



Universidade Estadual de Feira de Santana
Programa de Pós-Graduação em Computação Aplicada

Explorando Técnicas de Aprendizado Híbrido para o Reconhecimento Automático de Imagens de Plantas

Marcondes Coelho Feitoza

Feira de Santana

2019



Universidade Estadual de Feira de Santana
Programa de Pós-Graduação em Computação Aplicada

Marcondes Coelho Feitoza

**Explorando Técnicas de Aprendizado Híbrido
para o Reconhecimento Automático de Imagens
de Plantas**

Dissertação apresentada à Universidade Estadual de Feira de Santana como parte dos requisitos para a obtenção do título de Mestre em Computação Aplicada.

Orientador: Prof. Rodrigo Tripodi Calumby

Feira de Santana

2019

Ficha Catalográfica - Biblioteca Central Julieta Carteado - UEFS

F336

Feitoza, Marcondes Coelho

Explorando técnicas de aprendizado híbrido para o reconhecimento automático de imagens de plantas / Marcondes Coelho Feitoza. – 2019.

90 f.: il.

Orientador: Rodrigo Tripodi Calumby.

Dissertação (mestrado) – Universidade Estadual de Feira de Santana, Programa de Pós-graduação em Computação Aplicada, Feira de Santana, 2019.

1. Aprendizado de máquina. 2. Aprendizagem profunda. 3. Plantas – identificação. 4. Reconhecimento automático de imagens. 5. Redes neurais convolucionais.

I. Calumby, Rodrigo Tripodi, orient. II. Universidade Estadual de Feira de Santana.

III. Título.

CDU: 004.932:004.85

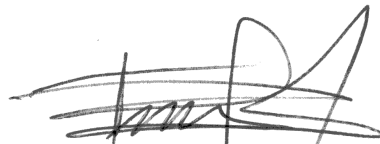
Marcondes Coelho Feitoza

Explorando Técnicas de Aprendizado Híbrido para Reconhecimento Automático de Imagens de Plantas

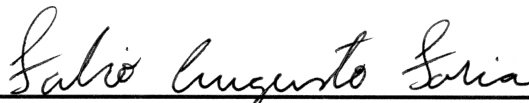
Dissertação apresentada à Universidade Estadual de Feira de Santana como parte dos requisitos para a obtenção do título de Mestre em Computação Aplicada.

Feira de Santana, 12 de agosto de 2019

BANCA EXAMINADORA



Dr. Rodrigo Tripodi Calumby (Orientador)
Universidade Estadual de Feira de Santana



Dr. Fábio Augusto Faria
Universidade Federal de São Paulo



Dr. Angelo Conrado Loula
Universidade Estadual de Feira de Santana

Abstract

In recent years, with the evolution of Convolutional Neural Networks (RNC's), the automatic recognition of plant species from images has become a very relevant research topic for scientists, researchers and students in both the botany and computer communities. The main challenges involving automatic recognition of plant species are directly related to intra-class variability and inter-class similarity, both arising from the complexity of the images in question. The main objective of this work is to explore hybrid learning techniques, that is, the combination of supervised and unsupervised learning techniques with the purpose of minimizing the impacts of this variability in the process of classification of plant images. In this work we explored the use of features extracted by RNC's for the recognition of plant species images. As object of study, the ImageCLEF2013 (PlantCLEF) image collection with 26,077 images from 250 plant species was used. Partitioning techniques were applied to each base with different features. In addition, more comprehensive classification approaches were explored, using classical methods such as the Random Forest algorithm and eight variations of the SVM classifier with features extracted by RNC's Inception V3, VGG-16 and VGG-19. Nevertheless, the Softmax classifier layer of each of the RNC's was also considered in order to verify the impact of partitioning on the image recognition process of plant species. As a result, the experiments show it is possible to improve the results of classifier effectiveness by combining feature extraction by RNC's and class partitioning with grouping techniques.

Keywords: machine learning, deep learning, plant identification

Resumo

Nos últimos anos, com a evolução das Redes Neurais Convolucionais (RNC's), o reconhecimento automático de espécies de plantas a partir de imagens tornou-se um tema de pesquisa muito relevante para cientistas, pesquisadores e estudantes tanto na área da botânica quanto na comunidade computacional. Os principais desafios que envolvem o reconhecimento automático de espécies de plantas está diretamente relacionado a variabilidade intra-classe e da similaridade inter-classe, ambas decorrentes da complexidade das imagens em questão. Este trabalho tem como principal objetivo explorar técnicas de aprendizado híbrido, ou seja, a combinação de técnicas de aprendizado supervisionado e não-supervisionado com o propósito é minimizar os impactos dessa variabilidade no processo de classificação de imagens de plantas. Neste trabalho foi explorada a utilização de *features* extraídas por RNC's para o reconhecimento de imagens de espécies de plantas. Como objeto de estudo, utilizou-se a coleção de imagens ImageCLEF2013 (PlantCLEF) com 26.077 imagens de 250 espécies de plantas. Foram aplicadas técnicas de particionamento em cada uma das bases com as diferentes *features*. Além disso, foram exploradas abordagens de classificação mais abrangentes, com uso métodos clássicos como o algoritmo *Random Forest* e oito variações do classificador SVM com *features* extraídas pelas RNC's *Inception V3*, VGG-16 e VGG-19. Ainda assim, considerou-se também a camada do classificador Softmax de cada uma das RNC's a fim de se verificar o impacto do particionamento no processo de reconhecimento de imagens de espécies de plantas. Como resultado, os experimentos mostram é possível melhorar os resultados da eficácia dos classificadores combinando extração de *features* por RNC's e o particionamento das classes com técnicas de agrupamento.

Palavras-chave: aprendizado de máquina, aprendizagem profunda, identificação de plantas

Prefácio

Esta dissertação de mestrado foi submetida a Universidade Estadual de Feira de Santana (UEFS) como requisito parcial para obtenção do grau de Mestre em Computação Aplicada.

A dissertação foi desenvolvido dentro do Programa de Pós-Graduação em Computação Aplicada (PGCA) tendo como orientador o Dr. **Rodrigo Tripodi Calumby**.

Agradecimentos

Agradeço primeiramente a Deus pela oportunidade de ter feito parte destes programa de mestrado onde pude enriquecer ainda mais minha carreira profissional e acadêmica, e a minha mãe Elizabeth Almeida Coelho que sempre me amparou e sustentou em todas as minhas decisões e jornadas da vida.

Agradeço especialmente ao meu orientador Prof. Dr. Rodrigo Tripodi Calumby pelo voto de confiança ainda no processo de seleção deste programa de mestrado, quando ele enviou uma carta de aceite de orientação a mim, que me encontrava do outro lado do país em uma das áreas mais remotas que se possa imaginar. Agradeço também por me receber em Feira de Santana de maneira tão generosa e solidária juntamente com seus demais orientandos. Agradeço por ter me dado força em cada uma das etapas e pelas excelentes orientações recebidas em todo o tempo. Pois levo para toda minha vida, tanto pessoal quanto profissional, um exemplo de professor, orientador e de humanidade.

Agradeço também pelos amigos Jeziel Lago, Marília Anjos, Wanderson Silva e Cadu Ritchelle, que conquistei e que tanto me ajudaram em diversos momentos dessa jornada, porque sem amigos não há ascensão.

Sumário

Abstract	i
Resumo	ii
Prefácio	iii
Agradecimentos	iv
Sumário	vii
Lista de Publicações	viii
Lista de Tabelas	ix
Lista de Figuras	xii
Lista de Abreviações	xiii
Lista de Símbolos	xiv
1 Introdução	1
1.1 Justificativa e Motivação	4
1.2 Objetivos	5
1.3 Contribuições	6
2 Fundamentação Teórica	7
2.1 Reconhecimento de Padrões	7
2.2 Classificação de Dados	8
2.2.1 Aprendizado Supervisionado e Não-Supervisionado	9
2.2.2 Classificadores Binários <i>vs.</i> Multi-classe	9
2.2.3 Redes Neurais Convolucionais Profundas	10
2.2.4 Extração de Características Utilizando <i>Deep Learning</i>	11
2.2.5 Arquiteturas de <i>Deep Learning</i>	12
2.3 Agrupamento de Dados	13
2.3.1 Tendência de Agrupamento	13

2.3.2	Qualidade de Agrupamentos e Coeficiente Silhueta	14
2.4	Medidas de avaliação	15
2.5	Redução de Dimensionalidade	17
2.6	Classificação de Imagens de Plantas	17
2.6.1	Aprendizado de Máquina para Identificação de Plantas	17
2.6.2	<i>Deep Learning</i> para Identificação de Plantas	19
3	Metodologia Proposta	22
3.1	Visão geral da Proposta	22
3.1.1	Fluxograma Detalhado da Proposta	23
3.2	Coleção de Imagens	24
3.3	Extração de Características	26
3.4	Descoberta de Subclasses	26
3.4.1	Identificando a Tendência de Particionamento	27
3.4.2	Método de Particionamento	27
3.5	Seleção de Features	27
3.6	Abordagens de Classificação	28
3.7	Modelo de Avaliação Comparativa de Eficácia	29
3.7.1	Análise de Estabilidade e Significância Estatística	30
3.8	Ferramentas Utilizadas	30
4	Resultados e Discussão	31
4.1	Caracterização do Problema de Variabilidade Intra-classe	31
4.2	Avaliação da Descoberta de Subclasses	34
4.2.1	Tendência de Agrupamento	34
4.2.2	Número de classes particionadas	36
4.2.3	Coeficiente de Silhueta	36
4.2.4	Quantidade de Subclasses Descobertas	39
4.2.5	Relação entre o número grupos e quantidade de amostras presentes nas classes	41
4.2.6	Quantidade de subclasses geradas no particionamento	42
4.2.7	Impacto do particionamento considerando o tamanho inicial das classes e desbalanceamento de amostras	44
4.2.8	Capacidade de diferenciação entre as subclasses descobertas	46
4.3	Avaliação da Classificação	47
4.3.1	Resultados Gerais de Eficácia	48
4.3.2	Estabilidade e significância estatística	51
4.4	Avaliação do Impacto da Seleção das <i>features</i> para a Classificação Baseada em Softmax	53
4.4.1	Estabilidade e significância estatística do classificador Softmax sem redução de dimensionalidade	56
4.5	Análise Visual dos Resultados	58
4.5.1	Ilustração Visual do processo de particionamento	58
4.5.2	Análise Visual das Matrizes de Confusão por Mapa de Calor	62

5	Considerações Finais	66
5.1	Trabalhos Futuros	67
	Referências Bibliográficas	68

Lista de Publicações

Artigo "*Exploring Deep Features and Transfer Learning for Plant Species Recognition*", publicado nos anais do XV Simpósio Brasileiro de Sistemas de Informação, realizado em Aracaju-SE, no período de 20 a 24 de maio de 2019.

