,66**	**71,94**	**72,60**	71,67	**73,32**	72,60	**73,65**	**72,99**	73,19	72,79	73,12	72,60	73,25	72,99	**73,85**	**73,52**
15	72,66	71,67	72,40	71,67	72,92	**72,92**	73,52	72,79	73,39	72,53	73,19	72,73	73,45	72,92	73,78	73,45
20	72,66	71,67	72,40	71,67	72,92	72,79	73,52	72,79	73,39	72,53	73,19	72,73	73,45	72,92	73,85	73,45

A maior acurácia foi 73,85% e ocorreu com 10 componentes principais e $k = 19$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão aleatória pode ser consultada na Tabela 42.

Tabela 42 – Acurácia do KNN (em %) com as imagens da divisão aleatória

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
5	75,89	74,80	77,07	76,33	76,72	75,98	**77,26**	76,92	76,87	76,72	77,46	77,51	77,66	77,56	77,56	77,76
10	77,76	**76,33**	77,26	**76,82**	77,66	76,87	77,21	**77,61**	78,20	**78,10**	**78,54**	78,35	**78,25**	**78,15**	78,10	**78,49**
15	77,90	76,33	**77,36**	76,77	**77,90**	**77,12**	77,12	77,56	78,20	77,90	78,40	78,35	78,20	78,10	**78,44**	78,40
20	**77,95**	76,33	77,36	76,77	77,90	77,12	77,17	77,56	**78,25**	77,90	78,40	**78,40**	78,15	78,10	78,44	78,40

A maior acurácia foi 78,49% e ocorreu com 10 componentes principais e $k = 20$.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens do conjuntos de teste, obtêm-se os resultados da Tabela 43.

Tabela 43 – Desempenho dos classificadores KNN nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	69,23%	66,20%	78,57%	59,89%	71,86%
Divisão por paciente (sem augm.)		63,46%	62,07%	69,23%	57,69%	65,45%
Divisão aleatória (com augm.)	*rndDiv_test*	73,60%	71,62%	78,17%	69,03%	74,75%
Divisão aleatória (sem augm.)		75,81%	74,51%	78,47%	73,16%	76,44%

5.3.3 Classificação utilizando Floresta Aleatória

Para o classificador Floresta Aleatória (RF) a relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente com *data augmentation* pode ser consultada na Tabela 44.

Tabela 44 – Acurácia do RF com as imagens da divisão por paciente com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	75,50%	74,86%	75,12%	75,36%	75,46%	75,46%
10	77,26%	77,62%	77,50%	77,64%	77,46%	77,34%
15	78,44%	77,82%	**78,06%**	78,52%	77,84%	78,38%
20	**78,74%**	**78,66%**	78,06%	**79,16%**	**78,76%**	**78,66%**

A maior acurácia foi 79,16% e ocorreu sem redução de dimensionalidade com 110 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória com *data augmentation* pode ser consultada na Tabela 45.

Tabela 45 – Acurácia do RF com as imagens da divisão aleatória com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	78,68%	78,56%	79,28%	79,60%	79,28%	79,14%
10	82,32%	82,74%	82,62%	82,50%	82,88%	82,84%
15	82,62%	82,74%	82,60%	82,72%	82,76%	82,66%
20	**82,92%**	**83,18%**	**83,14%**	**82,88%**	**83,14%**	**83,06%**

A maior acurácia foi 83,18% e ocorreu sem redução de dimensionalidade e 70 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente pode ser consultada na Tabela 46.

Tabela 46 – Acurácia do RF com as imagens da divisão por paciente

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	74,24%	74,77%	74,57%	74,64%	74,37%	74,57%
10	77,60%	77,87%	77,60%	77,34%	78,19%	77,93%
15	77,93%	77,60%	**78,52%**	**78,72%**	78,33%	78,00%
20	**78,13%**	**78,39%**	78,52%	78,72%	**78,52%**	**78,72%**

A maior acurácia foi 78,72% e ocorreu com 15 componentes principais e 110 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória pode ser consultada na Tabela 47.

Tabela 47 – Acurácia do RF com as imagens da divisão aleatória

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
5	79,18%	78,89%	79,43%	79,68%	78,64%	79,28%
10	82,14%	82,43%	82,48%	82,43%	83,12%	82,48%
15	**83,37**%	83,32%	**82,97**%	82,97%	83,22%	83,22%
20	82,78%	**83,46**%	82,23%	**83,07**%	**83,32**%	**83,46**%

A maior acurácia foi 83,46% e ocorreu sem redução de dimensionalidade e 70 árvores aleatórias.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 48.

Tabela 48 – Desempenho dos classificadores RF nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	70,88%	67,12%	81,87%	59,89%	73,76%
Divisão por paciente (sem augm.)		68,68%	66,35%	75,82%	61,54%	70,77%
Divisão aleatória (com augm.)	*rndDiv_test*	76,99%	73,89%	83,48%	70,50%	78,39%
Divisão aleatória (sem augm.)		79,35%	78,03%	81,71%	76,99%	79,83%

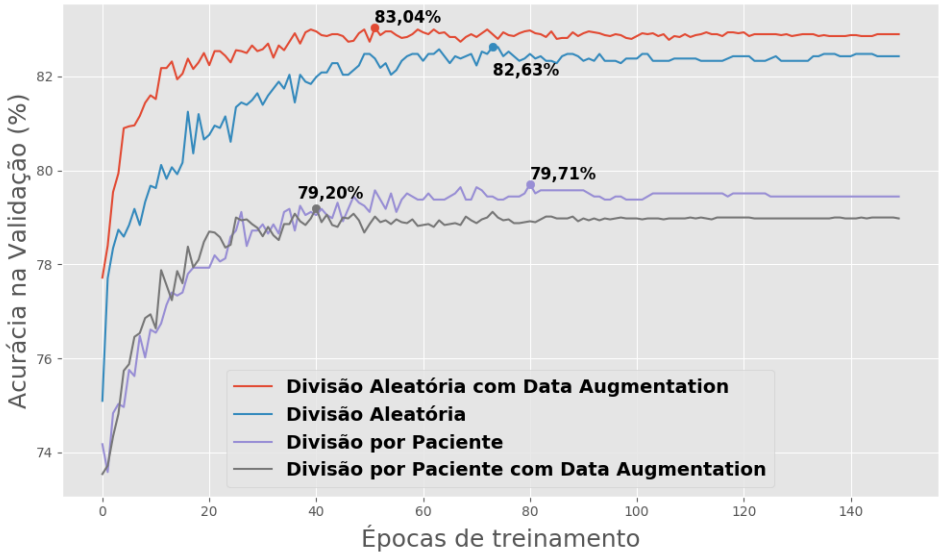
5.3.4 Classificação utilizando Redes Neurais

Das arquiteturas de Redes Neurais avaliadas, as que obtiveram as maiores acurácias foram:

- 4 camadas, cada qual com 128 neurônios, que utilizam tangente hiperbólica como função de ativação, para as imagens da divisão por Paciente com *data augmentation*;
- 4 camadas, cada qual com 512 neurônios, que utilizam ReLU como função de ativação, para as imagens da divisão aleatória com *data augmentation*;
- 3 camadas, cada qual com 512 neurônios, que utilizam ReLU como função de ativação, para as imagens da divisão por paciente;
- 4 camadas, cada qual com 1024 neurônios, que utilizam ReLU como função de ativação, para as imagens da divisão aleatória;

A acurácia na validação em função do número de épocas de treinamento de cada uma destas redes pode ser vista na Figura 57.

Figura 57 – Acurácia das Redes Neurais dos *dataframes* de validação em função das épocas de treinamento



As maiores acurácias foram: 83,04% com as imagens da divisão aleatória com *data augmentation*, 82,63% com as imagens da divisão aleatória, 79,71% com as imagens da divisão por paciente e 79,20% com as imagens da divisão por paciente com *data augmentation*.

Aplicando estas redes neurais, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 49.

Tabela 49 – Desempenho das Redes Neurais nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	68,68%	64,41%	83,52%	53,85%	72,73%
Divisão por paciente (sem augm.)		68,13%	64,10%	82,42%	53,85%	72,11%
Divisão aleatória (com augm.)	*rndDiv_test*	79,50%	73,81%	91,45%	67,55%	81,69%
Divisão aleatória (sem augm.)		80,24%	76,76%	86,73%	73,75%	81,44%

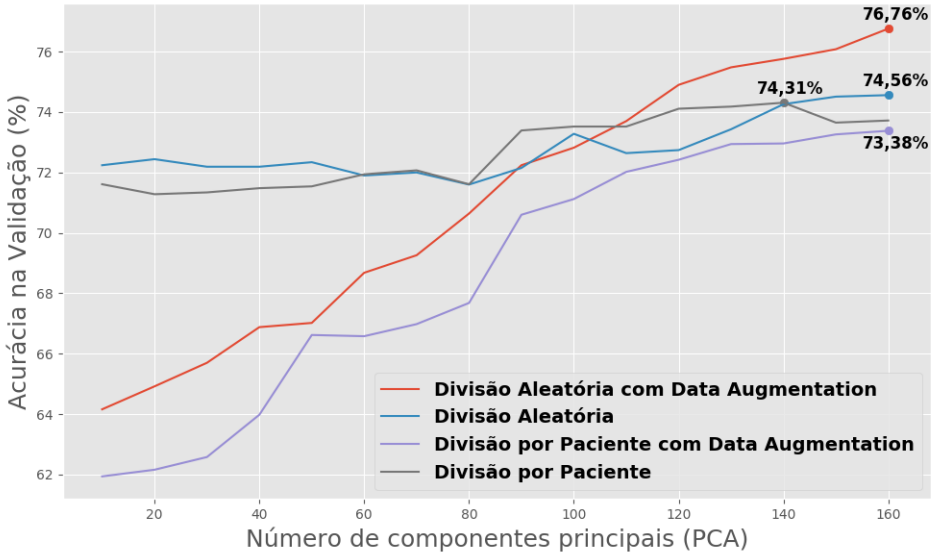
5.4 Classificação dos linfócitos utilizando as características de contorno

Os resultados apresentados nesta seção são referentes a classificação dos linfócitos utilizando as características de contorno extraídas da função distância do centroide extraídas de suas imagens, as suas subseções estão organizadas da seguinte forma: na subseção 5.4.1 serão apresentados os resultados obtidos com o algoritmo do SVM, na subseção 5.4.2 os resultados obtidos com o algoritmo KNN, na subseção 5.4.3 os resultados obtidos com algoritmo RF e, por fim, na subseção 5.4.4 os resultados obtidos com as Redes Neurais.

5.4.1 Classificação utilizando SVM

Para o classificador SVM com *kernel* linear (L-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 58.

Figura 58 – Acurácia do L-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



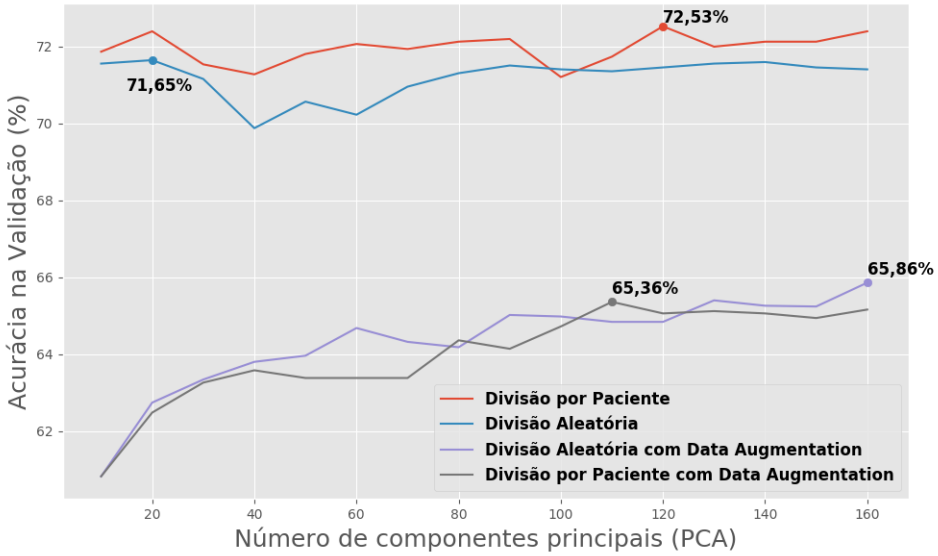
As maiores acurácias foram: 76,76% sem redução de dimensionalidade utilizando as imagens da divisão aleatória com *data augmentation*, 74,56% sem redução de dimensionalidade utilizando as imagens da divisão aleatória, 74,31% com 140 componentes principais utilizando as imagens da divisão por paciente e 73,38% sem redução de dimensionalidade utilizando as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 50.

Tabela 50 – Desempenho dos classificadores L-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	60,99%	59,80%	67,03%	54,95%	63,21%
Divisão por paciente (sem augm.)		62,64%	60,95%	70,33%	54,95%	65,30%
Divisão aleatória (com augm.)	*rndDiv_test*	73,60%	72,73%	75,52%	71,68%	74,10%
Divisão aleatória (sem augm.)		74,04%	72,96%	76,40%	71,68%	74,64%

Para o classificador SVM com *kernel* quadrático (Q-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 59.

Figura 59 – Acurácia do Q-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



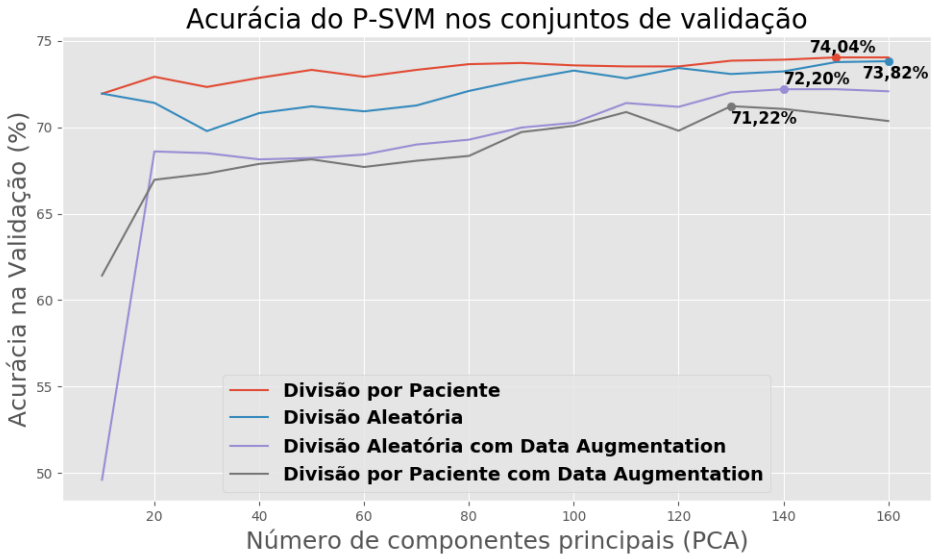
As maiores acurácias foram: 72,53% com 120 componentes principais utilizando as imagens da divisão por paciente, 71,65% com 20 componentes principais utilizando as imagens da divisão aleatória, 65,86% sem redução de dimensionalidade utilizando as imagens da divisão aleatória com *data augmentation* e 65,36% com 110 componentes principais utilizando as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 51.

Tabela 51 – Desempenho dos classificadores Q-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	57,42%	57,22%	58,79%	56,04%	57,99%
Divisão por paciente (sem augm.)		58,79%	57,41%	68,13%	49,45%	62,31%
Divisão aleatória (com augm.)	*rndDiv_test*	68,14%	69,40%	64,90%	71,39%	67,07%
Divisão aleatória (sem augm.)		69,47%	67,28%	75,81%	63,13%	71,29%

Para o classificador SVM com *kernel* polinomial (P-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 60.

Figura 60 – Acurácia do P-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



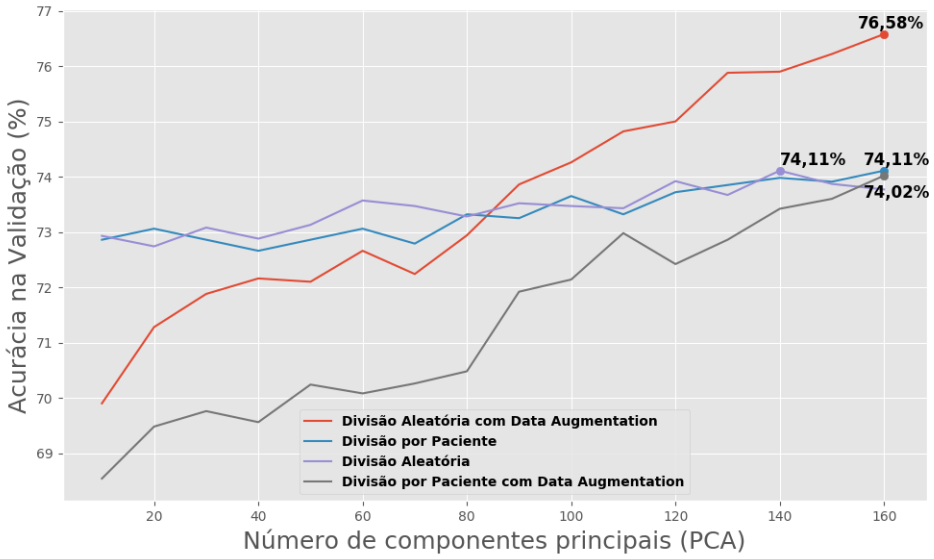
As maiores acurácias foram: 74,04% com 150 componentes principais utilizando as imagens da divisão por paciente, 73,82% sem redução de dimensionalidade utilizando as imagens da divisão aleatória, 72,20% com 140 componentes principais utilizando as imagens da divisão aleatória com *data augmentation* e 71,22% com 130 componentes principais utilizando as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 52.