Lista de Tabelas

3.1	Número de dimensões resultantes da Análise de Componentes Principais (PCA)	28
-----	--	----

Lista de Figuras

1.1	Processo tradicional de identificação da espécie de uma planta por comparação de características morfológicas de amostras catalogadas em herbários.	2
1.2	Variabilidade intra-classe: Diversidade de lóbulos de folhas de diferentes plantas da mesma espécie <i>Broussonetia papyrifera</i> (L.) L'H'er. ex Vent.	4
1.3	Variabilidade intra-classe: Quatro folhas de plantas distintas de uma mesma espécie (<i>Platanus x hispanica</i> Wild) coletadas em diferentes estágios de maturação.	5
1.4	Similaridade inter-classe: Folhas de diferentes espécies de plantas com características morfológicas similares: (a) <i>Adoxaceae viburnum tinus</i> , (b) <i>Lauraceae laurus nobilis</i> e (c) <i>Ericaceae arbutus unedo</i>	5
2.1	Estrutura de uma CNN composta por camadas convolucionais, de agregação (<i>pooling</i>) e totalmente conectadas.	11
2.2	Ilustração dos elementos envolvidos no cálculo de silhueta do objeto $i, i \in A$	14
2.3	Matriz de Confusão	15
2.4	Estrutura das MCDCNN proposta por [Anfeng He e Xinmei Tian 2016].	20
3.1	Fluxograma geral da proposta	24
3.2	Exemplos das imagens dos múltiplos órgãos de diferentes espécies divididas pelas categorias <i>SheetAsBackground</i> e <i>NaturalBackground</i>	25
3.3	Quadro-Resumo dos Classificadores utilizados para avaliação da técnica de particionamento.	28
3.4	Exemplo de matrizes de confusão com valores simulados de classificação com adaptação da matriz de confusão para a base particionada.	30
4.1	Dispersão dos valores mínimos e máximos de distância entre as amostras das classes considerando cada uma das <i>features</i>	32
4.2	Diferença entre os valores mínimos e máximos de distância das amostras de cada classe considerando cada uma das <i>features</i>	33
4.3	Histogramas de frequência dos valores de estatística de Hopkins para cada uma das <i>features</i> utilizadas.	35

4.4	Levantamento do número de espécies agrupadas e não agrupadas para cada uma das <i>features</i> utilizadas.	36
4.5	Frequência dos valores de silhueta utilizando <i>Features Inception V3</i> . . .	37
4.6	Frequência dos valores de silhueta utilizando <i>Features VGG 16</i>	38
4.7	Frequência dos valores de silhueta utilizando <i>Features VGG 19</i>	38
4.8	Quantidade de classes por número de grupos no particionamento das amostras com <i>Features Inception V3</i>	39
4.9	Quantidade de classes por número de grupos no particionamento das amostras com <i>Features VGG 16</i>	40
4.10	Quantidade de classes por número de grupos no particionamento das amostras com <i>Features VGG 19</i>	40
4.11	Relação do número de grupos pela quantidade de amostras considerando o particionamento com as <i>features</i> Inception V3.	41
4.12	Relação do número de grupos pela quantidade de amostras considerando o particionamento com as <i>features</i> VGG-16.	42
4.13	Relação do número de grupos pela quantidade de amostras considerando o particionamento com as <i>features</i> VGG-19.	42
4.14	Levantamento do número de subclasses geradas no particionamento para cada uma das <i>features</i> utilizadas.	43
4.15	Distribuição do número de amostras para cada uma das 250 classes originais da coleção de imagens ImageCLEF 2013.	44
4.16	Distribuição do número de amostras nas classes da base particionada com amostras de <i>features</i> Inception V3.	45
4.17	Distribuição do número de amostras para cada uma das classes da base particionada com amostras de <i>features</i> VGG-16	45
4.18	Distribuição do número de amostras para cada uma das classes da Base Particionada com amostras de <i>features</i> VGG-19.	46
4.19	Resultados de F1 com intervalo de confiança de 95% da classificação interna das particionadas considerando as três <i>features</i> utilizadas. . .	47
4.20	Resultados gerais de classificação (medida F1) para as Bases Original e Particionada com <i>features</i> extraídas pela Inception V3.	49
4.21	Resultados gerais de classificação com a medida F1 das Bases Original e Particionada com <i>features</i> extraídas pela VGG-16.	50
4.22	Resultados gerais de classificação com a medida F1 das Bases Original e Particionada com <i>features</i> extraídas pela VGG-19.	51
4.23	SVM - Linear - <i>One vs One</i> - Intervalo de Confiança 95% utilizando <i>Features</i> Inception V3.	52
4.24	SVM - Polynomial - <i>One vs Rest</i> - Intervalo de Confiança 95% utilizando <i>Features</i> VGG-16.	52
4.25	SVM - Polynomial - <i>One vs Rest</i> - Intervalo de Confiança 95% utilizando <i>Features</i> VGG-19.	53
4.26	Resultados comparativos da classificação “Com PCA” e “Sem PCA” usando o Softmax - Inception V3.	54

4.27	Resultados comparativos da classificação “Com PCA” e “Sem PCA” usando o Softmax - VGG-16.	55
4.28	Resultados comparativos da classificação “Com PCA” e “Sem PCA” usando o Softmax - VGG-19.	55
4.29	Eficácia do classificador Softmax com <i>features</i> sem redução de dimensionalidade - Intervalo de Confiança de 95% - Inception V3.	56
4.30	Eficácia do classificador Softmax com <i>features</i> sem redução de dimensionalidade - Intervalo de Confiança de 95% - VGG-16.	57
4.31	Eficácia do classificador Softmax com <i>features</i> sem redução de dimensionalidade - Intervalo de Confiança de 95% - VGG-19.	58
4.32	Análise visual do particionamento da espécie (classe) <i>Prunus avium</i> com <i>features</i> extraídas pela RNC Inception V3 e classificadas com Softmax.	60
4.33	Análise visual do particionamento da espécie (classe) <i>Epilobium hirsutum</i> com <i>features</i> extraídas pela RNC Inception V3 e classificadas com Softmax.	61
4.34	Mapa de calor da Matriz de Confusão da Base Original com <i>features</i> extraídas com a RNC Inception V3 e classificada com Softmax.	63
4.35	Mapa de calor da Matriz de Confusão da Base Particionada com <i>features</i> extraídas com a RNC Inception V3 e classificada com Softmax.	64
4.36	Mapa de calor da diferença entres as Matrizes de Confusão da Base Original e Particionada com <i>features</i> extraídas pela RNC Inception V3 e classificadas com Softmax.	65

Lista de Abrebiações

Abreviação	Descrição
MCDCNN	Redes Neurais Convolucionais de Múltiplas Colunas (Multi-Column Deep Convolutional Neural Networks)
PCA	Análise de Componentes Principais (Principal Component Analysis)
RNC	Redes Neurais Convolucionais (Convolutional Neural Network)
SVM	Máquinas de Vetores de Suporte (Support Vector Machine)

Lista de Símbolos

Símbolo	Descrição
HS_i	Estatística de Hopkins calculada para uma determinada classe

Capítulo 1

Introdução

*“Um diamante não pode ser polido
sem fricção, nem o homem pode
aperfeiçoar-se sem sacrifícios”*

– Provérbio Chinês

O planeta terra tem passado por diversas transformações ao longo dos anos. Grande parte dessas mudanças está diretamente associada à ação humana em relação ao meio ambiente, especialmente as ações que impactam a vegetação em toda sua extensão. No mundo todo, cientistas, pesquisadores e estudantes da área da botânica já catalogaram milhares de espécies vegetais, cuja importância é expressa em estudos que comprovam tanto a vida vegetal como sendo a base de todas as cadeias alimentares, quanto sua essencialidade para existência da humanidade. Há uma grande diversidade de espécies de plantas, algumas delas são essenciais para a medicina, outras são nocivas ao homem e ainda há aquelas cuja existência encontra-se ameaçada de extinção. Diante disso, é fundamental a correta identificação das plantas para seu monitoramento, estudo, controle populacional e conservação das mesmas [Schikora e Schikora 2014].

Segundo os trabalhos de [Wu et al. 2007] e [Judd 2008], o estudo e a catalogação de espécies de plantas baseiam-se no trabalho de botânicos especializados. Estes profissionais utilizam-se de diversos métodos, incluindo a identificação por comparação de órgãos da planta, como, por exemplo, ramos férteis, flores ou frutos dissecados, prensados e armazenados em folhas de papel. Porém, em seu trabalho, Judd afirma que este método traz desafios quando é utilizado para analisar flores e frutos, porque estes apresentam grandes variações, e.g., devido ao estágio de maturação, o habitat ou época do ano [Judd 2008]. Por isso, existe uma grande dependência do conhecimento adquirido e das habilidades dos seres humanos no processo de classificação das espécies de plantas.

Fidalgo *et al* apresentaram os principais métodos para coleta, herborização ¹ e identificação de amostras botânicas, bem como os procedimentos para seu registro e depósito em herbários.

O processo manual de classificação de espécies de plantas é trabalhoso e consequentemente demorado, o que torna a identificação de espécies de plantas praticamente impossível para o público em geral e, muitas vezes, uma tarefa difícil até mesmo para profissionais [Anfeng He e Xinmei Tian 2016]. A tarefa manual de identificação de plantas é muito importante em diversos campos de pesquisa, como ecologia, biodiversidade, farmacologia e áreas afins. Um dos métodos mais tradicionais de identificação utiliza a comparação de órgãos da planta, como ramos férteis, flores ou frutos dissecados, prensados e armazenados em folhas de papel [Judd 2008]. Como mostra a Figura 1.1 a seguir.



Figura 1.1: Processo tradicional de identificação da espécie de uma planta por comparação de características morfológicas de amostras catalogadas em herbários.

Fonte: <http://g1.globo.com/sp/piracicaba-regiao/noticia/2013/10/acervo-de-130-mil-plantas-de-herbario-da-usp-em-piracicaba-vai-para-web.html>.

Além disso, a identificação correta das espécies vegetais necessita inicialmente de uma boa amostra coletada, que deve ser preferencialmente fértil. Após essa atividade, é necessário buscar na literatura chaves de identificação do grupo de interesse.

¹Série de processos de manuseio de material botânico passando por coleta, prensagem, secagem, identificação e montagem. Modo pelo qual se faz um herbário [Fidalgo et al. 1989].

Essas chaves permitem dar nomes às espécies pelas características morfológicas que estas exibem. Por meio do nome encontrado, é feita uma busca em coleções de herbários para conferir a aplicação adequada desse nome. Entretanto, muitas vezes as amostras depositadas em coleções não possuem flores nem frutos, e ainda carecem de uma descrição adequada das condições do habitat e da própria planta no momento da coleta. Com isso, a falta de informação pode dificultar o processo de identificação [Fidalgo et al. 1989].

Nestes cenários, uma abordagem considerada promissora é o desenvolvimento de ferramentas eficazes baseadas em técnicas de visão computacional e aprendizagem de máquina para auxiliar na identificação automática (ou semi-automática) de uma ampla cadeia de espécies de plantas. Neste processo, de modo geral, são extraídas características visuais a partir de imagens com o objetivo de identificação de padrões. As propriedades da planta numa dada imagem são representadas, por exemplo, por um vetor de características descritivas como formas, cores ou texturas. Esses vetores são utilizados pelos sistemas de identificação tanto na criação de modelos de classificação quanto na fase de identificação de novas amostras [Witten et al. 2011].

Diante disso, a identificação automática de espécies de plantas vem se tornando uma ferramenta decisiva não só para a botânica, mas em diversas áreas do conhecimento, por exemplo, na agricultura por meio da identificação de espécies invasoras, e na educação através da disseminação das aplicações para dispositivos móveis aplicadas às aulas de biologia na educação básica.

A identificação automática de plantas nas lavouras pode auxiliar na identificação de espécies de plantas invasoras e sua distribuição dentro do campo, o que permite a aplicação dos insumos químicos com a dosagem correta e em locais precisos [Hamuda et al. 2016]. Isto porque, como reforçam os estudos de [Gianessi e Reigner 2007], um dos métodos mais econômicos para o controle de plantas invasoras é por herbicidas, tornando-o o mais utilizado. Neste sentido, Hamuda *et al* afirmam ainda que o uso da tecnologia de reconhecimento automático dessas espécies de ervas daninhas viabiliza o mapeamento de sua população no campo, permitindo assim a aplicação precisa dos insumos, diminuindo os custos de aplicação e a contaminação do solo, bem como proporcionando economia de 30% a 75% no consumo de herbicidas [Hamuda et al. 2016].

Em outro contexto, um sistema de reconhecimento automático de plantas medicinais, seja sob uma plataforma Web ou para dispositivos móveis, poderia trazer contribuições à população local no processo de construção e disseminação do conhecimento em plantas medicinais, além de auxiliar os taxonomistas no desenvolvimento de técnicas de identificação de espécies que propiciem a produção de medicamentos fitoterápicos [Begue et al. 2017].

Os trabalhos de [Goëau et al. 2016], [Ghazi et al. 2015], [Goeau et al. 2017] e [Mehdipour Ghazi et al. 2017], mostram a crescente utilização de Rede Neurais Artificiais Convolucionais Profundas como alternativa para extração dos melhores padrões de imagens e no desenvolvimento de técnicas de reconhecimento automático

de imagens de plantas, mas ainda assim, os problemas de variabilidade intra-classe e inter-classe influenciam interferem nos resultado da classificação automática de plantas.

1.1 Justificativa e Motivação

Alguns dos principais desafios que fazem parte do reconhecimento manual ou automático de plantas estão ligados à variabilidade intra-classe e à similaridade inter-classe [Joly et al. 2014]. Neste sentido, o trabalho de [Goëau et al. 2012] apresentou casos de imagens de folhas de diferentes plantas da mesma espécie, mas que são visualmente muito diferentes, chamada de variabilidade *intra-classe*, como pode ser visto na Figura 1.2.



Figura 1.2: Variabilidade intra-classe: Diversidade de lóbulos de folhas de diferentes plantas da mesma espécie *Broussonetia papyrifera* (L.) L'H'er. ex Vent.

Fonte: [Goëau et al. 2012].

A variabilidade nas imagens de plantas de uma mesma espécie torna sua identificação ainda mais complexa, porque suas características morfológicas podem ser visualmente diferentes dependendo do seu estágio de crescimento, maturidade e hábitat, o que impacta em diferentes tamanhos, cores ou formatos de seus órgãos, como pode ser observado na Figura 1.3.

Por sua vez, a similaridade *inter-classe* indica que os órgãos de diferentes espécies de plantas podem assumir características morfológicas muito similares às de outras espécies encontradas em ecossistemas diferentes [Judd 2008]. Esta situação é exemplificada na Figura 1.4.

É comum a ocorrência desses problemas em coleções de imagens, dada a forma como essas imagens são coletadas, por exemplo, a partir do aplicativo Leafsnap², cujo conjunto de dados foi construído de forma colaborativa, por meio de uma iniciativa de ciências cidadãs com imagens cedidas por usuários via aplicativos web e dispositivos móveis [Joly et al. 2014].

As variabilidades intra-classe e inter-classe trazem problemas para os classificadores, por exemplo, para a descoberta dos padrões característicos de uma determinada

²Disponível em <https://apps.apple.com/br/app/leafsnap/id430649829>.



Figura 1.3: Variabilidade intra-classe: Quatro folhas de plantas distintas de uma mesma espécie (*Platanus x hispanica* Wild) coletadas em diferentes estágios de maturação.

Fonte: [Goëau et al. 2012].

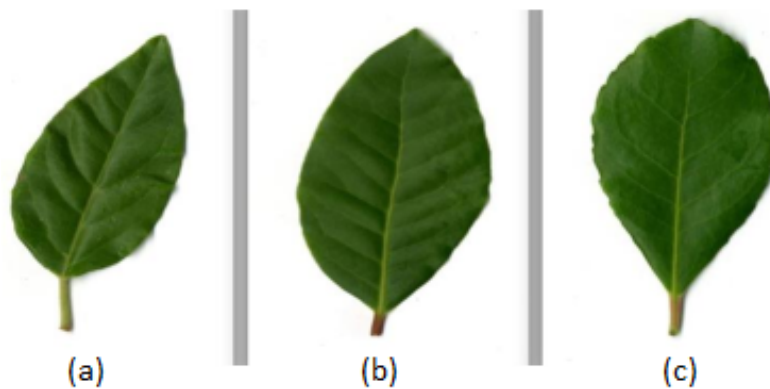


Figura 1.4: Similaridade inter-classe: Folhas de diferentes espécies de plantas com características morfológicas similares: (a) *Adoxaceae viburnum tinus*, (b) *Lauraceae laurus nobilis* e (c) *Ericaceae arbutus unedo*.

Fonte: [Judd 2008].

classe ou padrões que diferenciam uma classe de outra. Outro problema comum acerca da variabilidade intra-classe é que esta pode acarretar em subgrupos de instâncias que representam a mesma espécie, mas que tem características visuais distintas, dificultando a descoberta de padrões e conseqüentemente o processo de identificação.

1.2 Objetivos

Este trabalho tem como objetivo geral auxiliar estudantes, profissionais da botânica e pesquisadores no processo de identificação semiautomática de imagens de plantas e suas aplicações por meio de um estudo exploratório de técnicas de aprendizado

supervisionado e não supervisionado.

Especificamente a proposta é:

- Avaliar diferentes arquiteturas de redes neurais convolucionais para extração de *features*;
- Minimizar os impactos dos problemas de variabilidade intra-classe e inter-classe;
- Verificar o impacto da aplicação técnicas de agrupamento no particionamento de classes;
- Avaliar a eficácia de múltiplos classificadores considerando diferentes *features*.

1.3 Contribuições

Este trabalho visa contribuir com a comunidade científica, cujos estudos estão relacionados a taxonomia³ vegetal, mais precisamente no reconhecimento automático de espécies de plantas. Este trabalho traz como principais contribuições:

- Uma análise comparativa do uso de diferentes arquiteturas de redes neurais convolucionais para extração de *features*;
- Avaliação experimental de um método de particionamento de classes pré-classificação;
- Análise comparativa da eficácia de classificação de múltiplos classificadores sob as bases de dados original e particionadas;
- Avaliação do impacto das diferentes técnicas de aprendizado supervisionado e não-supervisionado para minimizar os problemas de variabilidade das classes.

³Na biologia, taxonomia é a ciência biológica que define os grupos de organismos biológicos, com base em suas características comuns e atribui nomes a esses grupos [Fidalgo et al. 1989].

Capítulo 2

Fundamentação Teórica

*“Nenhum problema pode ser
resolvido a partir do mesmo nível
de consciência que o criou.”*

– Albert Einstein

Este capítulo apresenta os conceitos fundamentais associados a este trabalho e está organizado como segue. A Seção 2.1 apresenta o conceito e algumas aplicações de técnicas de reconhecimento de padrões. A Seção 2.2 descreve os conceitos de classificação de dados, aprendizado supervisionado e não-supervisionado, tipos de classificadores, medidas de avaliação de eficácia, redes neurais convolucionais profundas e análise de componentes principais (PCA). Por fim, a Seção 2.6 apresenta trabalhos relacionados a classificação automática de imagens de plantas sob dois aspectos: Aprendizado de Máquina e *Deep Learning*.