Tabela 52 – Desempenho dos classificadores P-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	62,91%	60,93%	71,98%	53,85%	66,00%
Divisão por paciente (sem augm.)		61,54%	59,05%	75,27%	47,80%	66,18%
Divisão aleatória (com augm.)	*rndDiv_test*	72,12%	70,05%	77,29%	66,96%	73,49%
Divisão aleatória (sem augm.)		71,53%	67,30%	83,78%	59,29%	74,64%

Para o classificador SVM com *kernel* RBF (R-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 61.

Figura 61 – Acurácia do R-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



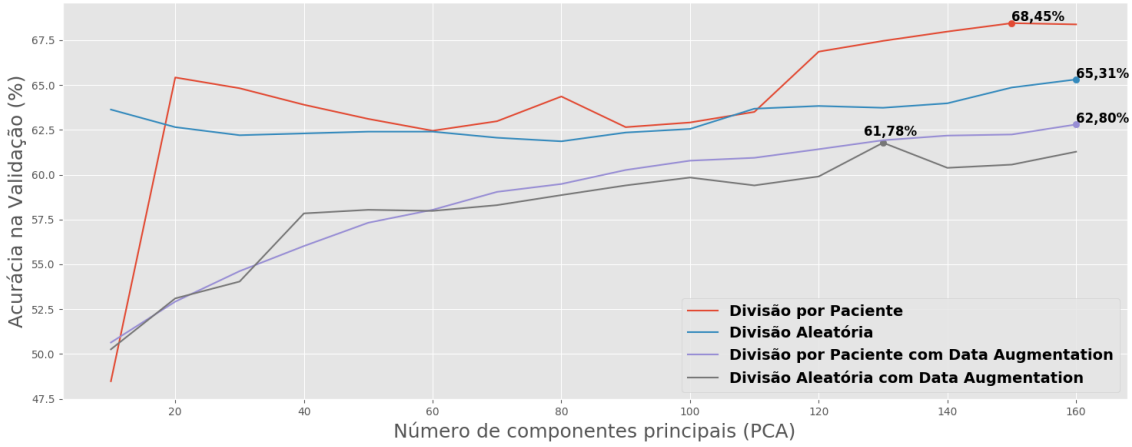
As maiores acurácias foram: 76,58% sem redução de dimensionalidade utilizando as imagens da divisão aleatória com *data augmentation*, 74,11% com 140 componentes principais utilizando as imagens da divisão aleatória, 74,11% sem redução de dimensionalidade utilizando as imagens da divisão por paciente e 74,04% sem redução de dimensionalidade utilizando as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 53.

Tabela 53 – Desempenho dos classificadores R-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	61,54%	59,72%	70,88%	52,20%	64,82%
Divisão por paciente (sem augm.)		60,16%	58,37%	70,88%	49,45%	64,02%
Divisão aleatória (com augm.)	*rndDiv_test*	73,60%	70,73%	80,53%	66,67%	75,31%
Divisão aleatória (sem augm.)		71,98%	69,35%	78,76%	65,19%	73,76%

Para o classificador utilizando SVM com *kernel* sigmoide (Sig-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 62.

Figura 62 – Acurácia do Sig-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



As maiores acurácias foram: 68,45% com 150 componentes principais utilizando as imagem da divisão por paciente, 65,31% sem redução de dimensionalidade utilizando as imagens da divisão aleatória, 62,80% sem redução de dimensionalidade utilizando as imagens da divisão por paciente com *data augmentation* e 61,78% com 130 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 54.

Tabela 54 – Desempenho dos classificadores Sig-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	54,40%	54,65%	51,65%	57,14%	53,11%
Divisão por paciente (sem augm.)		57,42%	56,99%	60,44%	54,40%	58,66%
Divisão aleatória (com augm.)	*rndDiv_test*	63,72%	63,88%	63,13%	64,31%	63,50%
Divisão aleatória (sem augm.)		61,65%	62,23%	59,29%	64,01%	60,72%

5.4.2 Classificação utilizando KNN

Para o classificador *K-Nearest Neighbors* (KNN) a relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação, utilizando as imagens da divisão por paciente com *data augmentation*, pode ser consultada na Tabela 55.

Tabela 55 – Acurácia do KNN (em %) com as imagens da divisão por paciente com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
10	65,76	64,42	66,68	65,58	67,04	66,20	67,62	66,62	67,40	66,76	67,46	66,76	67,42	67,04	67,24	66,84
20	66,98	65,10	67,54	65,50	66,80	66,50	67,08	67,12	68,18	67,70	68,24	67,28	68,16	67,40	67,86	67,54
30	67,22	65,34	67,08	66,22	67,12	65,58	67,92	**67,80**	67,80	67,68	67,56	67,86	**68,60**	68,04	68,50	67,54
40	66,98	65,04	67,10	66,36	67,74	67,14	68,24	67,20	67,98	67,22	68,06	66,94	68,22	67,96	68,20	68,12
50	67,46	65,42	66,88	66,26	67,26	66,12	68,66	67,04	67,94	67,78	68,24	67,24	67,90	68,20	67,96	67,54
60	67,20	64,98	67,82	66,14	67,98	66,36	68,02	67,22	68,32	66,92	68,32	67,64	68,00	67,44	67,50	68,12
70	67,06	65,64	66,76	65,74	68,10	67,00	68,38	67,04	**68,58**	67,66	68,00	67,78	68,38	68,06	68,62	67,70
80	67,60	65,30	67,68	**66,94**	67,82	66,96	68,22	67,16	68,34	68,24	68,38	68,02	68,38	67,10	68,16	67,90
90	67,38	65,50	67,60	66,38	68,28	67,30	**68,78**	67,28	68,12	67,82	68,16	67,88	67,54	67,64	68,30	67,88
100	67,66	66,52	68,14	66,82	68,18	67,14	68,30	67,52	68,14	67,62	**68,94**	**68,14**	68,38	68,10	68,38	68,14
110	67,04	**66,84**	67,96	66,72	**68,32**	66,90	68,34	67,58	68,46	**68,34**	68,30	68,00	67,96	68,48	68,52	67,96
120	67,84	66,54	68,22	66,68	68,24	**67,62**	68,06	67,60	68,18	67,06	68,34	67,26	68,60	67,72	68,60	**68,42**
130	67,08	66,52	67,78	66,64	68,00	67,32	68,14	67,06	68,22	67,48	68,20	67,94	68,38	67,94	68,48	68,04
140	67,42	66,40	68,16	66,86	68,04	67,34	67,98	67,32	67,88	67,82	68,36	67,90	68,28	68,08	68,38	67,74
150	67,86	66,76	68,06	66,62	67,74	67,16	68,32	67,44	68,48	67,52	68,62	68,06	68,46	68,04	68,66	67,98
160	**67,92**	66,74	**68,26**	66,78	67,92	67,02	68,02	67,26	68,10	67,82	68,24	68,00	68,42	**68,54**	**68,86**	68,22

A maior acurácia foi 68,86% e ocorreu sem redução de dimensionalidade e com $k = 19$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão da divisão aleatória com *data augmentation* pode ser consultada na Tabela 56.

Tabela 56 – Acurácia do KNN (em %) com as imagens da divisão aleatória com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
10	65,38	64,44	66,44	65,26	67,54	66,32	67,82	66,76	68,04	67,12	68,36	67,40	68,64	67,80	68,72	68,08
20	67,00	66,08	67,74	65,96	68,72	68,24	69,06	67,74	69,10	68,18	68,76	68,12	69,26	69,08	70,00	69,44
30	66,24	65,88	68,84	67,90	68,54	66,96	68,82	67,90	69,36	68,16	68,98	68,66	69,46	68,52	70,02	69,50
40	67,28	65,80	68,70	66,72	68,66	68,14	69,48	68,68	69,32	68,20	69,38	68,94	69,28	69,28	70,02	69,28
50	67,90	66,22	68,36	67,66	68,68	68,20	68,94	67,84	69,56	68,74	69,58	69,18	69,46	68,80	69,60	70,02
60	67,36	66,44	68,30	67,12	69,42	67,48	69,02	68,28	70,04	69,40	69,14	69,74	69,96	69,08	69,36	69,52
70	68,22	66,40	68,34	67,56	69,26	67,78	68,90	68,62	69,10	68,82	69,34	68,62	69,42	69,08	70,12	69,70
80	67,74	66,26	68,12	68,42	69,10	68,60	69,30	68,96	69,32	69,50	70,22	69,58	69,72	70,08	69,04	69,78
90	68,24	66,90	69,06	67,68	69,80	68,06	69,28	69,12	70,00	68,90	70,02	69,74	69,40	69,66	70,36	69,12
100	**69,06**	67,08	69,36	67,96	69,48	68,90	69,16	68,76	70,66	69,02	69,60	69,48	69,66	69,92	70,14	69,96
110	68,96	67,34	69,38	68,80	69,74	69,34	69,16	69,82	69,86	69,34	69,94	69,82	70,00	69,48	70,32	69,80
120	68,54	67,10	69,52	68,52	69,10	68,88	70,14	**70,12**	70,28	69,90	70,28	70,06	69,86	69,74	70,02	69,92
130	68,40	67,00	69,42	69,02	70,20	69,56	70,08	69,36	69,88	69,32	70,24	69,98	70,08	69,74	69,92	69,90
140	68,80	67,88	69,20	68,76	70,12	69,44	70,16	69,60	70,34	69,56	**70,50**	69,84	70,44	69,94	70,26	69,98
150	68,38	67,26	69,68	69,00	**70,46**	69,24	70,06	69,72	70,24	69,80	70,24	**70,24**	**70,74**	**70,48**	**70,88**	**70,34**
160	68,58	**68,44**	**69,96**	**69,70**	70,38	**69,98**	**70,48**	69,96	**70,76**	**70,14**	70,48	70,10	70,52	70,22	70,22	70,02

A maior acurácia foi 70,88% e ocorreu com 150 componentes principais e $k = 19$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão por paciente pode ser consultada na Tabela 57.

Tabela 57 – Acurácia do KNN (em %) com as imagens da divisão por paciente

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
10	71,01	69,76	71,94	70,36	71,74	70,62	70,69	70,62	72,27	71,48	**72,79**	**72,07**	72,60	**72,27**	**72,99**	**72,66**
20	**72,33**	69,70	**72,27**	70,22	71,94	70,88	71,81	**71,67**	71,54	71,08	71,61	71,28	72,00	71,28	71,87	71,21
30	71,08	68,58	71,87	69,43	**72,27**	69,96	72,27	71,08	72,27	71,87	72,27	71,08	72,00	71,87	72,53	72,33
40	70,82	69,83	71,61	69,70	72,20	70,69	72,27	71,01	71,67	70,42	72,00	71,81	**72,92**	71,21	72,00	71,08
50	70,42	**70,09**	71,81	**71,28**	72,07	70,69	71,87	71,54	**72,46**	72,20	72,20	71,21	71,61	71,15	72,00	71,28
60	71,01	69,30	71,74	69,96	71,48	70,49	71,01	71,48	71,81	**72,27**	72,00	71,08	71,74	71,08	72,20	71,54
70	71,41	69,63	71,74	70,16	71,74	70,62	**72,40**	71,15	71,81	71,41	71,15	71,48	71,67	71,94	71,87	71,41
80	71,15	69,50	72,00	70,55	71,08	**71,21**	71,28	70,49	70,95	70,88	71,54	70,42	71,21	71,01	71,21	71,74
90	70,16	68,91	71,41	69,43	71,01	69,89	71,28	70,29	70,55	69,89	71,34	71,48	71,01	70,95	71,61	70,62
100	70,49	68,58	71,21	69,43	70,29	70,36	70,55	69,89	71,54	70,62	71,87	70,82	71,48	71,74	71,54	71,48
110	71,01	68,77	70,82	68,97	70,95	70,55	70,62	70,75	71,54	70,55	71,21	70,82	71,54	70,82	71,54	71,28
120	70,55	68,64	71,34	69,70	71,15	70,36	70,95	70,22	70,55	70,75	71,48	71,21	71,54	70,69	71,28	71,61
130	69,83	68,84	71,34	69,37	69,96	69,70	71,08	70,36	70,95	70,03	71,74	70,62	70,88	70,62	71,74	71,28
140	70,82	69,04	70,82	70,16	70,82	70,69	71,34	70,69	70,55	69,96	70,62	70,36	70,82	70,75	71,01	70,95
150	70,36	68,64	70,75	69,04	71,08	70,16	70,49	70,29	70,88	70,42	70,82	70,82	70,95	70,36	70,95	70,95
160	70,95	68,97	70,75	69,57	70,16	69,04	70,75	70,09	70,55	70,16	70,62	70,16	70,69	70,36	71,01	70,82

A maior acurácia foi 72,99% e ocorreu com 10 componentes principais e $k = 19$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão aleatória pode ser consultada na Tabela 58.

Tabela 58 – Acurácia do KNN (em %) com as imagens da divisão aleatória

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
10	70,13	68,21	70,62	69,44	70,77	69,93	71,56	70,62	71,36	71,56	71,56	71,56	71,65	71,56	71,95	72,19
20	68,85	68,01	70,47	70,08	70,92	70,37	71,70	**71,75**	71,70	**72,00**	72,59	72,00	72,29	71,85	72,34	72,29
30	68,85	68,50	69,83	69,05	70,37	70,52	71,36	70,28	71,95	71,06	71,41	71,85	71,95	71,51	72,05	71,95
40	69,88	67,32	71,16	69,69	70,77	70,77	71,16	71,06	71,75	71,36	72,15	71,65	72,05	71,85	72,49	72,19
50	69,59	67,67	70,42	68,21	71,75	70,03	**71,85**	71,21	72,15	71,31	**72,74**	71,75	71,60	71,16	72,00	71,51
60	70,18	68,55	70,42	69,88	71,70	69,88	71,65	70,37	71,85	71,80	71,95	71,21	71,95	72,00	71,85	71,36
70	69,78	68,65	**71,75**	70,03	71,60	69,39	71,21	70,72	71,70	71,70	71,75	71,01	71,95	71,51	72,10	72,15
80	69,14	68,01	69,98	70,18	71,36	70,67	71,75	70,82	71,85	71,41	71,95	71,46	72,05	70,82	71,95	72,00
90	69,49	67,81	70,42	69,83	71,11	70,13	71,80	70,87	**72,49**	71,56	71,56	71,60	72,39	71,85	71,95	71,51
100	**70,37**	69,19	71,26	**70,62**	**72,34**	70,23	71,65	70,32	71,46	70,67	71,41	71,16	72,15	72,00	72,54	71,75
110	69,19	68,06	71,21	69,59	72,19	70,52	71,36	71,06	72,00	70,03	71,90	71,80	72,39	71,60	72,44	71,90
120	69,54	68,90	70,96	70,42	70,96	70,18	71,36	70,32	70,82	70,57	71,80	71,21	72,24	71,85	72,24	**72,39**
130	68,95	67,86	70,47	69,19	71,21	**70,87**	71,06	70,42	71,21	70,42	71,46	71,11	71,90	72,15	72,39	72,29
140	68,55	68,21	70,47	69,09	71,21	70,47	70,92	70,52	71,31	70,57	71,85	71,56	72,05	72,00	72,39	72,19
150	69,49	69,14	70,32	69,29	71,36	70,82	71,60	71,56	71,80	71,21	71,70	71,41	72,39	72,29	72,49	72,39
160	69,64	**69,39**	70,96	69,93	72,19	70,42	71,56	71,06	72,05	71,70	72,19	**72,19**	**72,83**	**72,39**	**73,03**	72,34

A maior acurácia foi 73,03% e ocorreu sem redução de dimensionalidade e $k = 19$.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens do conjuntos de teste, obtêm-se os resultados da Tabela 59.

Tabela 59 – Desempenho dos classificadores KNN nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	57,97%	57,07%	64,29%	51,65%	60,47%
Divisão por paciente (sem augm.)		60,99%	59,01%	71,98%	50,00%	64,85%
Divisão aleatória (com augm.)	*rndDiv_test*	70,65%	69,66%	73,16%	68,14%	71,37%
Divisão aleatória (sem augm.)		69,47%	67,93%	73,75%	65,19%	70,72%

5.4.3 Classificação utilizando Floresta Aleatória

Para o classificador Floresta Aleatória (RF) a relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente com *data augmentation* pode ser consultada na Tabela 60.

Tabela 60 – Acurácia do RF com as imagens da divisão por paciente com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
10	67,94%	67,60%	67,90%	68,16%	68,56%	68,12%
20	68,36%	68,76%	68,78%	68,52%	68,74%	69,26%
30	**69,24**%	68,78%	68,82%	69,26%	69,38%	69,16%
40	68,60%	68,42%	68,64%	68,36%	68,86%	68,80%
50	67,98%	68,44%	68,90%	68,62%	68,66%	68,84%
60	67,50%	68,02%	68,32%	68,28%	68,46%	68,58%
70	68,56%	67,80%	68,44%	68,52%	68,66%	68,26%
80	68,20%	67,54%	68,56%	68,54%	69,24%	68,44%
90	67,40%	68,50%	68,64%	68,86%	68,40%	68,38%
100	68,64%	68,90%	68,66%	69,16%	69,20%	68,94%
110	68,04%	68,30%	68,58%	68,80%	69,34%	68,78%
120	69,24%	67,98%	68,24%	69,22%	68,36%	69,62%
130	67,52%	68,92%	69,06%	68,68%	69,36%	68,48%
140	68,32%	**69,22**%	68,54%	69,46%	68,80%	69,60%
150	69,16%	68,48%	69,04%	**69,52**%	**69,72**%	**69,74**%
160	68,26%	68,66%	**69,08**%	69,52%	69,72%	69,22%

A maior acurácia foi 69,74% e ocorreu com 150 componentes principais e com 150 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória com *data augmentation* pode ser consultada na Tabela 61.

Tabela 61 – Acurácia do RF com as imagens da divisão aleatória com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
10	69,30%	69,38%	69,38%	69,72%	69,86%	69,88%
20	69,34%	70,20%	70,54%	70,42%	71,46%	70,86%
30	**70,52**%	70,28%	70,80%	71,34%	71,02%	71,62%
40	70,44%	**71,00**%	70,56%	70,30%	70,58%	70,88%
50	70,02%	70,02%	**71,18**%	71,10%	70,92%	70,82%
60	69,40%	70,02%	70,60%	70,44%	70,54%	70,52%
70	70,16%	69,96%	70,18%	70,36%	69,64%	70,76%
80	69,10%	68,96%	70,04%	70,28%	69,96%	70,22%
90	69,84%	69,44%	70,08%	70,26%	70,44%	70,96%
100	69,02%	69,78%	70,06%	70,82%	70,60%	70,70%
110	69,90%	70,34%	70,18%	70,46%	71,02%	71,00%
120	69,18%	70,92%	70,46%	70,92%	71,18%	**71,92**%
130	69,86%	70,84%	71,18%	70,52%	70,88%	71,18%
140	70,10%	70,62%	70,54%	70,44%	70,90%	71,26%
150	69,84%	70,44%	71,14%	**71,48**%	71,04%	71,44%
160	69,06%	70,92%	70,74%	71,20%	**71,74**%	71,54%

A maior acurácia foi 71,92% e ocorreu com 120 componentes principais e 150 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente pode ser consultada na Tabela 62.

Tabela 62 – Acurácia do RF com as imagens da divisão por paciente

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
10	71,48%	72,66%	72,00%	71,94%	71,94%	71,81%
20	71,15%	72,07%	72,27%	72,53%	72,07%	71,87%
30	**72,86**%	72,07%	72,79%	72,27%	72,20%	72,13%
40	71,87%	72,13%	71,48%	71,15%	72,86%	72,13%
50	72,13%	71,28%	72,60%	71,54%	72,79%	71,94%
60	71,81%	72,40%	72,07%	71,74%	72,53%	72,60%
70	71,67%	71,81%	71,74%	72,86%	72,66%	72,20%
80	71,81%	71,41%	72,40%	72,20%	72,46%	72,13%
90	71,81%	72,99%	71,61%	**73,25**%	72,60%	72,40%
100	72,40%	71,41%	72,07%	72,07%	72,79%	72,07%
110	71,67%	72,27%	72,79%	72,92%	72,73%	72,53%
120	72,27%	73,25%	72,27%	72,46%	71,94%	72,07%
130	72,40%	71,67%	72,40%	72,66%	72,53%	72,53%
140	71,08%	72,73%	72,66%	73,12%	72,53%	71,94%
150	71,67%	71,67%	72,73%	72,00%	72,53%	**72,86**%
160	72,20%	**73,45**%	**73,25**%	72,60%	**72,99**%	71,94%

A maior acurácia foi 73,45% e ocorreu sem redução de dimensionalidade e 70 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória pode ser consultada na Tabela 63.

Tabela 63 – Acurácia do RF com as imagens da divisão aleatória

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
10	72,69%	72,29%	72,39%	72,39%	72,93%	73,08%
20	72,29%	72,83%	73,62%	73,43%	73,72%	73,28%
30	72,34%	72,29%	73,33%	73,38%	73,77%	73,23%
40	72,44%	73,38%	72,79%	73,33%	73,23%	73,52%
50	72,39%	72,79%	74,06%	73,43%	73,38%	73,82%
60	**73,52**%	73,47%	73,87%	72,79%	74,51%	74,26%
70	72,79%	72,83%	73,62%	73,43%	74,02%	74,31%
80	73,18%	73,13%	**74,56**%	73,38%	73,28%	74,06%
90	72,88%	72,29%	74,56%	73,82%	73,82%	73,97%
100	72,79%	73,47%	73,23%	74,26%	74,51%	74,21%
110	72,19%	**74,21**%	73,43%	74,06%	73,67%	74,46%
120	73,33%	73,18%	72,83%	72,79%	73,67%	73,82%
130	73,33%	73,33%	74,16%	**74,46**%	74,16%	**75,10**%
140	72,24%	73,38%	72,88%	74,41%	**74,56**%	74,56%
150	71,90%	73,08%	73,87%	73,82%	74,16%	74,16%
160	72,98%	72,98%	73,03%	73,18%	73,43%	73,92%

A maior acurácia foi 75,10% e ocorreu com 130 componentes principais e 150 árvores aleatórias.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 64.

Tabela 64 – Desempenho dos classificadores RF nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	58,79%	57,92%	64,29%	53,30%	60,94%
Divisão por paciente (sem augm.)		58,79%	57,41%	68,13%	49,45%	62,31%
Divisão aleatória (com augm.)	*rndDiv_test*	71,98%	72,37%	71,09%	72,86%	71,72%
Divisão aleatória (sem augm.)		71,09%	68,57%	77,88%	64,31%	72,93%

5.4.4 Classificação utilizando Redes Neurais

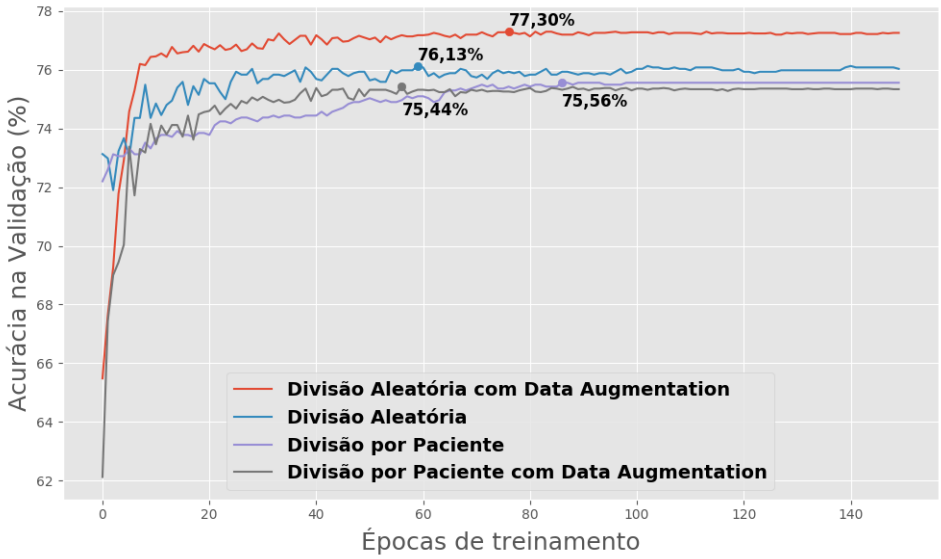
Das arquiteturas de Redes Neurais avaliadas, as que obtiveram as maiores acurácias foram:

- 2 camadas, cada qual com 1024 neurônios, que utilizam sigmoide como função de ativação, para as imagens da divisão por Paciente com *data augmentation*;
- 3 camadas, cada qual com 512 neurônios, que utilizam sigmoide como função de ativação, para as imagens da divisão aleatória com *data augmentation*;

- 2 camadas, cada qual com 128 neurônios, que utilizam ReLU como função de ativação, para as imagens da divisão por paciente;
- 4 camadas, cada qual com 1024 neurônios, que utilizam tangente hiperbólica como função de ativação, para as imagens da divisão aleatória;

A acurácia na validação em função do número de épocas de treinamento de cada uma destas redes pode ser vista na Figura 63.

Figura 63 – Acurácia das Redes Neurais dos *dataframes* de validação em função das épocas de treinamento



As maiores acurácias foram: 77,30% com as imagens da divisão aleatória com *data augmentation*, 76,13% com as imagens da divisão aleatória, 75,56% com as imagens da divisão por paciente e 75,44% com as imagens da divisão por paciente com *data augmentation*.

Aplicando estas redes neurais, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 65.

Tabela 65 – Desempenho das Redes Neurais nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	62,09%	59,48%	75,82%	48,35%	66,66%
Divisão por paciente (sem augm.)		60,16%	58,69%	68,68%	51,65%	63,29%
Divisão aleatória (com augm.)	*rndDiv_test*	75,07%	70,33%	86,73%	63,42%	77,67%
Divisão aleatória (sem augm.)		69,47%	70,50%	66,96%	71,98%	68,68%

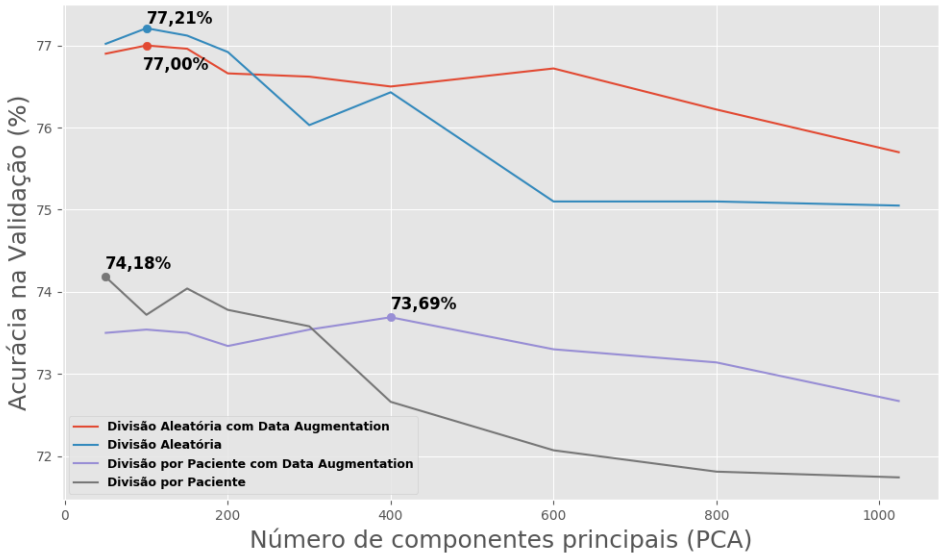
5.5 Classificação dos linfócitos utilizando a transformada discreta do cosseno

Os resultados apresentados nesta seção são referentes a classificação dos linfócitos utilizando as características da transformada discreta do cosseno extraídas de suas imagens, as suas subseções estão organizadas da seguinte forma: na subseção 5.5.1 são apresentados os resultados obtidos com o algoritmo do SVM, na subseção 5.5.2 os resultados obtidos com o algoritmo KNN, na subseção 5.5.3 os resultados obtidos com algoritmo RF e, por fim, na subseção 5.5.4 os resultados obtidos com as Redes Neurais.

5.5.1 Classificação utilizando SVM

Para o classificador SVM com *kernel* linear (L-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 64.

Figura 64 – Acurácia do L-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



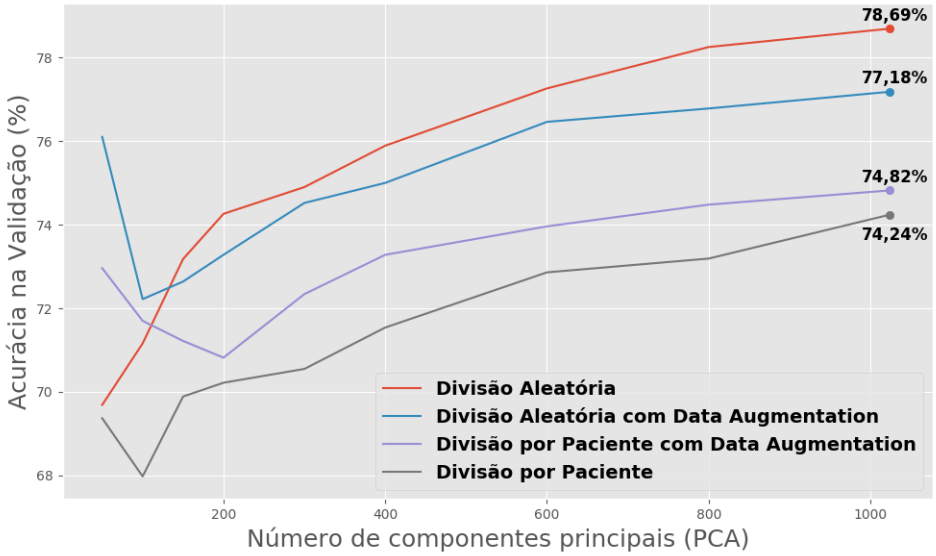
As maiores acurácias foram: 77,21% com 100 componentes principais utilizando as imagens da divisão aleatória, 77,00% com 100 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 74,18% com 50 componentes principais utilizando as imagens da divisão por paciente e 73,69% com 400 componentes principais utilizando as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 66.

Tabela 66 – Desempenho dos classificadores L-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	59,34%	58,10%	67,03%	51,65%	62,25%
Divisão por paciente (sem augm.)		64,01%	61,54%	74,73%	53,30%	67,50%
Divisão aleatória (com augm.)	*rndDiv_test*	74,78%	72,83%	79,06%	70,50%	75,82%
Divisão aleatória (sem augm.)		72,57%	71,55%	74,93%	70,21%	73,20%

Para o classificador SVM com *kernel* quadrático (Q-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia nos conjuntos de validação pode ser vista na Figura 65.

Figura 65 – Acurácia do Q-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



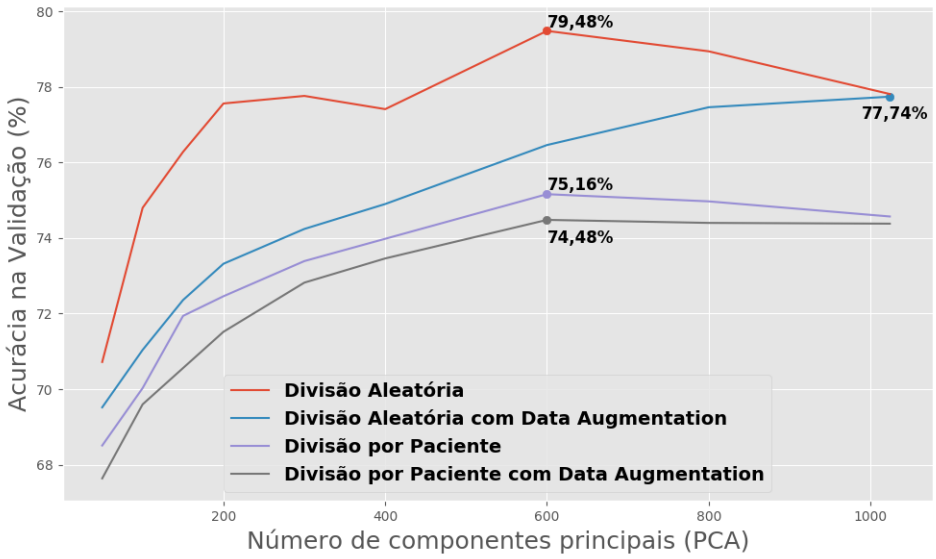
As maiores acurácias foram: 78,69% sem redução de dimensionalidade utilizando as imagens da divisão aleatória, 77,18% sem redução de dimensionalidade utilizando as imagens da divisão aleatória com *data augmentation*, 74,82% sem redução de dimensionalidade utilizando as imagens da divisão por paciente com *data augmentation* e 74,24% sem redução de dimensionalidade utilizando as imagens da divisão por paciente. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 67.

Tabela 67 – Desempenho dos classificadores Q-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	66,48%	63,76%	76,37%	56,59%	69,50%
Divisão por paciente (sem augm.)		64,29%	63,83%	65,93%	62,64%	64,86%
Divisão aleatória (com augm.)	*rndDiv_test*	74,04%	73,22%	75,81%	72,27%	74,49%
Divisão aleatória (sem augm.)		74,34%	74,05%	74,93%	73,75%	74,49%

Para o classificador SVM com *kernel* polinomial (P-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 66.