2.1 Reconhecimento de Padrões

O reconhecimento de padrões é uma das áreas de pesquisa mais promissoras da atualidade em virtude do aumento massivo no volume de dados multimídia que hoje são publicados na Internet por usuários do mundo todo. O reconhecimento de padrões tem por objetivo a classificação de objetos, ou seja, identificar padrões em diferentes tipos de dados e associá-los à uma ou mais categorias [Ripley 2007, Corrigan 2007].

As técnicas de reconhecimento de padrões estão presentes em estudos de diferentes áreas do conhecimento. Isto se dá pelas contribuições que sua aplicação trazem na resolução dos inúmeros problemas de cada uma dessas áreas. Algumas destas aplicações são:

- Biometria: no reconhecimento facial, impressões digitais e íris;
- Busca e classificação de dados em bases de dados multimídia;
- Análise de padrões de consumo e sistemas de recomendação.
- Identificação de espécies de plantas;

Diante de possíveis aplicações de técnicas de reconhecimento de padrões, uma das principais questões de sua utilização está relacionada a representação dos dados. Na classificação de imagens por conteúdo, por exemplo, as propriedades visuais de uma imagem precisam ser extraídas e representadas como um vetor de características, também conhecidas como *features*. Isso é possível por meio de técnicas e algoritmos de processamento, cujo objetivo é tornar os dados discerníveis à aplicação de técnicas de reconhecimento de padrões [Torres e Falcão 2006].

2.2 Classificação de Dados

Classificação é uma tarefa frequentemente realizada durante a vida cotidiana. Isto porque desde os primórdios da humanidade tem-se o hábito de se dividir objetos em uma série de categorias de acordo com suas características, também conhecidas como atributos. O termo *classificação*, em seu significado mais puro, é exclusivo e excludente, ou seja, define que cada objeto deve ser atribuído exclusivamente a uma classe tornando-o excludente de outras [Bramer 2007].

A tarefa de classificação faz parte do nosso cotidiano por meio de algumas das muitas tarefas práticas de tomada de decisão que realizamos no dia-a-dia, como por exemplo, a atribuição de pessoas ou objetos a uma série de categorias [Bramer 2007]. Por exemplo:

- Clientes que provavelmente comprarão ou não comprarão um produto específico em um supermercado;
- Pessoas que estão em alto, médio ou baixo risco de adquirir uma determinada doença;
- Espécies de plantas que podem ser identificadas como pertencentes à horta doméstica ou não;

É possível a aplicação de técnicas de aprendizado de máquina, cujos modelos de classificação baseiam-se em algoritmos de aprendizado capazes de identificar se objetos pertencem ou não à uma determinada categoria. Nestes modelos, os dados são representados por vetores de características (*features*), no caso de imagens elas são descritas considerando-se, formas, cores e texturas. Após extraídas as *features*, os conjuntos de dados podem ser divididos em subconjuntos de treino e teste, assim o algoritmo é executado identificando padrões e testado com as demais amostras para

mensurar a eficácia de aprendizado e predição, possibilitando o ajuste de parâmetros [Witten et al. 2011]. Esse processo de aprendizagem de máquina pode ser executado segundo duas abordagens principais: Supervisionado ou Não-supervisionado. Estas duas abordagens são brevemente descritas a seguir.

2.2.1 Aprendizado Supervisionado e Não-Supervisionado

A aprendizagem supervisionada consiste em algoritmos que recebem como entrada conjuntos de dados pré-rotulados para que o computador possa aprender a classificar novos dados. Em outras palavras, na aprendizagem supervisionada, o conjunto de treinamento contém dados já rotulados, assim, o algoritmo tem a saída correta da tarefa para cada amostra de dado. Desse modo, é possível utilizar os rótulos originais para validar as próprias respostas do algoritmo e mensurar sua taxa de acerto com base em uma parte dos dados não utilizada no processo de treinamento [Louridas e Ebert 2016].

O aprendizado não supervisionado é baseado apenas nas *features* dos objetos da coleção, cujos rótulos são desconhecidos. Basicamente, o algoritmo deve aprender a "categorizar" ou rotular os objetos pelo critério de similaridade entre os seus atributos. De forma análoga, isso é como dar uma criança um conjunto de objetos geométricos e pedir-lhe que construa grupos com tais objetos de acordo com suas características comuns [Louridas e Ebert 2016].

2.2.2 Classificadores Binários *vs.* Multi-classe

No processo de classificação automática, a maior parte dos problemas de classificação caracterizam-se por um número de classes maior que dois. Diante disso, são desenvolvidas técnicas que possibilitem a decomposição desses problemas multi-classe em subproblemas de classificação binária, tendo em vista que por meio da divisão de uma tarefa de classificação multi-classe em sub-tarefas mais simples possa ser possível diminuir o número de erros produzidos no problema original. Além disso, é possível que algoritmos cuja natureza de classificação seja exclusivamente binária possam ser utilizados em tarefas de identificação multi-classe.

A decomposição de problemas multi-classe consiste em decompor um conjunto de dados com várias classes, três ou mais, em vários conjuntos de duas classes (binário). A cada um desses conjuntos de dados binários é aplicado um algoritmo de classificação, obtendo-se vários modelos de decisão binários, ou seja, um para cada conjunto. Posteriormente, na fase de teste (classificação), cada nova amostra é classificada pelos vários modelos de decisão binários criados anteriormente, e para cada amostra tem-se um conjunto de predições. A esse conjunto de predições é aplicada uma função de agregação, como por exemplo, a votação majoritária, para

a obtenção da predição final por meio da contagem de votos dos classificadores binários [PIMENTA 2005]. Existem duas abordagens tradicionais para decomposição Um-contra-todos e Todos-contra-todos, que são descritas a seguir.

O método de decomposição *todos-contra-todos* (ovo)¹ consiste na criação de subconjuntos de dados por meio de todas as possíveis combinações das classes do problema multi-classe, reunindo-as em pares, ou seja, um problema com três ou mais classes é transformado em vários subproblemas de duas classes. Cada um dos subproblemas binários gerados terá seu próprio modelo de classificação, porém esse modelo de classificação próprio é construído tendo por base um conjunto de treino de tamanho inferior ao conjunto de treino do problema multi-classe original [Fürnkranz 2002].

Por sua vez, o método de decomposição *um-contra-todos* (ovr)² considera cada uma das classes do problema multi-classe, selecionando-as uma por vez como sendo a classe principal, e a compara com todas as outras classes unidas (classe alternativa), desse modo, cada conjunto de treino binário é composto pelos exemplos da classe principal e pelos exemplos de todas as outras classes. Assim, os vários conjuntos de treinamento binários têm o mesmo tamanho do conjunto de treino multi-classe [Rifkin e Klautau 2004].

2.2.3 Redes Neurais Convolucionais Profundas

Apesar dos estudos relacionados às Redes Neurais Convolucionais terem se iniciado ainda na década de 80, o termo *Deep learning* tem sido mais utilizado nos últimos anos, dado que as arquiteturas de aprendizagem profunda estão se popularizando e alcançado cada vez mais sucesso e melhores resultados na detecção automática das características necessárias para a detecção e classificação dos objetos [Dyrmann et al. 2016, Sun et al. 2017].

O trabalho de [Lecun et al. 2015] descrevem a arquitetura de uma *Convolution Neural Network* (CNN) típica. Em suma, as CNN têm sua estrutura composta por três tipos de camadas: camadas convolucionais, camadas de agregação (*pooling*) e camadas totalmente conectadas (*fully-connected*), conforme descritas a seguir e representadas pela Figura 2.1.

Em cada uma das camadas convolucionais, a imagem de entrada em forma de uma matriz de dados, passa por um conjunto de filtros e pesos reguladores gerando um novo mapa de características. Esse novo mapa de características é então submetido a uma função de ativação não linear, como um ReLU, e segue para a camada seguinte da rede [Goodfellow et al. 2016].

Nas camadas de agregação, os mapas de características gerados pelas camadas convolutivas são agregados por meio de uma função invariante à localização, reduzindo

¹todos-contra-todos ou um-contra-um, do inglês *one-vs-one* (ovo).

²um-contra-todos ou um-contra-restante, do inglês *one-vs-rest* (ovr).

o tamanho dos mapas de características gerados pelas convoluções. Ao final, são adicionadas camadas totalmente conectadas (*fully-connected*), cujos pesos não são mais compartilhados. Assim, uma camada totalmente conectada faz com que as informações espaciais sejam removidas [Dyrmann et al. 2016].

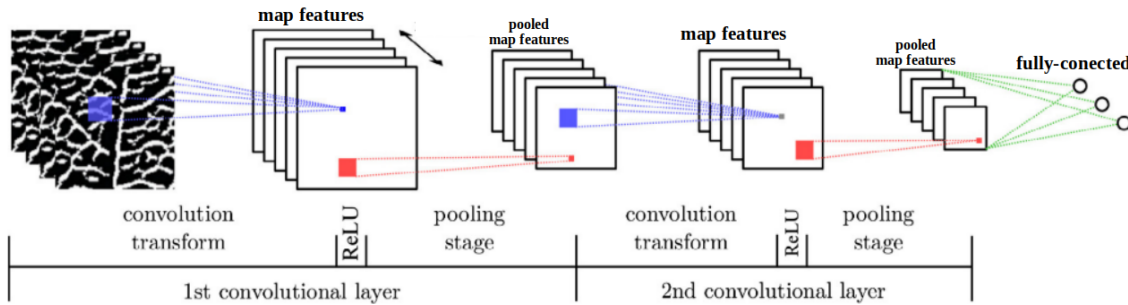


Figura 2.1: Estrutura de uma CNN composta por camadas convolucionais, de agregação (*pooling*) e totalmente conectadas.

Fonte: Adaptado de [Grinblat et al. 2016].

2.2.4 Extração de Características Utilizando *Deep Learning*

Diferente dos métodos clássicos de *Machine Learning*, em que as *features* são selecionadas manualmente e extraídas por meio de algoritmos específicos, as redes de aprendizado profundo descobrem automaticamente a partir dos dados características de nível cada vez mais alto [Lee et al. 2017]. Essas redes, por meio da utilização de grandes quantidades de dados e da exploração de arquiteturas paralelas com técnicas de computação de alto desempenho, são capazes de superar obstáculos que antes estavam associados a redes superficiais [Chen et al. 2017].