Figura 66 – Acurácia do P-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



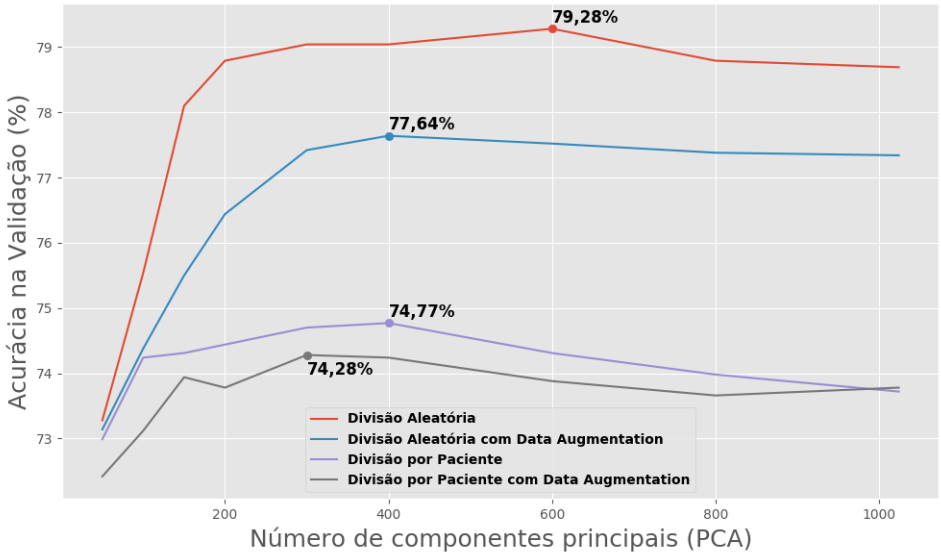
As maiores acurácias foram: 79,48% com 600 componentes principais utilizando as imagens da divisão aleatória, 77,74% sem redução de dimensionalidade utilizando as imagens da divisão aleatória com *data augmentation*, 75,16% com 600 componentes principais utilizando as imagens da divisão por paciente e 74,48% com 600 componentes principais utilizando as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de testee, obtêm-se os resultados da Tabela 68.

Tabela 68 – Desempenho dos classificadores P-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	67,03%	64,62%	75,27%	58,79%	69,54%
Divisão por paciente (sem augm.)		67,03%	66,85%	67,58%	66,48%	67,21%
Divisão aleatória (com augm.)	*rndDiv_test*	73,60%	74,39%	71,98%	75,22%	73,17%
Divisão aleatória (sem augm.)		75,52%	76,13%	74,34%	76,70%	75,22%

Para o classificador utilizando SVM com *kernel* RBF (R-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 67.

Figura 67 – Acurácia do R-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



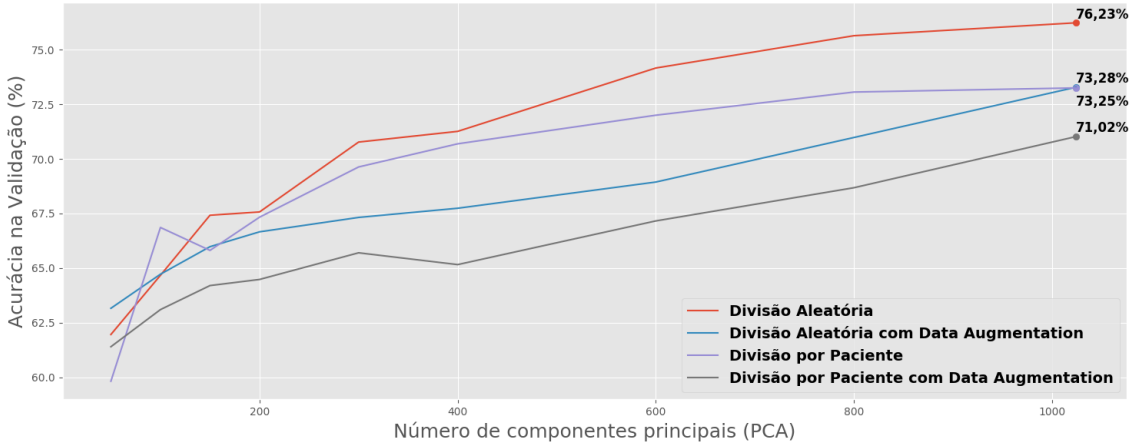
As maiores acurácias foram: 79,28% com 600 componentes principais utilizando as imagens da divisão aleatória, 77,64% com 400 componentes principais utilizando as imagens da divisão aleatória com *data augmentation*, 74,77% com 400 componentes principais utilizando as imagens da divisão por paciente e 74,28% com 300 componentes principais utilizando as imagen da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 69.

Tabela 69 – Desempenho dos classificadores R-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	66,21%	62,66%	80,22%	52,20%	70,36%
Divisão por paciente (sem augm.)		66,48%	64,02%	75,27%	57,69%	69,19%
Divisão aleatória (com augm.)	*rndDiv_test*	74,63%	70,52%	84,66%	64,60%	76,95%
Divisão aleatória (sem augm.)		75,81%	72,26%	83,78%	67,85%	77,59%

Para o classificador utilizando SVM com *kernel* sigmoide (Sig-SVM) a relação entre a quantidade de componentes principais *versus* a acurácia na validação pode ser vista na Figura 68.

Figura 68 – Acurácia do Sig-SVM nos *dataframes* de validação em função da quantidade de componentes principais.



As maiores acurácias foram: 76,23% sem redução de dimensionalidade utilizando as imagem da divisão aleatória, 73,28% sem redução de dimensionalidade com as imagens da divisão aleatória com *data augmentation*, 73,25% sem redução de dimensionalidade com as imagens da divisão por paciente e 71,02% sem redução de dimensionalidade com as imagens da divisão por paciente com *data augmentation*. Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 70.

Tabela 70 – Desempenho dos classificadores Sig-SVM nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	60,99%	59,43%	69,23%	52,75%	63,96%
Divisão por paciente (sem augm.)		63,19%	61,21%	71,98%	54,40%	66,16%
Divisão aleatória (com augm.)	*rndDiv_test*	72,57%	70,73%	76,99%	68,14%	73,73%
Divisão aleatória (sem augm.)		71,24%	70,93%	71,98%	70,50%	71,45%

5.5.2 Classificação utilizando KNN

Para o classificador *K-Nearest Neighbors* (KNN) a relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação, utilizando as imagens da divisão por paciente com *data augmentation*, pode ser consultada na Tabela 71.

Tabela 71 – Acurácia do KNN (em %) com as imagens da divisão por paciente com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
50	72,64	71,50	73,20	72,36	72,98	72,62	73,34	72,32	73,20	72,42	73,12	72,68	73,04	72,66	72,96	72,58
100	73,12	71,80	73,60	72,58	73,40	72,92	73,44	72,96	73,76	73,18	73,30	73,12	73,34	72,88	73,14	73,06
150	73,20	71,98	73,78	72,54	**73,98**	72,74	73,38	73,44	73,94	73,20	74,00	73,54	74,24	73,76	74,30	74,06
200	73,58	71,72	**73,98**	**72,82**	73,96	73,00	73,76	**73,62**	74,14	73,64	74,10	74,08	**74,60**	74,30	74,54	74,30
300	**73,64**	**72,08**	73,62	72,66	73,78	**73,44**	73,92	73,48	74,30	**74,02**	74,22	74,06	74,58	74,36	**74,66**	74,42
400	73,38	71,70	73,66	72,82	73,70	73,22	73,84	73,58	74,24	73,72	**74,28**	74,16	74,46	74,34	74,62	74,44
600	73,32	71,50	73,80	72,80	73,86	73,34	**74,00**	73,52	**74,34**	73,78	74,18	74,18	74,34	74,44	74,52	**74,62**
800	73,26	71,66	73,72	72,72	73,94	73,36	73,98	73,40	74,32	73,84	74,26	**74,34**	74,38	74,42	74,62	74,52
1024	73,26	71,64	73,74	72,62	73,92	73,34	73,98	73,44	74,26	73,86	74,24	74,30	74,42	**74,46**	74,56	74,50

A maior acurácia foi 74,66% e ocorreu com 300 componentes principais e $k = 19$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão da divisão aleatória com *data augmentation* pode ser consultada na Tabela 72.

Tabela 72 – Acurácia do KNN (em %) com as imagens da divisão aleatória com *data augmentation*

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
50	72,70	71,40	72,70	72,26	73,62	72,46	73,58	72,80	74,10	73,08	73,78	73,34	74,18	73,48	74,16	73,60
100	73,96	72,14	74,04	73,20	74,04	73,28	74,60	73,86	74,70	74,10	74,76	74,40	74,78	74,42	74,76	74,40
150	73,92	72,04	74,60	73,68	75,24	73,90	75,26	74,52	75,22	75,02	75,72	74,84	75,58	75,14	75,48	**75,42**
200	74,00	72,10	74,82	73,62	75,40	74,14	75,24	74,60	75,52	75,06	75,66	75,12	75,34	75,30	75,58	75,32
300	**74,06**	72,04	74,80	73,34	75,42	73,98	75,38	**74,70**	**75,90**	74,74	75,66	75,10	**75,74**	75,32	75,68	75,12
400	73,56	71,88	74,74	73,60	75,42	73,86	75,26	74,62	75,68	74,94	75,84	**75,40**	75,66	**75,40**	75,62	75,20
600	73,60	72,00	74,98	73,70	**75,50**	74,30	**75,44**	74,62	75,56	75,10	75,92	75,08	75,38	75,28	**75,70**	75,16
800	73,60	**72,18**	75,10	**73,82**	75,50	**74,34**	75,40	74,60	75,56	**75,18**	75,88	75,12	75,42	75,20	75,66	75,12
1024	73,56	72,12	**75,18**	73,82	75,42	74,30	75,38	74,64	75,62	75,18	**75,98**	75,06	75,38	75,16	75,62	75,12

A maior acurácia foi 75,98% e ocorreu sem redução de dimensionalidade e $k = 15$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão por paciente pode ser consultada na Tabela 73.

Tabela 73 – Acurácia do KNN (em %) com as imagens da divisão por paciente

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
50	72,27	69,89	71,21	71,08	71,28	71,01	71,15	70,82	71,94	71,21	72,00	70,88	71,54	71,48	71,74	71,74
100	72,13	71,15	71,61	71,01	71,87	71,15	72,13	71,67	71,74	70,75	71,74	71,08	71,87	71,34	71,48	71,48
150	**72,86**	71,08	72,20	71,41	72,73	**72,07**	**73,39**	**72,07**	72,33	71,54	72,13	71,87	72,66	72,27	72,46	72,33
200	72,40	71,41	72,92	**71,54**	**72,86**	71,94	72,60	71,41	72,53	71,87	72,40	72,13	72,46	72,20	**72,60**	72,27
300	71,67	71,48	72,86	71,41	72,33	71,41	72,53	71,48	72,40	71,67	72,27	**72,60**	**72,79**	**72,40**	72,60	**72,53**
400	71,87	71,34	**73,06**	71,54	72,40	71,54	72,13	72,00	**72,66**	**72,20**	72,66	72,46	72,79	72,27	72,53	72,33
600	71,81	**71,61**	73,06	71,54	72,40	71,67	72,20	72,07	72,60	72,20	72,66	72,53	72,79	72,20	72,60	72,33
800	71,81	71,61	73,06	71,54	72,40	71,67	72,20	72,07	72,60	72,20	**72,73**	72,53	72,79	72,20	72,60	72,33
1024	71,81	71,61	73,06	71,54	72,40	71,67	72,20	72,07	72,60	72,20	72,73	72,53	72,79	72,20	72,60	72,33

A maior acurácia foi 73,39% e ocorreu com 150 componentes principais e $k = 11$.

A relação entre quantidade de vizinhos *versus* quantidade de componentes principais *versus* acurácia na validação com o classificador utilizando as imagens da divisão aleatória pode ser consultada na Tabela 74.

Tabela 74 – Acurácia do KNN (em %) com as imagens da divisão aleatória

PCA	k=5	k=6	k=7	k=8	k=9	k=10	k=11	k=12	k=13	k=14	k=15	k=16	k=17	k=18	k=19	k=20
50	74,61	72,64	74,16	74,02	74,16	74,16	74,61	73,57	74,80	74,06	74,75	73,52	74,31	73,67	74,46	73,72
100	75,20	73,87	75,89	**75,69**	75,64	75,64	75,44	75,30	75,79	75,10	75,74	75,69	76,28	76,18	76,57	**77,07**
150	**75,79**	**74,31**	**76,13**	75,10	76,13	75,25	76,08	75,69	76,18	**76,53**	76,77	76,23	76,53	76,13	**76,92**	76,38
200	75,54	73,82	75,79	75,49	**76,48**	**76,03**	**76,72**	76,03	**76,92**	76,33	76,82	76,18	76,38	**76,43**	76,67	76,33
300	74,80	73,52	75,10	75,05	76,28	75,79	76,57	**76,33**	76,43	75,74	76,82	76,38	**76,62**	76,08	76,67	76,13
400	74,95	73,33	75,00	74,56	75,94	75,59	76,48	75,69	76,33	75,89	**77,17**	76,33	76,48	76,23	76,43	76,03
600	75,10	73,28	74,95	74,66	75,94	75,49	76,57	75,74	76,33	75,94	77,02	76,38	76,23	76,23	76,53	76,08
800	75,10	73,28	74,95	74,66	75,94	75,49	76,57	75,69	76,33	75,94	77,02	**76,43**	76,23	76,28	76,53	76,08
1024	75,10	73,28	74,95	74,66	75,94	75,49	76,57	75,69	76,33	75,94	77,02	76,43	76,23	76,28	76,53	76,08

A maior acurácia foi 77,17% e ocorreu com 400 componentes principais e $k = 15$.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens do conjuntos de teste, obtêm-se os resultados da Tabela 75.

Tabela 75 – Desempenho dos classificadores KNN nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	65,38%	62,28%	78,02%	52,75%	69,27%
Divisão por paciente (sem augm.)		64,84%	62,98%	71,98%	57,69%	67,18%
Divisão aleatória (com augm.)	*rndDiv_test*	74,48%	72,68%	78,47%	70,50%	75,46%
Divisão aleatória (sem augm.)		74,34%	73,64%	75,81%	72,86%	74,71%

5.5.3 Classificação utilizando Floresta Aleatória

Para o classificador Floresta Aleatória (RF) a relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente com *data augmentation* pode ser consultada na Tabela 76.

Tabela 76 – Acurácia do RF com as imagens da divisão por paciente com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
50	**74,28**%	**74,72**%	74,26%	74,30%	74,38%	74,50%
100	74,06%	74,18%	**74,32**%	74,24%	**74,58**%	**74,60**%
150	74,08%	74,16%	73,70%	**74,46**%	74,04%	73,92%
200	73,92%	73,52%	74,10%	73,98%	73,48%	73,60%
300	72,90%	73,14%	72,88%	73,12%	73,64%	73,34%
400	72,74%	72,94%	72,54%	72,88%	72,66%	73,62%
600	72,70%	72,20%	73,02%	72,98%	72,84%	73,04%
800	72,50%	72,84%	72,48%	73,36%	72,90%	72,76%
1024	72,88%	72,88%	73,26%	72,70%	72,86%	73,28%

A maior acurácia foi 74,72% e ocorreu com 50 componentes principais e 70 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória com *data augmentation* pode ser consultada na Tabela 77.

Tabela 77 – Acurácia do RF com as imagens da divisão aleatória com *data augmentation*

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
50	**77,94**%	**78,00**%	**78,04**%	**78,28**%	**78,62**%	**78,28**%
100	77,34%	77,48%	77,60%	78,10%	78,48%	77,92%
150	77,08%	77,64%	77,36%	77,74%	77,58%	78,00%
200	76,70%	77,10%	77,10%	77,08%	77,64%	77,64%
300	76,28%	76,54%	76,58%	76,04%	76,90%	76,92%
400	75,42%	76,46%	76,46%	76,52%	76,62%	76,14%
600	74,80%	75,56%	76,02%	76,42%	76,18%	75,90%
800	76,10%	75,22%	75,44%	76,54%	76,12%	76,06%
1024	75,04%	75,64%	76,08%	75,70%	76,54%	75,98%

A maior acurácia foi 78,62% e ocorreu com 50 componentes principais e 130 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão por paciente pode ser consultada na Tabela 78.

Tabela 78 – Acurácia do RF com as imagens da divisão por paciente

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
50	**75,96**%	75,69%	75,89%	**76,09**%	75,43%	75,96%
100	75,63%	**76,02**%	**75,96**%	75,82%	**76,48**%	**76,35**%
150	75,43%	75,30%	75,03%	75,63%	76,09%	75,63%
200	73,98%	75,63%	75,56%	76,02%	75,96%	75,03%
300	74,44%	74,64%	74,77%	75,10%	75,16%	75,89%
400	74,37%	74,31%	75,03%	74,90%	74,70%	74,77%
600	73,39%	74,44%	73,98%	75,43%	74,24%	74,24%
800	73,06%	74,77%	74,44%	74,18%	74,04%	73,45%
1024	73,32%	73,65%	74,31%	73,19%	74,04%	73,78%

A maior acurácia foi 76,48% e ocorreu com 100 componentes principais e 130 árvores aleatórias.