A utilização de *Deep Learning* para extração de *features* se dá pela capacidade das redes de aprender quais características são mais relevantes por meio de métodos de aprendizagem de representação com múltiplos níveis de abstração. Obtida pela composição de módulos simples, porém não lineares, a representação inicial, partindo da entrada dos dados brutos, é transformada em uma representação em um nível um pouco mais abstrato. Com a composição de tais transformações, funções muito complexas podem ser aprendidas. Para tarefas de classificação, camadas mais altas de representação amplificam aspectos da entrada que são importantes para a discriminação e suprimem variações irrelevantes [Lecun et al. 2015].

Uma imagem, por exemplo, vem na forma de uma matriz de valores de pixel, e as características aprendidas na primeira camada de representação, representam tipicamente a presença ou ausência de bordas em orientações e localizações particulares na imagem. A segunda camada, tipicamente detecta formas seguindo arranjos particulares de bordas, independentemente de pequenas variações nas posições da borda. A terceira camada pode agrupar padrões em combinações maiores que correspondem

a partes de objetos familiares, e camadas subsequentes detectariam objetos como combinações dessas partes [Lecun et al. 2015, Lee et al. 2017].

2.2.5 Arquiteturas de *Deep Learning*

Dentre as principais Redes Neurais Convolucionais Profundas presentes na literatura, as mais utilizadas em estudos de identificação de espécies de plantas são a GoogLeNet (conhecida como *Inception*) e a VGGNet, como pode ser observado nos resultados do desafio ImageCLEF dos últimos anos [Ghazi et al. 2015, Choi 2015, Hang et al. 2016, Goeau et al. 2017]. Estas duas arquiteturas de aprendizado profundo são descritas a seguir.

GoogLeNet (Inception)

De acordo com o trabalho de [Mehdipour Ghazi et al. 2017], a GoogLeNet é uma arquitetura *inception* e foi inspirada pela abordagem *network-in-network* descrita por [Lin et al. 2013]. Vencedora do ILSVRC 2014 [Russakovsky et al. 2015], possui aproximadamente 6,8 milhões de parâmetros com nove módulos de iniciação (*inception*), duas camadas convolucionais, uma camada convolucional para redução de dimensão, duas camadas de normalização, quatro camadas *max-pooling*, uma *average-pooling*, uma camada *fully-connected*, e uma camada linear com ativação Softmax na saída. Cada módulo de iniciação, por sua vez, contém duas camadas convolucionais, quatro camadas convolucionais para redução de dimensão e uma camada *max-pooling*. A GoogLeNet também usa ajuste por *dropout* na camada *fully-connected* e aplica a função de ativação ReLU em todas as camadas convolucionais [Szegedy et al. 2015].

VGGNet

A arquitetura de rede VGGNet foi apresentada no trabalho de [Simonyan e Zisserman 2014] e aplica uma arquitetura para investigar os efeitos do aumento da profundidade da rede convolucional no desempenho. A maior arquitetura VGGNet envolve 144 milhões de parâmetros, 16 camadas convolucionais com campos receptivos muito pequenos (3x3), cinco camadas *max-pooling* de tamanho 2x2, três camadas *fully-connected* e uma camada linear com ativação Softmax na saída. Este modelo também usa ajuste por *dropout* na camada *fully-connected* e aplica a ativação de ReLU a todas as camadas convolucionais. As terminologias “VGG16” e “VGG19” representam o número de camadas de pesos na rede.

2.3 Agrupamento de Dados

Os trabalhos de [Witten et al. 2011] e [Gorunescu 2011] definem agrupamento como técnicas que se aplicam a dados cuja classe não está predita, a fim de dividi-los em grupos naturais. Neste mesmo sentido, o processo de agrupamento de um conjunto de objetos, físicos ou abstratos, em classes de objetos semelhantes é chamado de *clustering*. Um *cluster* é um grupo de objetos de dados que são semelhantes entre si, e que são desejavelmente diferentes dos objetos de outros *clusters*.

O agrupamento de dados é feito por meio de algoritmos capazes de segmentar bases de dados em subconjuntos de forma que os objetos de cada grupo compartilhem características comuns, em geral, considerando sua proximidade dada alguma medida de similaridade ou distância. Idealmente, objetos pertencentes ao mesmo grupo são mais similares entre si do que a objetos pertencentes a grupos distintos. Assim, um grupo pode ser definido em função da coesão interna, homogeneidade, e do isolamento externo, separação, de seus objetos [Gorunescu 2011].

2.3.1 Tendência de Agrupamento

A tendência de agrupamento é uma pré-disposição dos dados à formação de grupos naturais. Isto significa que objetos com características semelhantes tendem a manter-se mais próximos devido a apresentarem distâncias menores entre suas características. A literatura contempla a estatística de Hopkins como um método bastante utilizado para a verificação dessa tendência de formação de *clustering*, porém o método não informa quantos *clusters* podem ser formados [Banerjee e Davé 2004].

A estatística Hopkins é um teste de amostragem esparsa para aleatoriedade espacial. Dado um conjunto de dados \mathbf{D} contendo n pontos, são geradas t subamostras aleatórias \mathbf{R}_i de m pontos cada, onde $m \ll n$. Essas amostras são extraídas do mesmo espaço de dados que \mathbf{D} , mas são geradas uniformemente ao acaso em cada dimensão. Além disso, também são geradas t subamostras de m pontos diretamente de \mathbf{D} , usando amostragem sem reposição. Deixe \mathbf{D}_i denotar a subamostra direta i -ésima. Em seguida, calcula-se a distância mínima entre cada ponto $\mathbf{x}_j \in \mathbf{D}_i$ e pontos em \mathbf{D}

$$\delta_{min}(\mathbf{x}_j) = \min_{\mathbf{x}_i \in \mathbf{D}, \mathbf{x}_i \neq \mathbf{x}_j} \{\delta(\mathbf{x}_j, \mathbf{x}_i)\} \quad (2.1)$$

Da mesma forma, calcula-se a distância mínima $\delta_{min}(\mathbf{y}_j)$ entre um ponto $\mathbf{y}_j \in \mathbf{R}_i$ e pontos em \mathbf{D} .

A estatística de Hopkins (em d dimensões) para o i -ésimo par de amostras \mathbf{R}_i e \mathbf{D}_i é então definida como

$$HS_i = \frac{\sum_{\mathbf{y}_j \in \mathbf{R}_i} (\delta_{\min}(\mathbf{y}_j))^d}{\sum_{\mathbf{y}_j \in \mathbf{R}_i} (\delta_{\min}(\mathbf{y}_j))^d + \sum_{\mathbf{x}_j \in \mathbf{D}_i} (\delta_{\min}(\mathbf{x}_j))^d} \quad (2.2)$$

Essa estatística compara a distribuição vizinha mais próxima de pontos gerados aleatoriamente com a mesma distribuição para subconjuntos aleatórios de pontos de \mathbf{D} . Se os dados forem bem agrupados, espera-se que os valores de $\delta_{\min}(\mathbf{x}_j)$ sejam menores em comparação com os valores de $\delta_{\min}(\mathbf{y}_j)$. Neste caso, HS_i tende a um. Se ambas as distâncias do vizinho mais próximo forem semelhantes, o HS_i assumirá valores próximos a 0,5, o que indica que os dados são essencialmente aleatórios e não há agrupamento aparente. Finalmente, se os valores de $\delta_{\min}(\mathbf{x}_j)$ são maiores em comparação com os valores de $\delta_{\min}(\mathbf{y}_j)$, então HS_i tende a zero, e indica repulsão de ponto, sem agrupamento. A partir dos diferentes valores de HS_i , pode-se calcular a média e a variância da estatística para determinar se \mathbf{D} é agrupável ou não [Biot e Academy 1977].

2.3.2 Qualidade de Agrupamentos e Coeficiente Silhueta

De acordo com o trabalho de [Rousseeuw 1987], o método de Silhueta é um índice que pode ser interpretado como o fator de pertinência dos objetos aos grupos, informando quão bem agrupados os objetos estão. Conforme exemplificado na Figura 2.2, a silhueta é uma medida interna inerente às características dos próprios objetos entre si, avaliando o quão semelhante o objeto é ao seu grupo e o quão diferente são os objetos de um grupo quando comparado aos demais grupos formados.

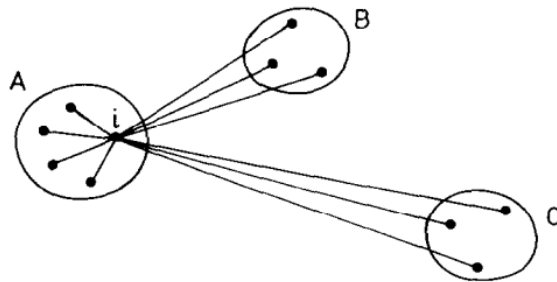


Figura 2.2: Ilustração dos elementos envolvidos no cálculo de silhueta do objeto i , $i \in A$.

Fonte: [Rousseeuw 1987].

Os valores da silhueta variam no intervalo $[-1,1]$. Quanto mais próximo de 1, mais o objeto é compatível com seu próprio *cluster* e diferente dos *clusters* vizinhos. Já quando os valores estão próximos de -1, isso indica que os objetos são homogêneos e há pouca formação (distinção) de *clusters*.

2.4 Medidas de avaliação

A literatura trás diferentes métricas que podem ser utilizadas na avaliação de eficácia de classificadores [Gorunescu 2011]. Para a avaliação da eficácia dos classificadores utilizados nos experimentos deste trabalho, foram adotadas as métricas de Acurácia, *Precision*, *Recall* e F1. A escolha dessas métricas foi baseada, entre outros, em trabalhos que estudam a classificação automática de plantas [Begue et al. 2017, Hamuda et al. 2016, Kumar et al. 2012].