A relação entre a quantidade de componentes principais *versus* quantidade de árvores *versus* acurácia utilizando as imagens da divisão aleatória pode ser consultada na Tabela 79.

Tabela 79 – Acurácia do RF com as imagens da divisão aleatória

PCA	50 árvores	70 árvores	90 árvores	110 árvores	130 árvores	150 árvores
50	**79,28**%	**79,82**%	79,92%	**80,12**%	**80,46**%	**80,41**%
100	79,28%	79,58%	**80,07**%	78,89%	79,87%	79,18%
150	78,40%	79,33%	79,28%	79,23%	78,89%	79,28%
200	78,20%	78,05%	78,49%	78,25%	78,69%	78,49%
300	77,36%	78,44%	77,41%	77,56%	77,81%	77,81%
400	76,43%	76,33%	77,66%	77,17%	77,41%	76,87%
600	76,67%	76,62%	75,89%	76,13%	76,77%	76,72%
800	75,54%	75,44%	75,30%	76,92%	75,25%	75,54%
1024	73,33%	73,08%	74,51%	74,80%	74,61%	74,46%

A maior acurácia foi 80,46% e ocorreu com 50 componentes principais e 130 árvores aleatórias.

Aplicando estes classificadores, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 80.

Tabela 80 – Desempenho dos classificadores RF nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	70,05%	67,80%	76,37%	63,74%	71,83%
Divisão por paciente (sem augm.)		68,13%	66,50%	73,08%	63,19%	69,63%
Divisão aleatória (com augm.)	*rndDiv_test*	75,37%	73,63%	79,06%	71,68%	76,25%
Divisão aleatória (sem augm.)		75,81%	73,46%	80,83%	70,80%	76,97%

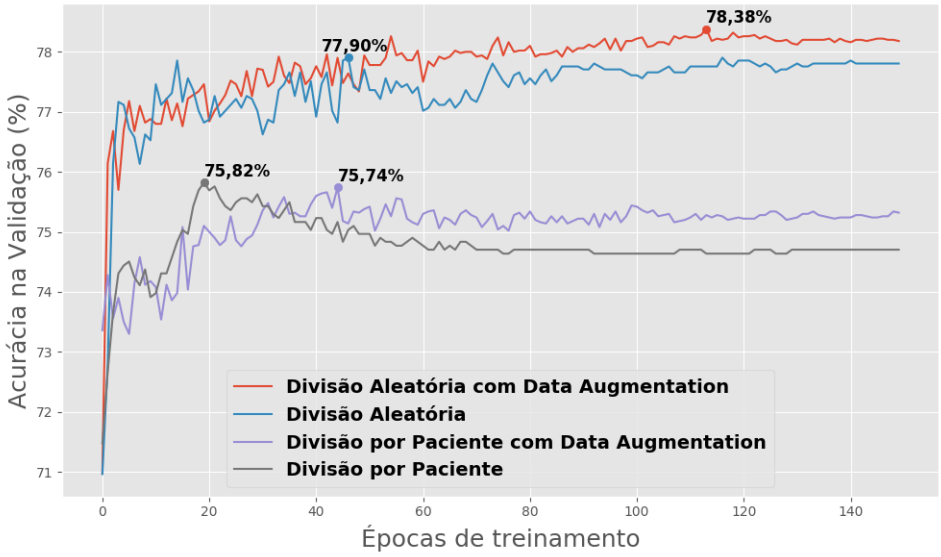
5.5.4 Classificação utilizando Redes Neurais

Das arquiteturas de Redes Neurais avaliadas, as que obtiveram as maiores acurácias foram:

- 4 camadas, cada qual com 512 neurônios, que utilizam tangente hiperbólica como função de ativação, para as imagens da divisão por Paciente com *data augmentation*;
- 4 camadas, cada qual com 1024, neurônios que utilizam sigmoide como função de ativação, Para as imagens da divisão aleatória com *data augmentation*;
- 3 camadas, cada qual com 32, neurônios que utilizam ReLU como função de ativação, para as imagens da divisão por paciente;
- 2 camadas, cada qual com 512, neurônios que utilizam ReLU como função de ativação, para as imagens da divisão aleatória;

A acurácia na validação em função do número de épocas de treinamento de cada uma destas redes pode ser vista na Figura 69.

Figura 69 – Acurácia das Redes Neurais dos *dataframes* de validação em função das épocas de treinamento



As maiores acurácias foram: 78,38% com as imagens da divisão aleatória com *data augmentation*, 77,90% com as imagens da divisão aleatória, 75,82% com as imagens da divisão por paciente e 75,74% com as imagens da divisão por paciente com *data augmentation*.

Aplicando estas redes neurais, que obtiveram as maiores acurácias nos conjuntos de validação, nas imagens dos conjuntos de teste, obtêm-se os resultados da Tabela 81.

Tabela 81 – Desempenho das Redes Neurais nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	69,78%	67,14%	77,47%	62,09%	71,94%
Divisão por paciente (sem augm.)		64,56%	64,02%	66,48%	62,64%	65,23%
Divisão aleatória (com augm.)	*rndDiv_test*	76,84%	74,73%	81,12%	72,57%	77,79%
Divisão aleatória (sem augm.)		73,75%	72,42%	76,70%	70,80%	74,50%

5.6 Redes Convolucionais

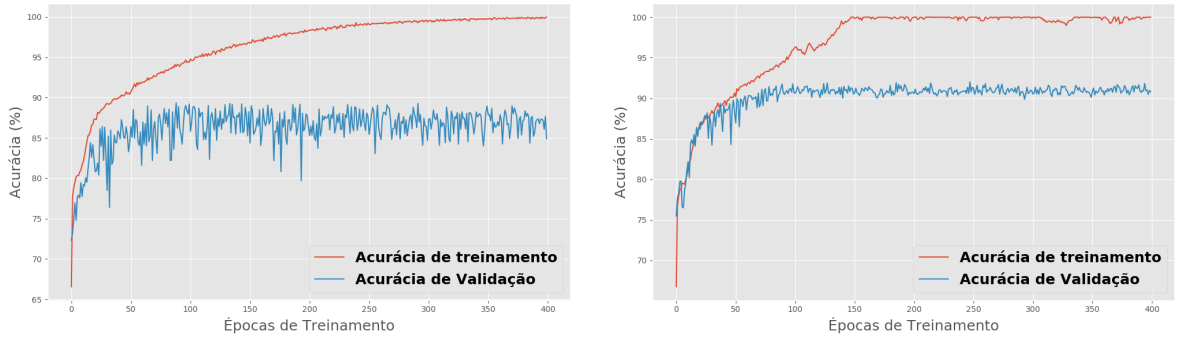
As redes arquiteturas *Xception*, VGG16, VGG19, *ResNet50* e *InceptionV3* foram treinadas na máquina virtual da *Google Cloud Plataform* com a placa de vídeo Tesla T4 da Nvidia por 100 épocas, neste treinamento foram ajustados os hiper-parâmetros de cada rede e suas respectivas camadas totalmente conectadas para que se obtivessem as maiores acurácias nos conjuntos de validação. A acurácia obtida por cada rede após este treinamento pode ser consultada na Tabela 82.

Tabela 82 – Desempenho das Redes Convolucionais nos conjuntos de Validação

Arquitetura	Acurácia (c/ imagens da Divisão por Paciente)	Acurácia (c/ imagens da Divisão Aleatória)
Xception	84,62%	89,90%
VGG16	**89,62%**	91,98%
VGG19	88,40%	**92,78%**
ResNet50	77,98%	79,90%
InceptionV3	78,74%	81,40%

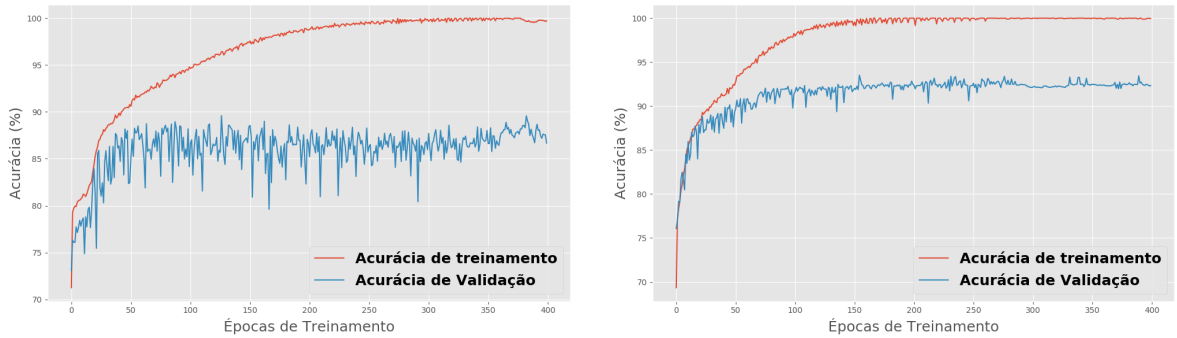
As redes que obtiveram os maiores resultados foram: *Xception*, VGG16, VGG19. Estas redes foram treinadas novamente, porém por 400 épocas, isto foi feito para que obtivessem a maior acurácia possível. A evolução da aprendizagem destas redes em cada conjunto de treinamento pode ser visto nas Figuras 70, 71 e 72.

Figura 70 – Treinamento por 400 épocas da VGG16



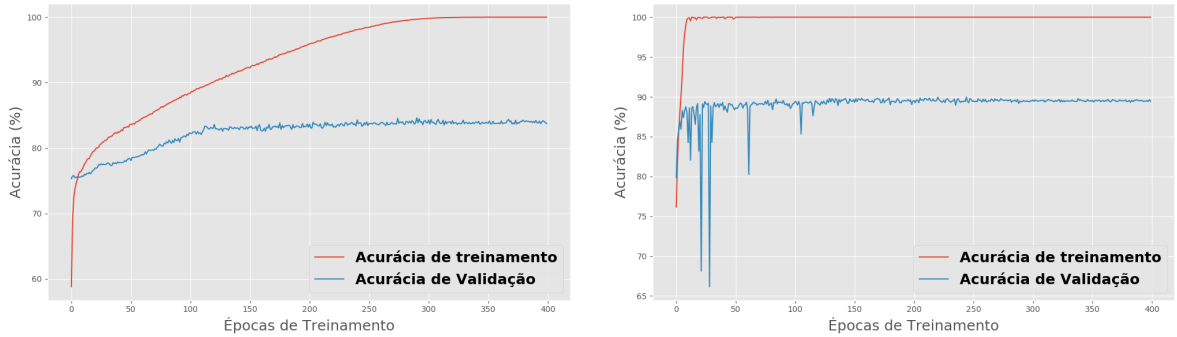
(a) Treinamento c/ imagens da Divisão por Paciente (b) Treinamento c/ imagens da Divisão Aleatória

Figura 71 – Treinamento por 400 épocas da VGG19



(a) Treinamento c/ imagens da Divisão por Paciente (b) Treinamento c/ imagens da Divisão Aleatória

Figura 72 – Treinamento por 400 épocas da *Xception*



(a) Treinamento c/ imagens da Divisão por Paciente (b) Treinamento c/ imagens da Divisão Aleatória

Após treinadas por 400 épocas, as redes convolucionais obtiveram as seguintes acurácias nos seus respectivos conjuntos de validação: a *Xception* obteve a acurácia de 87,42% quando treinada com as imagens da divisão por paciente e 89,96% quando treinada com as imagens da divisão aleatória; a VGG16 obteve a acurácia de 89,34% quando treinada com as imagens da divisão por paciente e 92,50% quando treinada com as imagens da divisão aleatória; e por fim a

VGG19 obteve a acurácia de 89,12% quando treinada com as imagens da divisão por paciente e 92,74% quando treinada com as imagens da divisão aleatória. Estes classificadores que obtiveram as maiores acurácias foram submetidos as imagens do conjunto de teste, e seus desempenhos nestes conjuntos podem ser consultados nas Tabelas 83, 84 e 85.

Tabela 83 – Desempenho da Rede *Xception* nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	77,47%	70,33%	95,05%	59,89%	80,84%
Divisão aleatória (com augm.)	*rndDiv_test*	90,41%	87,64%	94,10%	86,73%	90,76%

Tabela 84 – Desempenho da Rede VGG16 nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	75,00%	68,13%	93,96%	56,04%	78,99%
Divisão aleatória (com augm.)	*rndDiv_test*	92,48%	91,14%	94,10%	90,86%	92,60%

Tabela 85 – Desempenho da Rede VGG19 nos conjuntos de teste

Conj.Treino	Conj.Teste	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	*patLvDiv_test*	73,90%	66,42%	96,70%	51,10%	78,75%
Divisão aleatória (com augm.)	*rndDiv_test*	91,59%	90,06%	93,51%	89,68%	91,75%

5.7 Resumo dos resultados

Com relação as **estatísticas de 1ª ordem na Divisão por Paciente**, a maior acurácia no conjunto de teste *patLvDiv_test* foi de 81,04%, obtida com uma rede neural (2 camadas, cada qual com 1024 neurônios que utilizam a tangente hiperbólica como função de ativação) treinada com as imagens do conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 3,57% na acurácia do classificador em comparação com a rede treinada com as imagens do conjunto *patLvDiv_train*. A segunda maior acurácia, 76,37%, foi obtida pelo classificador SVM com *kernel* RBF e 75 componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 7,96% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A terceira maior acurácia, 72,80%, foi obtida pelo classificador RF ao utilizar 150 árvores aleatórias e 75 componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 0,82% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A quarta maior acurácia, 69,78%, foi obtida pelo classificador KNN ao utilizar k=7 e 20 componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 7,42% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*.

Com relação as **estatísticas de 1ª ordem na Divisão Aleatória**, a maior acurácia no conjunto de teste *rndDiv_test* foi de 89,53%, obtida com uma rede neural (3 camadas, cada qual

com 1024 neurônios que utilizam a ReLU como função de ativação) treinada com as imagens do conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 0,44% na acurácia do classificador em comparação com a rede treinada com as imagens do conjunto *rndDiv_train*. A segunda maior acurácia, 87,02%, foi obtida pelo classificador SVM ao utilizar o *kernel* RBF e 45 componentes principais, treinado com o conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 1,77% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*. A terceira maior acurácia, 84,66%, foi obtida pelo classificador RF ao utilizar 150 árvores aleatórias e 100 componentes principais, treinado com o conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 0,44% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*. A quarta maior acurácia, 82,74%, foi obtida pelo classificador KNN ao utilizar k=6 e 85 componentes principais, treinado com o conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 3,98% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*.

O resumo destes resultados pode ser consultado na Tabela 86.

Tabela 86 – Resumo dos resultados obtidos com as imagens dos conjuntos de teste pelos classificadores que utilizaram as estatísticas de 1ª ordem

	Classificador	Acurácia	Quantidade de Componentes Principais	Aumento da Acurácia introduzido pelo *data augmentation*
Divisão por Paciente	Rede Neural	81,04%	N/A	+3,57%
	R-SVM	76,37%	75	+7,96%
	RF	72,80%	75	+0,82%
	KNN	69,78%	20	+7,42%
Divisão Aleatória	Rede Neural	89,53%	N/A	+0,44%
	R-SVM	87,02%	45	+1,77%
	RF	84,66%	100	+0,44%
	KNN	82,74%	85	+3,98%

Com relação as **características de textura na Divisão por Paciente**, a maior acurácia no conjunto de teste *patLvDiv_test* foi de 67,86%, obtida pelo classificador RF ao utilizar 70 árvores aleatórias, sem redução de componentes principais, treinado com as imagens do conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 2,75% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A segunda maior acurácia, 65,93%, foi obtida pelo classificador KNN ao utilizar k=7 e 35 componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 7,41% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A terceira maior acurácia, 65,11%, foi obtida pelo classificador SVM ao utilizar o *kernel* polinomial e 40 componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de

data augmentation produziu um aumento de 4,12% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A quarta maior acurácia, 64,56%, foi obtida com uma rede neural (3 camadas, cada qual com 128 neurônios que utilizam ReLU como função de ativação) treinada com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 3,02% na acurácia do classificador em comparação com a rede treinada com as imagens do conjunto *patLvDiv_train*.

Com relação às **características de textura na Divisão Aleatória**, a maior acurácia no conjunto de teste *rndDiv_test* foi de 78,47%, obtida pelo classificador RF ao utilizar 150 árvores aleatórias e 70 componentes principais, treinado com as imagens do conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 1,33% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*. A segunda maior acurácia, 78,02%, foi obtida com uma rede neural (2 camadas, cada qual com 1024 neurônios que utilizam a ReLU como função de ativação) treinada com o conjunto *rndDiv_train*, o procedimento de *data augmentation* produziu uma redução de 1,62% na acurácia do classificador treinado com o conjunto *augm_rndDiv_train*. A terceira maior acurácia, 76,70%, foi obtida pelo classificador SVM ao utilizar o *kernel* polinomial e 25 componentes principais, treinado com as imagens do conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 1,48% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*. A quarta maior acurácia, 74,78%, foi obtida pelo classificador KNN ao utilizar k=18 com 30 componentes principais, treinado com as imagens do conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 0,74% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*.

O resumo destes resultados pode ser consultado na Tabela 87.

Tabela 87 – Resumo dos resultados obtidos com as imagens dos conjuntos de teste pelos classificadores que utilizaram as características de textura

	Classificador	Acurácia	Quantidade de Componentes Principais	Aumento da Acurácia introduzido pelo *data augmentation*
Divisão por Paciente	RF	67,86%	75	+2,75%
	KNN	65,93%	35	+7,41%
	P-SVM	65,11%	40	+4,12%
	Rede Neural	64,56%	N/A	+3,02%
Divisão Aleatória	RF	78,47%	70	+1,33%
	Rede Neural	78,02%	N/A	0%
	P-SVM	76,70%	25	+1,48%
	KNN	74,78%	30	+0,74%

Com relação as **características morfológicas na Divisão por Paciente**, a maior acurácia no conjunto de teste *patLvDiv_test* foi de 70,88%, obtida pelo classificador RF ao utilizar

110 árvores aleatórias sem redução de componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 2,20% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A segunda maior acurácia, 69,78%, foi obtida pelo classificador SVM com *kernel* linear sem redução de componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 2,20% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A terceira maior acurácia, 69,23%, foi obtida pelo classificador KNN ao utilizar k=5 e 10 componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 5,77% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A quarta maior acurácia, 68,68%, foi obtida com uma rede neural (4 camadas, cada qual com 128 neurônios que utilizam a tangente hiperbólica como função de ativação) treinada com as imagens do conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 0,55% na acurácia do classificador em comparação com a rede treinada com as imagens do conjunto *patLvDiv_train*.

Com relação as **características morfológicas na Divisão Aleatória**, a maior acurácia no conjunto de teste *rndDiv_test* foi de 80,24%, obtida com uma rede neural (4 camadas, cada qual com 1024 neurônios que utilizam a ReLU como função de ativação) treinada com as imagens do conjunto *rndDiv_train*, o procedimento de *data augmentation* produziu uma redução de 0,74% na acurácia da rede treinada com as imagens do conjunto *augm_rndDiv_train*. A segunda maior acurácia, 79,35%, foi obtida pelo classificador RF ao utilizar 70 árvores aleatórias sem redução de componentes principais, treinado com o conjunto *rndDiv_train*, o procedimento de *data augmentation* produziu uma redução de 2,36% na acurácia do classificador treinado com as imagens do conjunto *augm_rndDiv_train*. A terceira maior acurácia, 77,14%, foi obtida pelo classificador SVM ao utilizar o *kernel* RBF e 10 componentes principais, treinado com o conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 1,03% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*. A quarta maior acurácia, 75,81%, foi obtida pelo classificador KNN ao utilizar k=20 e 10 componentes principais, treinado com o conjunto *rndDiv_train*, o procedimento de *data augmentation* produziu uma redução de 2,21% na acurácia do classificador treinado com as imagens do conjunto *augm_rndDiv_train*.

O resumo destes resultados pode ser consultado na Tabela 88.

Tabela 88 – Resumo dos resultados obtidos com as imagens dos conjuntos de teste pelos classificadores que utilizaram as características morfológicas

	Classificador	Acurácia	Quantidade de Componentes Principais	Aumento da Acurácia introduzido pelo *data augmentation*
Divisão por Paciente	RF	70,88%	20	+2,20%
	L-SVM	69,78%	20	+2,20%
	KNN	69,23%	10	+5,77%
	Redes Neurais	68,68%	N/A	+0,55%
Divisão Aleatória	Rede Neural	80,24%	N/A	0%
	RF	79,35%	20	0%
	R-SVM	77,14%	10	+1,03%
	KNN	75,81%	10	0%

Com relação as **características de contorno na Divisão por Paciente**, a maior acurácia no conjunto de teste *patLvDiv_test* foi de 62,91%, obtida pelo classificador SVM com *kernel* polinomial e 130 componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 1,37% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A segunda maior acurácia, 62,09%, foi obtida com uma rede neural (2 camadas, cada qual com 1024 neurônios que utilizam sigmoide como função de ativação) treinada com as imagens do conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 1,93% na acurácia do classificador em comparação com a rede treinada com as imagens do conjunto *patLvDiv_train*. A terceira maior acurácia, 60,99%, foi obtida pelo classificador KNN ao utilizar k=19 e 10 componentes principais, treinado com o conjunto *patLvDiv_train*, o procedimento de *data augmentation* produziu uma redução de 3,02% na acurácia do classificador treinado com as imagens do conjunto *augm_patLvDiv_train*. A quarta maior acurácia, 58,79%, foi obtida pelo classificador RF ao utilizar 70 árvores aleatórias sem redução de componentes principais, treinado com o conjunto *patLvDiv_train*, o procedimento de *data augmentation* não produziu nenhuma alteração em comparação com o classificador treinado com as imagens do conjunto *augm_patLvDiv_train*.

Com relação as **características de contorno na Divisão Aleatória**, a maior acurácia no conjunto de teste *rndDiv_test* foi de 75,07%, obtida com uma rede neural (3 camadas, cada qual com 512 neurônios que utilizam a sigmoide como função de ativação) treinada com as imagens do conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 5,60% na acurácia do classificador em comparação com a rede treinada com as imagens do conjunto *rndDiv_train*. A segunda maior acurácia, 74,04%, foi obtida pelo classificador SVM ao utilizar o *kernel* linear sem redução componentes principais, treinado com o conjunto *rndDiv_train*, o procedimento de *data augmentation* produziu uma redução de 0,44% na acurácia do classificador treinado com as imagens do conjunto *augm_rndDiv_train*. A terceira maior acurácia, 71,98%, foi obtida pelo classificador RF ao utilizar 150 árvores aleatórias e 120

componentes principais, treinado com o conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 0,89% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*. A quarta maior acurácia, 70,65%, foi obtida pelo classificador KNN ao utilizar k=19 e 150 componentes principais, treinado com o conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 1,18% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*.

O resumo destes resultados pode ser consultado na Tabela 89.

Tabela 89 – Resumo dos resultados obtidos com as imagens dos conjuntos de teste pelos classificadores que utilizaram as características de contorno

	Classificador	Acurácia	Quantidade de Componentes Principais	Aumento da Acurácia introduzido pelo *data augmentation*
Divisão por Paciente	P-SVM	62,91%	130	+1,37%
	Rede Neural	62,09%	N/A	+1,93%
	KNN	60,99%	10	0%
	RF	58,79%	160	0%
Divisão Aleatória	Rede Neural	75,07%	N/A	+5,60%
	L-SVM	74,04%	160	+1,77%
	RF	71,98%	120	+0,89%
	KNN	70,65%	150	+1,18%

Com relação as **características da transformada discreta do cosseno na Divisão por Paciente**, a maior acurácia no conjunto de teste *patLvDiv_test* foi de 70,05%, obtida pelo classificador RF ao utilizar 130 árvores aleatórias e 50 componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 1,92% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*. A segunda maior acurácia, 69,78%, foi obtida com uma rede neural (4 camadas, cada qual com 512 neurônios que utilizam a tangente hiperbólica como função de ativação) treinada com as imagens do conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 5,22% na acurácia do classificador em comparação com a rede treinada com as imagens do conjunto *patLvDiv_train*. A terceira maior acurácia, 67,03%, foi obtida pelo classificador SVM com *kernel* polinomial e 600 componentes principais, treinado com o conjunto *patLvDiv_train*, o procedimento de *data augmentation* não produziu alteração em comparação com o classificador treinado com as imagens do conjunto *augm_patLvDiv_train*. A quarta maior acurácia, 65,38%, foi obtida pelo classificador KNN ao utilizar k=15 sem redução de componentes principais, treinado com o conjunto *augm_patLvDiv_train*, o procedimento de *data augmentation* produziu um aumento de 0,54% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *patLvDiv_train*.

Com relação às **características da transformada discreta do cosseno na Divisão**

Aleatória, a maior acurácia no conjunto de teste *rndDiv_test* foi de 76,74%, obtida com uma rede neural (4 camadas, cada qual com 1024 neurônios que utilizam sigmoide como função de ativação) treinada com as imagens do conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 3,09% na acurácia do classificador em comparação com a rede treinada com as imagens do conjunto *rndDiv_train*. A segunda maior acurácia, 75,81%, foi obtida pelo classificador RF ao utilizar 130 árvores aleatórias e 50 componentes principais, treinado com o conjunto *rndDiv_train*, o procedimento de *data augmentation* produziu uma redução de 0,44% na acurácia do classificador treinado com as imagens do conjunto *augm_rndDiv_train*. A terceira maior acurácia, 75,81%, foi obtida pelo classificador SVM ao utilizar o *kernel* RBF e 600 componentes principais, treinado com o conjunto *rndDiv_train*, o procedimento de *data augmentation* produziu uma redução de 1,18% na acurácia do classificador treinado com as imagens do conjunto *rndDiv_train*. A quarta maior acurácia, 74,48%, foi obtida pelo classificador KNN ao utilizar k=15 sem redução de componentes principais, treinado com o conjunto *augm_rndDiv_train*, o procedimento de *data augmentation* produziu um aumento de 0,14% na acurácia do classificador em comparação com o classificador treinado com as imagens do conjunto *rndDiv_train*.

O resumo destes resultados pode ser consultado na Tabela 90.

Tabela 90 – Resumo dos resultados obtidos com as imagens dos conjuntos de teste pelos classificadores que utilizaram as características da transformada discreta do cosseno

	Classificador	Acurácia	Quantidade de Componentes Principais	Aumento da Acurácia introduzido pelo *data augmentation*
Divisão por Paciente	RF	70,05%	50	+1,92%
	Rede Neural	69,78%	N/A	+5,22%
	P-SVM	67,03%	600	0%
	KNN	65,38%	1024	+0,54%
Divisão Aleatória	Rede Neural	76,74%	N/A	+3,09%
	RF	75,81%	50	0%
	R-SVM	75,81%	600	0%
	KNN	74,48%	1024	+3,98%

Com relação as **redes convolucionais**, a maior acurácia no conjunto de teste *patLvDiv_test* foi de 77,47%, obtida pela *Xception* treinada com as imagens da divisão por paciente com *data augmentation*. Com relação ao conjunto de teste *rndDiv_test* a maior acurácia foi de 92,48%, obtida pela VGG16. Como só foram utilizadas imagens com *data augmentation* não é possível inferir a influência ou não do uso desta técnica na acurácia das redes, a mesma lógica vale para a técnica do PCA.

As maiores acurácias para cada um dos métodos testados com as imagens da divisão por paciente estão presentes no Quadro 4, os resultados para os métodos testados com as imagens da divisão aleatória estão presentes no Quadro 5.

Quadro 4 – Resultados obtidos com a Divisão por Paciente

Metódo de Classificação	**Classificador**	**Acurácia**	***Data Augmentation***	**Redução com PCA**
Estatísticas de 1ª ordem	Rede Neural	81,04%	Sim	N/A
Redes Convolucionais	N/A	77,47%	Sim	N/A
Características Morfológicas	Floresta Aleatória	70,88%	Sim	Não
Características da Transformada Discreta do Cosseno	Floresta aleatória	70,05%	Sim	Sim
Características de Textura	Floresta Aleatória	67,86%	Sim	Não
Características de Contorno	P-SVM	62,91%	Sim	Sim

Quadro 5 – Resultados obtidos com a Divisão Aleatória

Método de Classificação	**Classificador**	**Acurácia**	***Data Augmentation***	**Redução com PCA**
Redes Convolucionais	N/A	92,48%	Sim	N/A
Estatísticas de 1ª ordem	Rede Neural	89,53%	Sim	N/A
Características Morfológicas	Rede Neural	80,24%	Não	N/A
Características de Textura	Floresta Aleatória	78,47%	Sim	Sim
Características da Transformada Discreta do Cosseno	Rede Neural	76,74%	Sim	N/A
Características de Contorno	Rede Neural	75,07%	Sim	N/A

No próximo capítulo serão apresentadas as conclusões obtidas a partir da análise dos resultados.

6
Conclusões

O objetivo de dividir os conjuntos de treino, validação e teste de duas formas diferentes (Divisão Aleatória e Divisão por Paciente) foi para avaliar a afirmação feita por Mourya et al. (2018), que diz que as variações celulares particulares de cada paciente influenciam o desempenho do classificador. Os classificadores treinados com as imagens da Divisão Aleatória obtiveram, em média, uma acurácia $10,40\% \pm 2,69\%$ maior quando comparado com os classificadores treinados com as imagens da Divisão por Paciente.

Houveram exemplares das células de cada um dos 73 pacientes (47 doentes e 26 saudáveis) dentro dos conjuntos de treinamento da Divisão Aleatória, ou seja, durante o processo de treino os classificadores foram expostos às variações celulares particulares de cada um destes 73 pacientes, o mesmo não ocorreu com os classificadores da Divisão por Paciente. Dos 47 pacientes acometidos por LLA, apenas as células de 29 deles foram utilizadas para o treinamento dos classificadores, e dos 26 pacientes saudáveis, apenas as células de 16 foram utilizadas no treino.

Com relação à quantidade de amostras positivas (células malignas) e negativas (células saudáveis) dentro dos conjuntos de treinamento, em números absolutos, 4364 linfócitos malignos e 2181 linfócitos saudáveis distintos foram utilizados para o treino na Divisão Aleatória e 4776 linfócitos malignos e 2128 linfócitos saudáveis foram utilizados para treino na Divisão por Paciente. Posteriormente as quantidades de amostras positivas e negativas foram igualadas para 10000 linfócitos malignos e 10000 linfócitos saudáveis com a técnica de *Data Augmentation* nos dois conjuntos de treinamento.

Como não há grande diferença na quantidade de amostras positivas e negativas entre as duas Divisões e que a diferença entre elas está no fato de que os conjuntos de treinamentos da Divisão por Paciente não contêm os exemplares de células de todos os pacientes, pode-se chegar à conclusão feita por Mourya et al. (2018) de que as variações celulares existentes entre as células dos pacientes influenciam a eficiência do classificador.

Logo, a presença das células de todos paciente nos conjuntos de treino fez com que os

classificadores da Divisão Aleatória, expostos às variações individuais das células de todos os pacientes, obtivessem maior acurácia do que aqueles expostos a uma quantidade limitada de pacientes.

Com relação à técnica de *Data Augmentation* foi observado um aumento, em média, de $2,79\% \pm 2,17\%$ na acurácia dos classificadores que foram treinados com as imagens produzidas por esta técnica. A exceção foi o classificador que utiliza as características morfológicas, o qual obteve a maior acurácia no conjunto de teste quando treinado com as imagens da Divisão Aleatória sem *Data Augmentation*. Entretanto o classificador que utiliza as características morfológicas da divisão por paciente obteve aumento da acurácia quando utilizou as imagens com *Data Augmentation*. Esta disparidade de resultados pode ter sido resultado das técnicas empregadas na geração das novas imagem que podem não ter sido as mais adequadas para serem usadas neste tipo de característica. De qualquer forma, a técnica de *Data Augmentation* esteve presente nas imagens utilizadas no treinamento dos melhores os classificadores.

Durante o treinamento dos classificadores a redução de dimensionalidade utilizando PCA produziu aumentos na acurácia, entretanto as maiores acurácias alcançadas não utilizaram esta técnica. Todavia vale lembrar que tanto as redes neurais como as redes neurais convolucionais (os métodos que obtiveram as maiores acurácias) utilizam variações do algoritmo de *Backpropagation* para o aprendizado de seus parâmetros internos, e este algoritmo apresenta propriedades de redução de dimensionalidade em conjuntos de dados (TAN; MAYROVOUNIOTIS, 1995; HINTON, 2006).

Dos conjuntos de características extraídas das imagens, as que obtiveram os melhores resultados quando utilizadas pelos classificadores foram, em ordem da maior acurácia até a menor: as estatísticas de 1ª ordem, as características morfológicas, as características de textura, as características DCT e as características de contorno.

Dos classificadores treinados com as características extraídas das imagens da divisão por paciente, aqueles que utilizaram as estatísticas de 1ª ordem obtiveram, em média, a acurácia de $75,00\% \pm 4,20\%$, os que utilizaram as características morfológicas obtiveram $69,64\% \pm 0,81\%$, os que utilizaram as características DCT obtiveram $68,06\% \pm 1,95\%$, os que utilizaram as características de textura obtiveram $65,87\% \pm 1,25\%$ e aqueles que utilizaram as características de contorno obtiveram $61,92\% \pm 1,55\%$.

Dos classificadores treinados com as características extraídas das imagens da divisão aleatória, aqueles que utilizaram as estatísticas de 1ª ordem obtiveram, em média, a acurácia de $85,99\% \pm 2,55\%$, os que utilizaram as características morfológicas obtiveram $78,13\% \pm 1,75\%$, os que utilizaram as características de textura obtiveram $76,99\% \pm 1,43\%$, os que utilizaram as características da DCT obtiveram $75,71\% \pm 0,81\%$ e aqueles que utilizaram as características de contorno obtiveram $72,94\% \pm 1,73\%$.