O cálculo das medidas é obtido por meio dos valores fornecidos pela matriz de confusão, uma ferramenta utilizada na verificação de eficácia de um classificador na identificação das amostras de diferentes classes. A Figura 2.3 apresenta uma matriz de confusão genérica, ou seja, uma tabela com 4 combinações diferentes de classes reais e preditas. Os valores da matriz de confusão são representados da seguinte forma:

- *True Positive* (TP): itens positivos corretamente classificados;
- *False Positive* (FP): itens negativos classificados como positivos;
- *True Negative* (TN): itens negativos corretamente classificados;
- *False Negative* (FN): itens positivos classificados como negativos;
- *Positive* ($P = TP + FN$): número de casos positivos reais nos dados;
- *Negative* ($N = FP + TN$): número de casos negativos reais nos dados.

		Classe Predita	
		Positive	Negative
Classe Real	Positive	A	B
	Negative	B	A

		Classe Predita	
		Positive	Negative
Classe Real	Positive	A	B
	Negative	B	A

Figura 2.3: Matriz de Confusão

A acurácia é uma medida que expressa o percentual de predições corretas e pode ser obtida conforme a Equação 2.3. *Precision* é o percentual de amostras rotuladas como classe “A” que realmente pertencem a “A”, obtido pela equação 2.4. *Recall* é o percentual de amostras classe “A” rotuladas como classe “A”, obtido pela equação 2.5. F1 é a média harmônica combinando *precision* e *recall* por meio da equação 2.6

$$Accuracy = \frac{TP + TN}{P + N} \quad (2.3)$$

$$Precision = \frac{TP}{TP + FP} \quad (2.4)$$

$$Recall = \frac{TP}{TP + FN} = \frac{TP}{P} \quad (2.5)$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (2.6)$$

Para cada uma das medidas mencionadas, considerando um problema multiclasse, é possível calcular diferentes tipos de média, por exemplo, *Macro* e *Micro*.

A média *Macro* de uma dada medida, aqui representada por $B(tp, tn, fp, fn)$, pode ser computada da seguinte forma dada pela Equação 2.7 [Forman e Scholz 2010]:

$$B_{macro} = \frac{1}{q} \sum_{\lambda=1}^q B(tp_{\lambda}, fp_{\lambda}, tn_{\lambda}, fn_{\lambda}) \quad (2.7)$$

B_{macro} é computada pelo somatório das médias de B calculadas para cada uma das classes λ e dividido pela quantidade total de classes q . Porém, este tipo de média não é sensível ao desbalanceamento das classes uma vez que ela considera pesos iguais entre as médias calculadas para cada classe.

Segundo os trabalhos de [Fabrizio Sebastiani 2002], [Manning et al. 2008] e [Forman e Scholz 2010], as médias *Micro* computadas para uma dada medida B podem ser calculadas considerando a seguinte Equação 2.8:

$$B_{micro} = B \left(\sum_{\lambda=1}^q tp_{\lambda}, \sum_{\lambda=1}^q fp_{\lambda}, \sum_{\lambda=1}^q tn_{\lambda}, \sum_{\lambda=1}^q fn_{\lambda} \right) \quad (2.8)$$

B_{micro} é igual a medida B computada considerando o somatório dos valores tp , fp , fn e tn de todas classes λ . Deste modo, a medida torna-se mais sensível ao desbalanceamento das classes, pois a taxa de acerto é calculada pelo número total de amostras e não pelo número de classes.

Portanto, dado o problema de desequilíbrio entre o número de amostras por classe da coleção de imagens utilizada como objeto de estudo neste trabalho, optou-se pela utilização das *Micro* médias no computo das medidas de avaliação de classificação.

2.5 Redução de Dimensionalidade

O termo dimensionalidade é atribuído ao número de características de uma representação de padrões, ou seja, a dimensão do espaço de características. As duas principais razões para que a dimensionalidade seja a menor possível são: custo de medição e precisão do classificador. Quando o espaço de características contém somente as características mais salientes, o classificador será mais rápido e ocupará menos memória. Além disso, quando o conjunto de exemplos de treinamento não é muito grande, um espaço de características pequeno pode evitar a maldição da dimensionalidade e propiciar pequenas taxas de erro ao classificador.

Existem diversas abordagens na literatura que propõe técnicas de redução de dimensionalidade, como, *Random Forests / Ensemble Trees* [Breiman 2001], *High Correlation Filter* [Li et al. 2004], *Backward Feature Elimination* [Kohavi e John 1997] e a análise de componentes principais (PCA) [Tipping e Bishop 1997].

Dentre os métodos citados, o PCA é uma alternativa simples, que se caracteriza por uma abordagem estatística para encontrar as principais características de um conjunto de dados com base na variância total. Esta técnica é aplicada em diversas áreas de estudo, incluindo compactação de dados, análise de imagens, visualização, reconhecimento de padrões, regressão e previsão de séries temporais. Na área de análise de imagens, a PCA reduz o problema da alta dimensionalidade do vetor de características da imagem por meio da obtenção dos componentes principais. Esses componentes contêm as informações das características elementares da imagem original e são valores resultantes dos cálculos realizados com os autovetores e autovalores da matriz de covariância [Tipping e Bishop 1997, Ng 2017].

2.6 Classificação de Imagens de Plantas

A literatura contempla diferentes abordagens acerca da classificação automática de plantas, cujos estudos visam amparar o trabalho de taxonomia por meio de diferentes técnicas de aprendizagem de máquina e visão computacional [Psgr e Selvadoss Thanamani 2012]. Dentre essas técnicas, os trabalhos relacionados podem ser observados sob duas abordagens descritas nas Seções 2.6.1 e 2.6.2.

2.6.1 Aprendizado de Máquina para Identificação de Plantas

O trabalho de revisão de [Hamuda et al. 2016], apresenta diferentes abordagens do processo de segmentação de imagens de folhas, isto porque elas possuem características muito importantes para diferenciação entre as espécies, como por exemplo do

formato da folha, cor e textura. Além disso, os autores abordam diversos desafios que justificam os estudos de técnicas que possibilitem o reconhecimento automático de plantas, principalmente no campo da agricultura. Em seguida são apresentadas, analisadas e avaliadas técnicas de segmentação baseadas em índices de cores, limiar e aprendizagem. Essas técnicas classificam, dentro da imagem, as áreas de interesse. Neste caso, os *pixels* são classificados como planta e não-planta, ou seja, separando o objeto do plano de fundo da imagem, cuja relevância é crucial para o processo de extração de características e identificação das espécies nas imagens a serem analisadas. O autor conclui que a segmentação baseada em cores possui os melhores resultados em relação à segmentação baseada em limiar e aprendizagem, pois sofrem menos interferências de imagens obtidas em condições adversas de luminosidade.

Utilizando técnicas tradicionais de Aprendizado de Máquina, tais como, classificadores *Support Vector Machines* (SVM) com kernel RBF e *K-Nearest Neighbors* (k-NN) [Kumar et al. 2012] desenvolveram uma ferramenta chamada *Leafsnap*. O sistema é um aplicativo móvel cujo objetivo é ajudar os usuários a identificar espécies de árvores a partir de fotografias de suas folhas. O banco de dados de pesquisa da ferramenta consiste em *features* extraídas de 23.915 imagens de laboratório (tratadas) de folhas escaneadas e 5.192 imagens de campo obtidas por dispositivos móveis de 184 espécies de árvores do nordeste dos Estados Unidos. O sistema realiza quatro etapas principais: i) definição da imagem como folha e não-folha através de um classificador binário SVM com kernel RBF; ii) segmentação da imagem com base no sistema de cores HSV para destacar a folha do fundo da imagem; iii) extração do vetor de características utilizando um descritor de forma *Histograms of Curvature over Scale*(HoCS); e iv) classificação por meio da execução do algoritmo (k-NN). Os resultados revelaram que os histogramas de curvatura são descritores de forma eficazes, o que torna prático e simples a recuperação usando vizinhos mais próximos.

O trabalho de [Begue et al. 2017] também realizou estudos com técnicas tradicionais de Aprendizado de Máquina por meio de experimentos com múltiplos classificadores: *k-NN*, *Naïve Bayes*, *SVM*, *Random Forest* e Redes Neurais Artificiais. A coleção utilizada pelos autores dispõe de imagens de plantas medicinais da ilha tropical de *Mauritius* localizada no Oceano Índico, próximo à costa africana (cerca de 800 km), contendo 720 imagens de 24 espécies encontradas na ilha. O principal objetivo da pesquisa foi o desenvolvimento de um método totalmente automatizado para o reconhecimento de plantas medicinais que ajudasse a população local a melhorar seus conhecimentos sobre plantas medicinais. A etapa de pré-processamento foi realizada com a segmentação das imagens das folhas convertidas para o sistema de cores HSV e submetidas à aplicação do filtro de *Otsu* para binarização da imagem. A extração das *features* das folhas foi realizada utilizando o descritor de forma HoCS. Dentre os algoritmos de classificação utilizados no estudo, destacou-se como melhor classificador a *Random Forest* obtendo 90,1% de acurácia.

2.6.2 *Deep Learning* para Identificação de Plantas

Os trabalhos de [Mehdipour Ghazi et al. 2017] propuseram a identificação automática de imagens de plantas usando redes neurais convolucionais profundas via otimização dos parâmetros de *transfer learning* para extração de características e classificação. O conjunto de dados utilizado faz parte do desafio internacional LifeCLEF 2015 [Concetto et al. 2015] composto por 1.000 espécies de árvores, ervas e samambaias, contendo imagens de diferentes órgãos da planta, por exemplo, flores, frutas, folhas e caules. O dataset foi subdividido em dois conjuntos de dados, um contendo 91.758 imagens de treinamento e o outro com 21.446 imagens de teste. Para o estudo, os autores utilizaram três redes neurais convolucionais profundas a AlexNet [Krizhevsky et al. 2012], GoogLeNet [Szegedy et al. 2015] e VGGNet [Simonyan e Zisserman 2014]. A VGGNet obteve os melhores resultados com 78,44%. Entretanto, o treinamento da AlexNet a partir do zero superou VGGNet. Assim, foi possível verificar que apesar das vantagens do método de *transfer learning*, as redes treinadas a partir do zero obtiveram eficácia superior.