Quanto aos algoritmos de aprendizagem de máquina, as Redes Neurais obtiveram os

melhores resultados, sendo que este foi o melhor classificador para as características de 1ª ordem tanto na divisão por paciente como na divisão aleatória. As maiores acurácias obtidas por este método de classificação foi, em média, de $69,23\% \pm 6,52\%$ para as imagens da divisão por paciente e $79,92\% \pm 5,09\%$ para as imagens da divisão aleatória.

O segundo melhor resultado foi alcançado pelos classificadores SVM que utilizaram os *kernels* polinomial e gaussiano (RBF). Dentre os diferentes *kernels* avaliados o sigmoide foi o que apresentou os piores resultados. As maiores acurácias obtidas pelo SVM foram, em média, de $68,24\% \pm 4,65\%$ para as imagens da divisão por paciente e $78,14\% \pm 5,76\%$ para as imagens da divisão aleatória.

O terceiro melhor resultado foi alcançado pelos classificadores Floresta Aleatória. A escolha ótima da quantidade de arvores aleatória e da redução de componentes principais com PCA varia em função do conjunto de características utilizado, não existindo uma regra evidente capaz de definir o valor ótimo destes dois parâmetros do classificador. As maiores acurácias obtidas por este método de classificação foi, em média, de $68,04\% \pm 4,98\%$ para as imagens da divisão por paciente e $78,05\% \pm 4,18\%$ para as imagens da divisão aleatória.

O método que apresentou os piores resultados foi o (*K-Nearest Neighbors*). Da mesma forma que ocorreu com os parâmetros do classificador Floresta Aleatória, a escolha ótima da quantidade de vizinhos observados e da redução de componentes principais com PCA varia em função do conjunto de características utilizado, não existindo uma regra evidente capaz de definir o valor ótimo destes dois parâmetros do classificador. As maiores acurácias obtidas por este método de classificação foi, em média, de $66,24\% \pm 3,14\%$ para as imagens da divisão por paciente e 75,69% ± 3,93% para as imagens da divisão aleatória.

A respeito das redes convolucionais avaliadas neste trabalho, as arquiteturas VGGNet (VGG16 e VGG19) e *Xception* obtiveram os melhores resultados. Com as imagens da divisão por paciente a maior acurácia, 77,47%, foi obtida pela arquitetura *Xception*. Com as imagens da divisão aleatória a maior acurácia foi de 92,48%, obtida pela arquitetura VGG16.

Dentre os métodos de classificação avaliados aquele que obteve a maior acurácia com as imagens da divisão por paciente foi obtida por uma rede neural, que utiliza as estatísticas de 1ª ordem para classificar os linfócitos. Para as imagens da divisão aleatória a maior acurácia foi obtida pela rede convolucional VGG16. As estatísticas de desempenho deste dois classificadores pode ser vista na tabela 91.

Tabela 91 – Desempenho dos melhores métodos de classificação

Conj.Treino	Classificador	Acurácia	Precisão	Sensibilidade	Especificidade	*F1-score*
Divisão por paciente (com augm.)	Rede Neural	81,04%	76,78%	89,01%	73,08%	82,44%
Divisão aleatória (com augm.)	VGG16	92,48%	87,64%	94,10%	86,73%	90,76%

Estes resultados podem ser comparados com alguns dos principais trabalhos apresentados no capitulo 2 apesar de haver diferenças nas metodologias adotadas em cada trabalho. As duas

principais diferenças estão na quantidade de imagens utilizadas em cada trabalho e na divisão destas imagens em treino, validação e teste. Dentre a bibliografia estudada somente o artigo de Mourya et al. (2018) leva em consideração a divisão por paciente, logo somente neste trabalho podem ser comparados os resultados obtidos com as imagens da divisão por paciente, nos outros casos só é válida a comparação com os resultados da divisão aleatória.

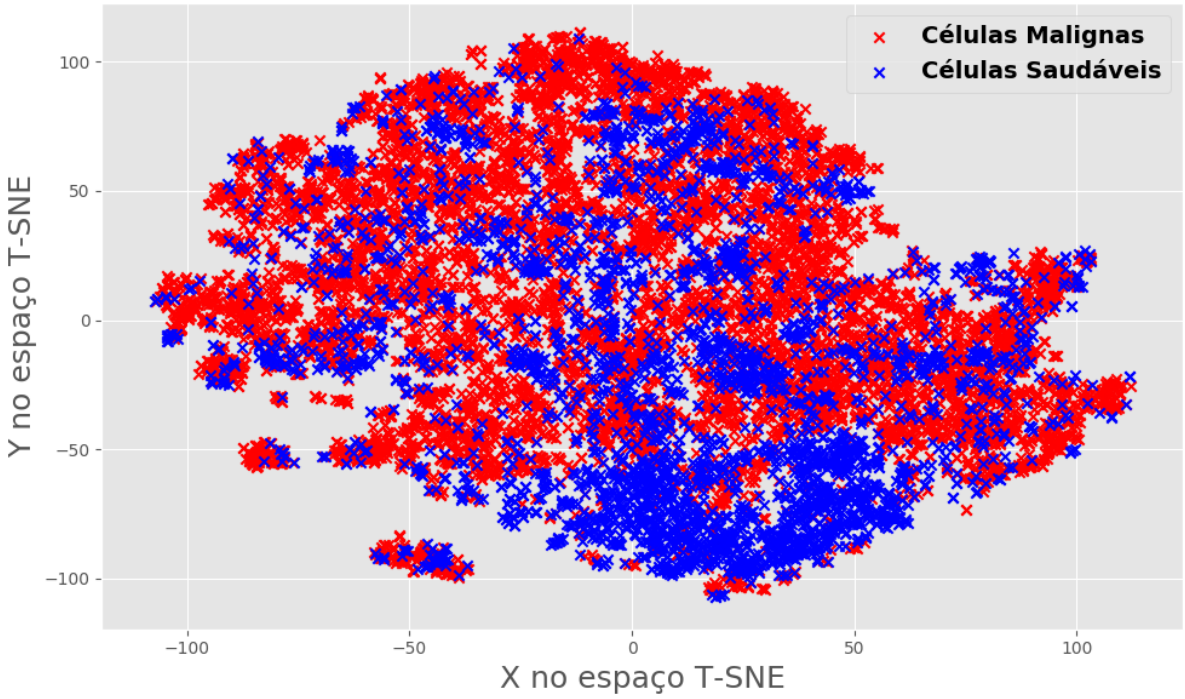
O artigo de Putzu et al. (2014) apresenta como resultado uma acurácia de 93,20% na identificação de linfócitos malignos, 0,72% maior do que os resultados alcançados neste trabalho. Entretanto foi utilizado um conjunto de imagens muito menor (108 imagens comparado com as 10661 imagens utilizadas por este trabalho). Não houve a criação de um conjunto de teste e foi utilizada a técnica de *cross-validation*, a qual permite que durante o treinamento o classificador aprenda a classificar todas as imagens do conjunto de validação.

O artigo de Shafique e Tehsin (2018) apresenta como resultados uma acurácia de 99,50% na identificação dos linfócitos malignos, 7,02% maior do que os resultados alcançados neste trabalho. Entretanto também foi utilizado um número muito menor de imagens (apenas 500) e não foi utilizado conjunto de teste, somente treino e validação.

O artigo de Mourya et al. (2018) é o único que leva em consideração a divisão por paciente e em seus resultados é apresentada uma acurácia de 89,70%, que é 8,66% maior do que os resultados alcançados neste trabalho. Para conseguir este resultado foram utilizados 9211 linfócitos malignos de 65 pacientes doentes e 4528 linfócitos saudáveis de 52 paciente saudáveis. Em comparação, este trabalho alcançou 81,04% de acurácia na divisão por paciente com 7272 linfócitos malignos de 47 pacientes doentes e 3389 linfócitos saudáveis de 26 pacientes saudáveis.

O principal desafio neste problema é a semelhança que existe entre as duas classes de linfócitos, o que dificulta a criação de classificador ótimo para este tipo de problema. A semelhança entre as classes pode ser visualizada aplicando a redução de dimensionalidade *t-distributed Stochastic Neighbor Embedding* (t-SNE) (MAATEN; HINTON, 2008) nas estatísticas de 1ª ordem (que obtiveram o melhor desempenho de classificação dentre todos os conjuntos de características) reduzindo as 108 características para apenas 2 e plotando o resultado em um plano cartesiano de 2 dimensões. O resultado disto pode ser visto na Figura 73.

Figura 73 – Redução de dimensionalidade das estatísticas de 1ª ordem de 108 características para apenas 2 com o método T-SNE. Foram utilizadas todas as 10661 imagens do *dataset* e cada ponto no gráfico representa um linfócito.



Ao analisar a Figura 73 nota-se que, apesar de ser possível identificar visualmente uma separação entre os linfócitos saudáveis e malignos, ainda ocorrem inúmeras colisões de elementos de classes distintas. Nota-se também uma ausência de separação linear entre as duas classes, fato que pode se manter em dimensões mais elevadas.

Como orientação para trabalhos futuros sugere-se a adição de outras estatísticas que possam ser coletadas dos histogramas dos *pixels* dos linfócitos, pois dentre os conjuntos de características avaliados, esta foi a que apresentou a melhor capacidade de separar as células saudáveis das malignas.

Com relação aos algoritmos de aprendizagem de máquina foi observado que as Redes Neurais Multicamadas apresentaram as maiores acurácias dentre os métodos avaliados. A função de ativação dos neurônios, a quantidade de camadas e a quantidade de neurônios são parâmetros destas redes que apresentaram grande influência na acurácia do classificador. Neste sentido a utilização de outras funções de ativação como a *Leaky*ReLU (MAAS et al., 2013), PReLU (HE et al., 2015), ELU (CLEVERT et al., 2015) ou *Thresholded*ReLU (KONDA et al., 2014) pode apresentar melhores resultados.

As arquiteturas *Xception* e VGGNet (VGG16 e VGG19) apresentaram os melhores desempenhos dentre as redes neurais convolucionais, com destaque a VGG16 que apresentou a maior acurácia dentre todos os classificadores desenvolvidos com as imagens da divisão

aleatória, de 92,48%. Entretanto esta acurácia não difere muito da acurácia alcançada pelas arquiteturas VGG19 (91,59%) e *Xception* (90,41%). Estes três resultados foram alcançados somente modificando a camada totalmente conectada no topo da rede destas para se adaptar ao problema de classificação dos linfócitos. Como sugestão para trabalhos futuros pode ser produtivo o estudo de meios para modificar as camadas convolucionais de forma a otimizá-las para classificação de células. Desta forma pode se tornar possível aumentar ainda mais a acurácia do classificador.

Com relação às imagens utilizadas, o fato de existirem mais amostras de células malignas do que saudáveis fez com que os classificadores se tornassem mais especializados em detectar as células malignas do que saudáveis, isto se traduz no fato da Sensibilidade (Taxa de Verdadeiro Positivo) ser sempre superior a Especificidade (Taxa de Verdadeiro Negativo). Esta diferença pode ser reduzida com a adição de mais amostras de células saudáveis no conjunto de imagens de treinamento. A utilização de linfócitos de pacientes distintos também pode mostrar-se útil, pois o aumento da variedade de células as quais os classificadores são expostos durante o treinamento parece influenciar positivamente as suas acurácias.

Referências

ALEXIS COOK. *Global Average Pooling Layers for Object Localization*. 2017. Disponível em: <https://alexisbcook.github.io/2017/global-average-pooling-layers-for-object-localization/>. Acesso em: 29 jul. 2019. Citado na página 44.

ALHILAL, M. S.; SOUDANI, A.; AL-DHELAAN, A. Image-based object identification for efficient event-driven sensing in wireless multimedia sensor networks. *International Journal of Distributed Sensor Networks*, SAGE Publications, v. 11, n. 3, p. 850869, jan. 2015. Disponível em: <https://doi.org/10.1155/2015/850869>. Citado na página 73.

AMADASUN, M.; KING, R. Textural features corresponding to textural properties. *IEEE Transactions on Systems, Man, and Cybernetics*, Institute of Electrical and Electronics Engineers (IEEE), v. 19, n. 5, p. 1264–1274, 1989. Disponível em: <https://doi.org/10.1109/21.44046>. Citado na página 67.

BAIN, B. J. *Blood Cells: A Pratical Guide*. 5. ed. West Sussex, Chichester, UK: John Wiley & Sons Ltd., 2015. ISBN 978-1-118-81733-9. Citado na página 22.

BENNETT, J. M. et al. Proposals for the classification of the acute leukaemias french-american-british (FAB) co-operative group. *British Jornal of Haematology*, Wiley, v. 33, n. 4, p. 451–458, ago. 1976. Disponível em: <https://doi.org/10.1111/j.1365-2141.1976.tb03563.x>. Citado na página 22.

BUDUMA, N. *Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms*. O'Reilly Media, 2017. ISBN 9781491925614. Disponível em: <https://www.amazon.com/Fundamentals-Deep-Learning-Next-Generation-Intelligence/dp/1491925612?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=1491925612>. Citado 4 vezes nas páginas 38, 41, 42 e 43.

CHOLLET, F. Xception: Deep learning with depthwise separable convolutions. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017. Disponível em: <https://doi.org/10.1109/cvpr.2017.195>. Citado 2 vezes nas páginas 48 e 50.

CLEVERT, D.-A.; UNTERTHINER, T.; HOCHREITER, S. *Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)*. 2015. Citado na página 156.

CORTES, C.; VAPNIK, V. Support-vector networks. *Machine Learning*, Springer Science and Business Media LLC, v. 20, n. 3, p. 273–297, set. 1995. Disponível em: <https://doi.org/10.1007/bf00994018>. Citado na página 30.

COSGRIFF, R. L. Identification of shape. *OHIO STATE UNIVERSITY RESEARCH FOUNDATION, COLUMBUS, REP. 820-11,. ASTIA AD 254 792*, 1960. Citado na página 73.

DUAN, K. bo; KEERTHI, S. S. Which is the best multiclass svm method? an empirical study. In: *Proceedings of the Sixth International Workshop on Multiple Classifier Systems*. [S.l.: s.n.], 2005. p. 278–285. Citado na página 30.

ELWYSLAN. *Source Code: ISBI-2019-Cells-Classification*. 2019. Disponível em: <https://github.com/Elwyslan/ISBI-2019-Cells-Classification>. Acesso em: 23 set. 2019. Citado na página 88.

FREECODECAMP. *An intuitive guide to Convolutional Neural Networks*. 2018. Disponível em: <https://www.freecodecamp.org/news/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050/>. Acesso em: 1 jul. 2019. Citado na página 42.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. MIT Press, 2016. Disponível em: <http://www.deeplearningbook.org>. Citado 3 vezes nas páginas 40, 41 e 42.

GRIETHUYSEN, J. J. van et al. Computational radiomics system to decode the radiographic phenotype. *Cancer Research*, American Association for Cancer Research (AACR), v. 77, n. 21, p. e104–e107, out. 2017. Disponível em: <https://doi.org/10.1158/0008-5472.can-17-0339>. Citado 2 vezes nas páginas 52 e 83.

HE, K. et al. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2015. Disponível em: <https://doi.org/10.1109/iccv.2015.123>. Citado na página 156.

HE, K. et al. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016. Disponível em: <https://doi.org/10.1109/cvpr.2016.90>. Citado na página 51.

HINTON, G. E. Reducing the dimensionality of data with neural networks. *Science*, American Association for the Advancement of Science (AAAS), v. 313, n. 5786, p. 504–507, jul. 2006. Disponível em: <https://doi.org/10.1126/science.1127647>. Citado 2 vezes nas páginas 86 e 153.

HINTON, G. E. et al. *System and method for addressing overfitting in a neural network*. 2012. Patente US9406017B2 pertencente a Google LLC. Citado na página 44.

HOFFBRAND, A. V.; MOSS, P. A. H. *Fundamentos em Hematologia*. 6. ed. Porto Alegre, RS, BRA: Artmed, 2013. ISBN 978-85-65852-30-2. Citado 3 vezes nas páginas 20, 21 e 22.

HOUBY, E. M. F. E. Framework of computer aided diagnosis systems for cancer classification based on medical images. *Journal of Medical Systems*, Springer Science and Business Media LLC, v. 42, n. 8, jul. 2018. Disponível em: <https://doi.org/10.1007/s10916-018-1010-x>. Citado na página 83.

IOFFE, S.; SZEGEDY, C. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. Citado 2 vezes nas páginas 44 e 45.

JUNQUEIRA, L. C.; CARNEIRO, J. *Histologia Básica*. 10. ed. Rio de Janeiro, RJ, BRA: Guanabara Koogan S.A., 2004. ISBN 978-85-277-0906-4. Citado na página 21.

KITWARE INC. 2019. *Run-Length Matrices For Texture Analysis*. 2019. Pyradiomics Community Copyright 2016. Disponível em: <http://hdl.handle.net/1926/1374>. Acesso em: 1 jul. 2019. Citado na página 64.

KONDA, K.; MEMISEVIC, R.; KRUEGER, D. *Zero-bias autoencoders and the benefits of co-adapting features*. 2014. Citado na página 156.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*. USA: Curran Associates Inc., 2012. (NIPS'12), p. 1097–1105. Disponível em: <http://dl.acm.org/citation.cfm?id=2999134.2999257>. Citado na página 26.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, Association for Computing Machinery (ACM), v. 60, n. 6, p. 84–90, maio 2017. Disponível em: <https://doi.org/10.1145/3065386>. Citado na página 40.

KUNLUN BAI. *A Comprehensive Introduction to Different Types of Convolutions in Deep Learning*. 2011. Disponível em: <https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215>. Acesso em: 1 jul. 2019. Citado 3 vezes nas páginas 48, 49 e 50.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, Springer Science and Business Media LLC, v. 521, n. 7553, p. 436–444, maio 2015. Disponível em: <https://doi.org/10.1038/nature14539>. Citado na página 38.

LORENZI, T. F. *Manual De Hematologia Propedêutica E Clínica*. 4. ed. Rio de Janeiro, RJ, BRA: Guanabara Koogan S.A., 2006. ISBN 978-85-277-1791-5. Citado 3 vezes nas páginas 20, 21 e 22.

LOURDES CHAUFFAILLE, M. de; MIHOKO, Y. *Classificação das Leucemias Agudas. Citologia, Citoquímica, Imunofenotipagem, Citogenética e Genética Molecular*. 2019. Disponível em: <https://edisciplinas.usp.br/mod/resource/view.php?id=1268091>. Acesso em: 2 mar. 2019. Citado na página 22.

MAAS, A.; HANNUN, A.; NG, A. Rectifier nonlinearities improve neural network acoustic models. In: *Proceedings of the International Conference on Machine Learning*. Atlanta, Georgia: [s.n.], 2013. Citado na página 156.

MAATEN, L. van der; HINTON, G. Visualizing High-Dimensional Data Using t-SNE. *Journal of Machine Learning Research*, v. 9: 2579–2605, 2008. Citado na página 155.

MISHRA, S. et al. Gray level co-occurrence matrix and random forest based acute lymphoblastic leukemia detection. *Biomedical Signal Processing and Control*, Elsevier BV, v. 33, p. 272–280, mar 2017. Disponível em: <https://doi.org/10.1016/j.bspc.2016.11.021>. Citado 2 vezes nas páginas 26 e 83.

MISHRA, S. et al. Microscopic image classification using DCT for the detection of acute lymphoblastic leukemia (ALL). In: *Advances in Intelligent Systems and Computing*. Springer Singapore, 2016. p. 171–180. Disponível em: <https://doi.org/10.1007/978-981-10-2104-6_16>. Citado na página 26.

MORADIAMIN, M. et al. Computer aided detection and classification of acute lymphoblastic leukemia cell subtypes based on microscopic image analysis. *Microscopy Research and Technique*, Wiley, v. 79, n. 10, p. 908–916, jul 2016. Disponível em: <https://doi.org/10.1002/jemt.22718>. Citado 3 vezes nas páginas 25, 76 e 83.

MOURYA, S. et al. *LeukoNet: DCT-based CNN architecture for the classification of normal versus Leukemic blasts in B-ALL Cancer*. 2018. Citado 6 vezes nas páginas 27, 28, 76, 82, 152 e 155.

NAIR, V.; HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In: *ICML - 27th International Conference on Machine Learning*. Omnipress, 2010. p. 807–814. Disponível em: <https://www.cs.toronto.edu/~hinton/absps/reluICML.pdf>. Citado na página 39.

NASCIMENTO, J. P. R. do; MADEIRA, H. M. F.; PEDRINI, H. Classificação de imagens utilizando descritores estatísticos de textura. *Anais do XI Simpósio Brasileiro de Sensoriamento Remoto - SBSR*, p. 2099–2106, 2003. Disponível em: <http://marte.sid.inpe.br/col/ltid.inpe.br/sbsr/2002/11.14.15.15/doc/15_185.pdf>. Citado na página 56.

NIXON, M. *Feature Extraction & Image Processing*. Academic Press, 2008. ISBN 0123725380. Disponível em: <https://www.amazon.com/Feature-Extraction-Image-Processing-Nixon/dp/0123725380?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=0123725380>. Citado 4 vezes nas páginas 42, 73, 74 e 75.

ONCOGUIA, E. *Exames para Diagnóstico da Leucemia Linfoide Aguda (LLA)*. 2018. Capítulo de livro disponibilizado online. Disponível em: <http://www.oncoguia.org.br/conteudo/exames-para-diagnostico-da-leucemia-linfoide-aguda-lla/1150/317/>. Acesso em: 2 mar. 2019. Citado na página 22.

PUTZU, L.; CAOCCI, G.; RUBERTO, C. D. Leucocyte classification for leukaemia detection using image processing techniques. *Artificial Intelligence in Medicine*, Elsevier BV, v. 62, n. 3, p. 179–191, nov. 2014. Disponível em: <https://doi.org/10.1016/j.artmed.2014.09.002>. Citado 3 vezes nas páginas 25, 83 e 155.

PYRADIOMICS. *Pyradiomics*. 2019. Pyradiomics Community Copyright 2016. Disponível em: <https://pyradiomics.readthedocs.io/en/latest/index.html>. Acesso em: 1 jul. 2019. Citado 3 vezes nas páginas 56, 60 e 67.

RUSSAKOVSKY, O. et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, v. 115, n. 3, p. 211–252, 2015. Citado na página 45.

SBILAB. *Classification of Normal vs Malignant Cells in B-ALL White Blood Cancer Microscopic Images: ISBI 2019*. 2019. Disponível em: <https://competitions.codalab.org/competitions/20395>. Acesso em: 2 mar. 2019. Citado 3 vezes nas páginas 22, 78 e 167.

SCIKIT-LEARN. *Support Vector Machines — scikit-learn 0.20.2 documentation*. 2019. Scikit-learn developers (BSD License). Disponível em: <https://scikit-learn.org/stable/modules/svm.html>. Acesso em: 1 jul. 2019. Citado 2 vezes nas páginas 30 e 86.

SHAFIQUE, S.; TEHSIN, S. Acute lymphoblastic leukemia detection and classification of its subtypes using pretrained deep convolutional neural networks. *Technology in Cancer Research & Treatment*, SAGE Publications, v. 17, p. 153303381880278, jan 2018. Disponível em: <https://doi.org/10.1177/1533033818802789>. Citado 3 vezes nas páginas 26, 27 e 155.

SHALEV-SHWARTZ, S.; BEN-DAVID, S. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014. ISBN 1107057132. Disponível em: <https://www.amazon.com/Understanding-Machine-Learning-Theory-Algorithms/dp/1107057132?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=1107057132>. Citado 6 vezes nas páginas 29, 32, 33, 35, 36 e 52.

SIMONYAN, K.; ZISSERMAN, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2014. Citado 2 vezes nas páginas 45 e 46.

SRIVASTAVA, N. et al. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, v. 15, p. 1929–1958, 2014. Disponível em: <http://jmlr.org/papers/v15/srivastava14a.html>. Citado na página 45.

STANFORD UNIVERSITY. *CS231n: Convolutional Neural Networks for Visual Recognition*. 2017. Stanford University course under MIT License. Disponível em: <http://cs231n.stanford.edu/>. Acesso em: 1 jul. 2019. Citado 4 vezes nas páginas 38, 41, 43 e 44.

STRALEN, K. J. van et al. Diagnostic methods i: sensitivity, specificity, and other measures of accuracy. *Kidney International*, Elsevier BV, v. 75, n. 12, p. 1257–1263, jun. 2009. Disponível em: <https://doi.org/10.1038/ki.2009.92>. Citado na página 76.

SURYANI, E.; WIHARTO; POLVONOV, N. Identification and counting white blood cells and red blood cells using image processing case study of leukemia. *International journal of Computer Science & Network Solutions*, v. 2, n. 6, p. 35–49, jun. 2014. ISSN 2345-3397. Citado 2 vezes nas páginas 24 e 25.

SZEGEDY, C. et al. *Going Deeper with Convolutions*. 2014. Citado 2 vezes nas páginas 46 e 47.

SZEGEDY, C. et al. Rethinking the inception architecture for computer vision. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016. Disponível em: <https://doi.org/10.1109/cvpr.2016.308>. Citado 2 vezes nas páginas 47 e 48.

TAN, S.; MAYROVOUNIOTIS, M. L. Reducing data dimensionality through optimizing neural network inputs. *AIChE Journal*, Wiley, v. 41, n. 6, p. 1471–1480, jun. 1995. Disponível em: <https://doi.org/10.1002/aic.690410612>. Citado na página 153.

UCI DEPARTMENT OF MATHEMATICS. *Centroid Distance Function and the Fourier Descriptor with Applications to Cancer Cell Clustering*. 2011. Disponível em: <https://www.math.uci.edu/icamp/summer/research_11/klinzmann/cdfd.pdf>. Acesso em: 1 jul. 2019. Citado na página 73.

VERRASTRO, T.; LORENZI, T. F.; NETO, S. W. *Hematologia e Hemoterapia - Fundamentos de Morfologia, Fisiologia, Patologia e Clínica*. 1. ed. Rio de Janeiro, RJ, BRA: Editora Atheneu LTDA., 2006. ISBN 8573792272. Citado na página 22.

VISHWAKARMA, V. P.; PANDEY, S.; GUPTA, M. A novel approach for face recognition using DCT coefficients re-scaling for illumination normalization. In: *15th International Conference on Advanced Computing and Communications (ADCOM 2007)*. IEEE, 2007. Disponível em: <https://doi.org/10.1109/adcom.2007.12>. Citado na página 84.

WIKIMEDIA FOUNDATION, INC. *Support-Vector machine wiki*. 2019. Creative Commons Attribution-ShareAlike License. Disponível em: <https://en.wikipedia.org/wiki/Support-vector_machine>. Acesso em: 26 nov. 2019. Citado na página 31.

WU, Y.-G. Medical image compression by sampling DCT coefficients. *IEEE Transactions on Information Technology in Biomedicine*, Institute of Electrical and Electronics Engineers (IEEE), v. 6, n. 1, p. 86–94, mar. 2002. Disponível em: <https://doi.org/10.1109/4233.992167>. Citado na página 84.

ZWANENBURG, A. et al. *Image Biomarker Standardisation Initiative*. 2016. Citado 3 vezes nas páginas 52, 53 e 69.

Apêndices

APÊNDICE A – Código responsável por realizar a divisão aleatória do dataset

```
import numpy as np
ALLtrainImgs = ALLvalidImgs = ALLtestImgs = []
#Retorna a lista com todas as células malignas do dataset
ALLcells = getCellsImgPath('ALL')
#60% das células malignas serão utilizadas para Treino
maxTrainSize = int(np.ceil(len(ALLcells)*0.6))
#30% das células malignas serão utilizadas para Validação
maxValidSize = int(np.floor(len(ALLcells)*0.3))

#Distribuir aleatoriamente as células entre Treino, Validação e Teste
while len(ALLcells) > 0:
  np.random.shuffle(ALLcells)
  if len(ALLtrainImgs) < maxTrainSize:
    ALLtrainImgs.append(ALLcells.pop())
    continue
  if len(ALLvalidImgs) < maxValidSize:
    ALLvalidImgs.append(ALLcells.pop())
    continue
  ALLtestImgs.append(ALLcells.pop())

HEMtrainImgs = HEMvalidImgs = HEMtestImgs = []
#Retorna a lista com todas as células saudáveis do dataset
HEMcells = getCellsImgPath('HEM')
#60% das células saudáveis serão utilizadas para Treino
maxTrainSize = int(np.ceil(len(HEMcells)*0.6))
#30% das células saudáveis serão utilizadas para Validação
maxValidSize = int(np.floor(len(HEMcells)*0.3))

#Distribuir aleatoriamente as células entre Treino, Validação e Teste
while len(HEMcells) > 0:
  np.random.shuffle(HEMcells)
  if len(HEMtrainImgs) < maxTrainSize:
    HEMtrainImgs.append(HEMcells.pop())
    continue
  if len(HEMvalidImgs) < maxValidSize:
    HEMvalidImgs.append(HEMcells.pop())
    continue
  HEMtestImgs.append(HEMcells.pop())
#Definir as células dos conjuntos de Treino, Validação and Teste
trainImgs = ALLtrainImgs + HEMtrainImgs
validImgs = ALLvalidImgs + HEMvalidImgs
testImgs = ALLtestImgs + HEMtestImgs
```

APÊNDICE B – Código responsável por realizar a divisão por paciente do dataset

```
import numpy as np
ALLtrainPat = ALLvalidPat = ALLtestPat = []
#Retorna os IDs de todos os pacientes diagnosticados com LLA
ALLpatientsIDs = getIdsALLPatients()
#60% dos pacientes serão utilizados para Treino
maxPatTrain = int(np.ceil(len(ALLpatientsIDs)*0.6))
#30% dos pacientes serão utilizados para Validação
maxPatValid = int(np.floor(len(ALLpatientsIDs)*0.3))

#Distribuir aleatoriamente pacientes entre Treino, Validação e Teste
while len(ALLpatientsIDs) > 0:
  np.random.shuffle(ALLpatientsIDs)
  if len(ALLtrainPat) < maxPatTrain:
    ALLtrainPat.append(ALLpatientsIDs.pop())
    continue
  if len(ALLvalidPat) < maxPatValid:
    ALLvalidPat.append(ALLpatientsIDs.pop())
    continue
  ALLtestPat.append(ALLpatientsIDs.pop())

HEMtrainPat = HEMvalidPat = HEMtestPat = []
#Retorna os IDs de todos os pacientes saudáveis
HEMpatientsIDs = getIdsHEMPatients()
#60% dos pacientes serão utilizados para Treino
maxPatTrain = int(np.ceil(len(HEMpatientsIDs)*0.6))
#30% dos pacientes serão utilizados para Validação
maxPatValid = int(np.floor(len(HEMpatientsIDs)*0.3))

#Distribuir aleatoriamente pacientes entre Treino, Validação e Teste
while len(HEMpatientsIDs) > 0:
  np.random.shuffle(HEMpatientsIDs)
  if len(HEMtrainPat) < maxPatTrain:
    HEMtrainPat.append(HEMpatientsIDs.pop())
    continue
  if len(HEMvalidPat) < maxPatValid:
    HEMvalidPat.append(HEMpatientsIDs.pop())
    continue
  HEMtestPat.append(HEMpatientsIDs.pop())
#Definir os pacientes dos conjuntos de Treino, Validação and Teste
trainPat = HEMtrainPat + ALLtrainPat
validPat = HEMvalidPat + ALLvalidPat
testPat = HEMtestPat + ALLtestPat
```

APÊNDICE C – ISBI 2019

A fase final da competição ISBI 2019 (SBILAB, 2019) consiste na classificação de 2586 imagens de linfócitos entre malignos e saudáveis. A métrica de avaliação utilizada é o *Weighted F1-Score*, o seu cálculo é realizado pelos servidores que hospedam a competição, o participante só é responsável por submeter no site somente um arquivo de texto que contenha a predição da classe de cada imagem do conjunto de teste fornecido. Após a submissão os servidores calculam o *Weighted F1-Score*. Os maiores resultados obtidos nesta competição já foram divulgados e podem ser vistos no quadro 6.

Quadro 6 – Resultados Finais da competição IBSI2019

Top Entries of C-NMC 2019 Challenge		
Name	Rank	Weighted F1 Score
Yongsheng Pan	1	0.910
Ekansh Verma	2	0.894
Jonas Prellberg	3	0.889
Fenrui Xiao	4	0.885
Tian Shi	5	0.879
Ying Liu	6	0.876
Salman Shah	8	0.866
Yifan Ding	9	0.855
Xinpeng Xie	10	0.848

Fonte: SBILab (2019)

Nesta competição foram submetidos os modelos das arquiteturas *Xception*, VGG16 e VGG19 treinados por 400 épocas com as imagens da divisão por paciente e divisão aleatória. Os resultados obtidos por estes modelos podem ser vistos no quadro 7.

Quadro 7 – Resultados obtidos pelas redes convolucionais treinadas

0.8191186178	isbi_valid_VGG16_AugmPatDiv.zip	09/20/2019 03:38:54	Finished	
0.8324786937	isbi_valid_VGG16_AugmRndDiv.zip	09/20/2019 03:39:13	Finished	
0.8031937427	isbi_valid_VGG19_augmPatLvDiv.zip	09/20/2019 03:39:30	Finished	
0.8252010758	isbi_valid_VGG19_augmRndDiv.zip	09/20/2019 03:39:52	Finished	
0.82619161	isbi_valid_xception_AugmPatDiv.zip	09/20/2019 03:40:08	Finished	
0.8635177019	isbi_valid_xception_augmRndDiv.zip	09/20/2019 03:40:26	Finished	✔

Fonte: SBILab (2019)

O maior resultado, 0,8635177019%, foi obtido pela rede *Xception* treinada com as imagens da divisão aleatória. Sua pontuação coloca-a na 9ª posição do *ranking* do desafio.