Já os trabalhos de [Anfeng He e Xinmei Tian 2016], por sua vez, apresentaram um modelo de redes neurais convolucionais profundas de várias colunas - *Multi-Column Deep Convolutional Neural Networks* (MDCNN) - para identificação automática de plantas. A base de imagens utilizada foi obtida da LifeCLEF 2014 [Joly et al. 2014] composta de 60.961 imagens multi-órgão, sendo sete tipos diferentes: planta inteira, caule, flor, fruto, galho, folhas e folhas escaneadas. A amostragem foi feita com 47.815 imagens para treinamento e 13.146 imagens para teste. Seu modelo de classificação treina sete modelos da rede neural AlexNet, em que cada uma das redes tem seus pesos reajustados através do método *fine-tuning* separadamente com um tipo de órgão, em seguida os modelos são combinados e interligados por duas camadas totalmente conectadas para treinar e compor o modelo integrado, como é apresentado na Figura 2.4. Os autores concluíram que ao testar o modelo com imagens de vários órgãos de uma planta desconhecida, em vez de apenas um único órgão, os resultados de eficácia de identificação da planta são melhorados significativamente em relação as Top-20 melhores técnicas das propostas desenvolvidas no contexto da LifeCLEF 2014 [Joly et al. 2014], com resultado médio de acurácia de 50,8%.

Em outros contexto, o estudo de [Sladojevic et al. 2016] descreve um trabalho baseado no reconhecimento automático de doenças de plantas através de imagens de folhas. A base de imagens utilizada no modelo proposto era composta por 4.483 imagens de folhas saudáveis e doentes de 4 tipos de cultivo de frutas, com 15 classes de doenças de plantas. A rede neural convolucional utilizada foi a CaffeNet [Jia et al. 2014]. Neste modelo a rede foi treinada com a própria base de dados. Desta forma, os resultados obtidos foram capazes de identificar 13 tipos de doenças de plantas com uma acurácia de 91%. Utilizando também a CaffeNet, [Barré et al. 2017] propôs um modelo chamado LeafNet, sendo um sistema de aprendizado profundo para descoberta características discriminativas de imagens de folhas de plantas. As bases de dados utilizadas nos experimentos foram a *LeafS-*

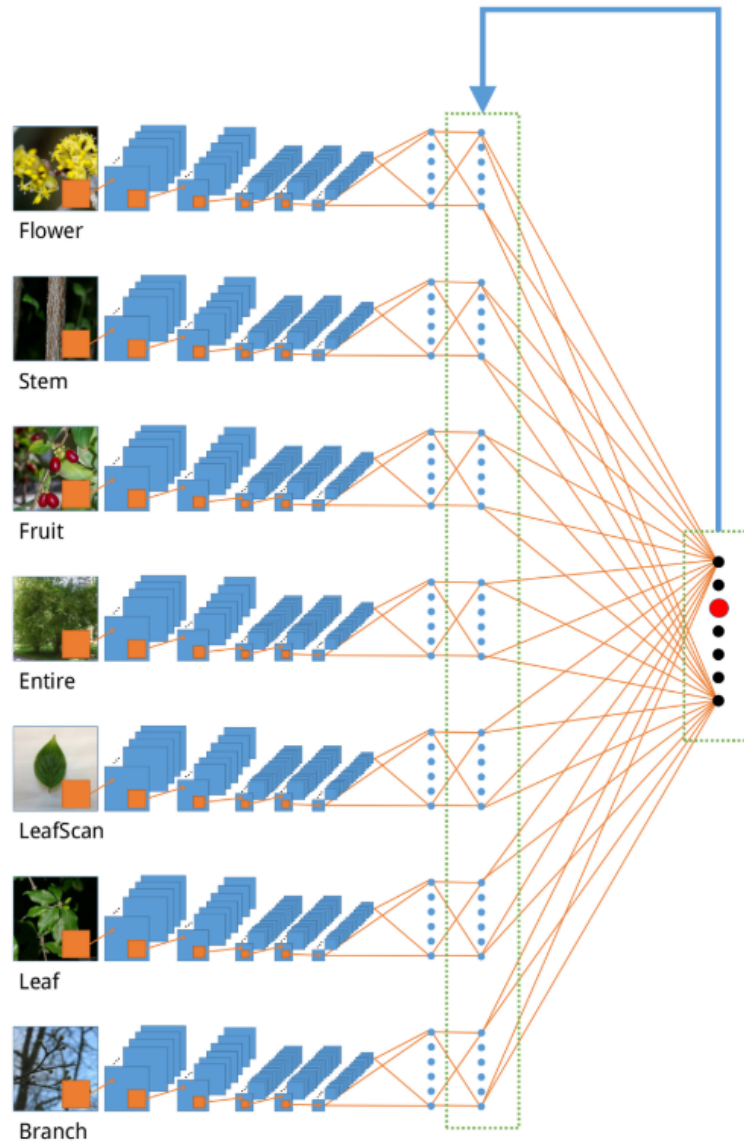


Figura 2.4: Estrutura das MCDCNN proposta por [Anfeng He e Xinmei Tian 2016].
 Fonte: [Anfeng He e Xinmei Tian 2016].

nap [Kumar et al. 2012], *Flavia* (com 32 espécies, onde um número de imagens por espécie varia de 50 a 60 amostras), e a *Foliage* que dispõe de imagens de 60 espécies, sendo 120 imagens para cada uma. Os três conjuntos de dados foram testados separadamente. Seus resultados foram comparados com um modelo SVM e os resultados indicaram melhor eficácia do modelo LeafNet com 86,3% de acurácia média, contra 82,3% do modelo SVM.

Mesmo com avanços proporcionados pelo desenvolvimento de técnicas cada vez mais eficazes no âmbito da identificação automática de plantas, percebe-se que ainda é desafiador o desenvolvimento de métodos capazes de lidar com a variabilidade intra-

classe, onde o processo de extração de características das imagens, como por exemplo das folhas, é dificultado devido à ocorrência de fatores que modificam suas características morfológicas, como a variação do clima, o nível de maturidade, a presença de fungos ou a ação de insetos. Além disso, o trabalho de [Begue et al. 2017] ressalta um outro problema, a similaridade interclasse, que, ainda levando em consideração o exemplo das folhas, resulta na ocorrência de folhas de espécies de plantas diferentes apresentando características similares entre si. Isso pode ser atenuado, e.g., com a utilização de imagens de diferentes órgãos da planta, como as flores, frutos, caule, galhos, entre outros para facilitar a identificação da espécie.

A partir do cenário aqui descrito, este trabalho propõe um estudo experimental sobre o problema de variabilidade intra-classe no contexto de classificação de imagens de plantas. Para isso, considera-se o *framework* comum de predição baseado em *features* extraídas por meio de redes neurais convolucionais profundas e que tem alcançado resultados promissores em diversas aplicações, incluindo o reconhecimento de espécies de plantas. Assim, investigamos a possibilidade de descoberta de subclasses implícitas em coleções de imagens de plantas e o consequente impacto de eficácia do processo de predição a partir destas subclasses.

Capítulo 3

Metodologia Proposta

“Faça o teu melhor, na condição que você tem, enquanto você não tem condições melhores, para fazer melhor ainda!”

– Mario Sergio Cortella

A metodologia proposta nesse trabalho visa explorar a utilização de diferentes arquiteturas de RNC's para extração de *features* combinadas com abordagens de classificação mais abrangentes para o reconhecimento automático de imagens de plantas. Além disso, a proposta também explora o uso de técnicas de aprendizado não-supervisionado, particionando as classes por meio do agrupamento de suas amostras com características similares, cujo objetivo é minimizar os impactos da variabilidade intra-classe e da similaridade inter-classe no processo de classificação da base de imagens objeto do estudo.

3.1 Visão geral da Proposta

A metodologia proposta parte da extração das *features* de uma base de dados de imagens de plantas, em seguida para cada uma das classes é feita uma verificação da pré-disposição das amostras em formar grupos por critério de similaridade. Depois, as classes propensas ao agrupamento interno das amostras são particionadas. Estas novas subclasses irão formar a base particionada com o mesmo número de amostras da base original, porém com um número maior de “classes”. Considerando a alta dimensionalidade das *features* extraídas inicialmente, é feita uma redução de dimensionalidade para que então as bases original e particionada passem pela etapa de classificação. Por fim, é feita a avaliação comparativa da eficácia dos classificadores para ambas as bases.

3.1.1 Fluxograma Detalhado da Proposta

Nesta seção são descritas as etapas de desenvolvimento da proposta, partindo da descrição da coleção de imagens, extração de características, particionamento, classificação e modelo de avaliação. Para avaliação experimental, o processo de classificação é executado sob duas perspectivas, SEM o particionamento e COM o particionamento, de modo que seja possível a comparação dos resultados para avaliação dos impactos do particionamento na classificação automática de plantas.

A Figura 3.1 apresenta o fluxograma da proposta, onde pode-se observar sequencialmente as ações realizadas no processo de avaliação experimental.

A primeira ação importante do fluxo é a extração das *features* da base de dados, neste caso, a coleção de imagens PlantCLEF 2013. Para a execução do processo de extração das *features* foram utilizadas três redes neurais convolucionais, Inception V3, VGG-16 e VGG-19, e, com as três diferentes *features* extraídas o particionamento é feito separadamente para cada uma delas.

Seguindo a extração de *features*, as classes do conjunto de dados são submetidas à uma análise da estatística de Hopkins para avaliação da tendência natural de agrupamento dos dados internos de cada classe.

Após análise da tendência de agrupamento natural, as classes, cujos valores indicam agrupamento natural intraclasse, são particionadas por meio do algoritmo de agrupamento *K-Means*. Nesta etapa de particionamento das classes, o número de subclasses geradas para cada uma das classes é definido pelo método de Silhueta. Finalizado o particionamento, as subclasses geradas irão compor uma nova base de dados com o mesmo número de amostras da base original, porém agora assumindo um número maior de classes, por exemplo, uma única classe da base original pode resultar em 2 ou mais na base particionada.

Levando-se em consideração a alta dimensionalidade das *features* obtidas com cada uma das redes neurais convolucionais utilizadas, é feita uma redução de dimensionalidade por meio de uma análise de componentes principais considerando 90% de variância dos atributos.

A etapa de classificação foi realizada utilizando-se de diferentes variações do classificador SVM, o método de *Random Forest* e o classificador *Softmax*. Esta etapa aplica-se tanto para a base PlantCLEF 2013 quanto para PlantCLEF 2013 particionadas com base em cada uma das três diferentes *features* obtidas na etapa de extração.

Na etapa de avaliação, para quantificar os resultados da eficácia dos classificadores foi adotado o computo de médias do tipo micro, isto porque, segundo o livro de [Manning et al. 2008], as micro-médias são medidas eficazes quando há o desbalanceamento das classes, tendo em vista que este tipo de média tende a considerar as classes por iguais. Em seguida, são comparados os resultados da classificação da base original com a base particionada para cada uma das *Features*.

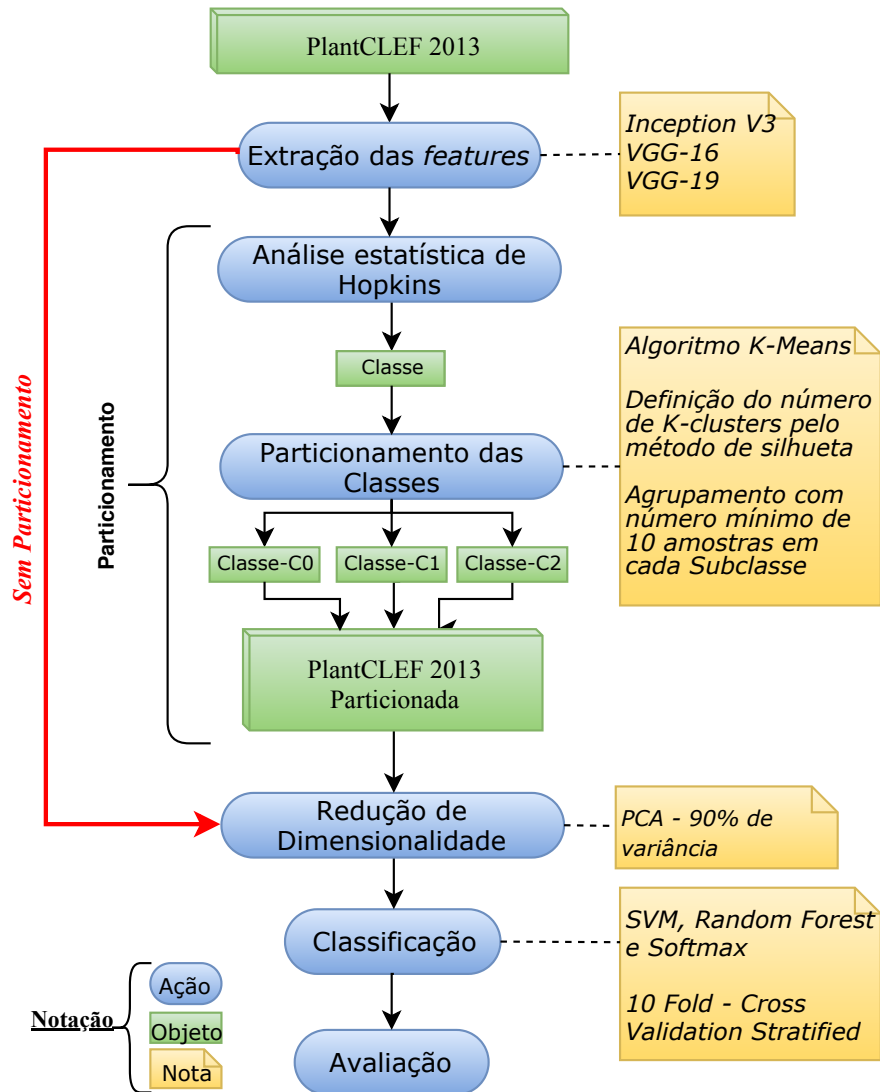


Figura 3.1: Fluxograma geral da proposta

3.2 Coleção de Imagens

O conjunto de dados da ImageCLEF 2013 [Goëau et al. 2013], aqui chamado IC2013, contém 26.077 imagens de 250 espécies de plantas da flora Francesa. O conjunto de dados é composto por 2 tipos principais de imagens: *SheetAsBackground* com 11.031 amostras, representando 42% do total de imagens e composta exclusivamente por fotos de folhas na frente de um fundo uniforme branco produzido com um escâner ou uma câmera com uma folha sobre uma superfície homogênea. E *Natural-Background* com 15.046 amostras representando 58% do total de imagens composta por fotografias naturais capturadas ao ar livre de diferentes ângulos e diferentes

órgãos de plantas na natureza. Detalhadamente, o subconjunto do tipo *NaturalBackground* é composto de folhas, flores, de frutos, de caules e de toda a planta). A Figura 3.2 apresenta exemplos das duas categorias principais e sete subcategorias: Scan (folhas escaneadas), Scan-like (folhas pseudo-escaneadas - fotografadas), fruto, flor, caule, ramos e toda a planta).

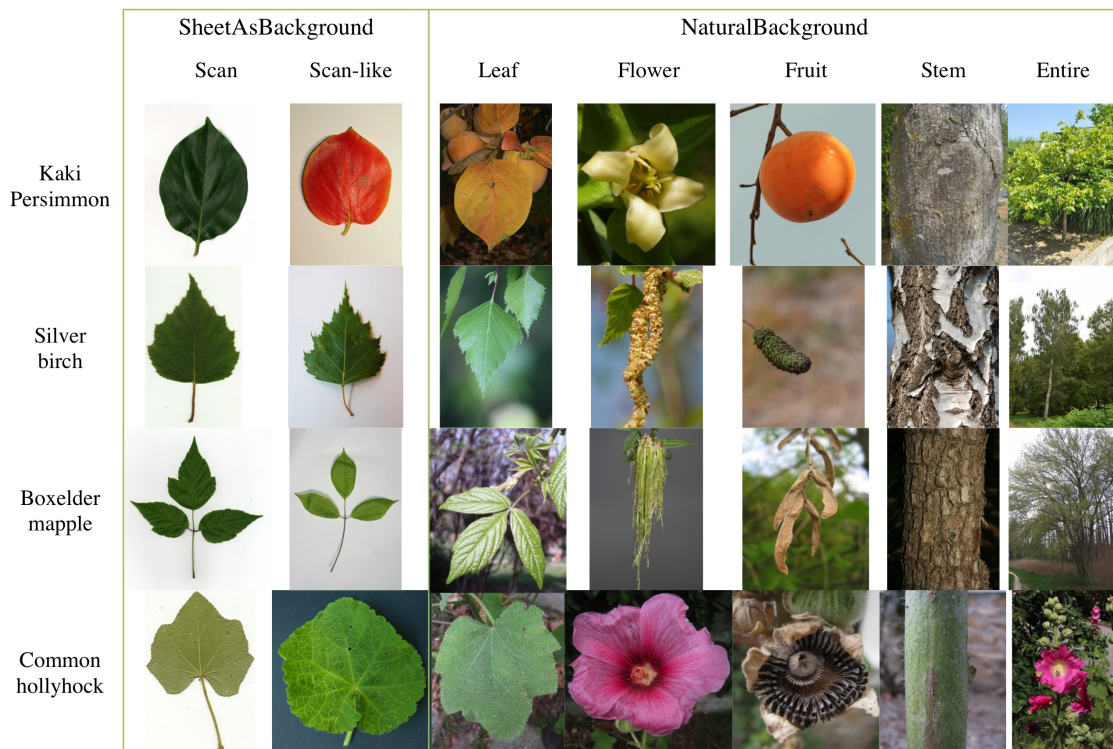


Figura 3.2: Exemplos das imagens dos múltiplos órgãos de diferentes espécies divididas pelas categorias *SheetAsBackground* e *NaturalBackground*.

Fonte: www.imageclef.org/2013/plant

Metadados

Cada imagem da coleção IC2013 possui metadados contendo, dentre outras, as seguintes informações principais:

- Tipo de Aquisição: *SheetAsBackground* ou *NaturalBackground*
- Conteúdo: Folha, Flor, Fruta, Caule, Inteiro.
- *IndividualPlantId*: o id da planta à qual várias imagens podem estar associadas.
- *Taxon*: Os nomes completos dos táxons (*Regnum*, *Class*, *Subclass*, *Superorder*, *Order*, *Family*, *Genus*, *Species*).

- ClassId: O rótulo de classe que deve ser usado como ground-truth.
- VernacularName : O nome comum em inglês.
- Data da Coleta
- Localidade: Nome da localidade (Cidade ou Região de um País).
- GPSLocality: Coordenadas GPS da localidade onde a planta foi observada.
- Autor: Nome do autor da imagem.
- Organização: nome da organização do autor.

3.3 Extração de Características

Conforme observado nos trabalhos das melhores equipes participantes do desafio ImageClef dos últimos anos, as rede neurais convolucionais profundas foram amplamente utilizadas no processo de extração de características e obtiveram os melhores resultados no reconhecimento automático de imagens de plantas [Ghazi et al. 2015, Choi 2015, Hang et al. 2016, Goeau et al. 2017]. Diante disso, utilizando a ferramenta Orange Data Mining, desenvolvida em Python e totalmente gratuita, foi possível obter os vetores de características de todas as imagens do conjunto de dados utilizando três redes neurais convolucionais profundas, Inception V3, VGG-16 e VGG19, disponíveis no componente *Image Embedding* do módulo *Image Analytics* da ferramenta.

De acordo com a documentação oficial do Orange [Biolab 2018], tecnicamente o componente *Image Embedding* envia a imagem para um servidor remoto, onde o servidor submete a imagem através da CNN pré-treinada com a ImageNet [Russakovsky et al. 2015] selecionada pelo usuário na aplicação. Porém, neste processo é desconsiderada a camada do classificador, considerando somente até a penúltima camada da rede para obter a representação da imagem baseada em vetores de características. Após esse processo o servidor retorna os vetores de características.

Nesta etapa, foram obtidas três diferentes *features* do conjunto de dados PlantClef2013, a primeira por meio da rede *InceptionV3* cuja penúltima camada possui 2048 dimensões (*features*), a segunda e a terceira por meio das redes *VGG16* e *VGG19* cuja penúltima camada de ambas possuem 4096 dimensões.

3.4 Descoberta de Subclasses

A técnica de particionamento proposta parte inicialmente de uma análise da tendência natural de agrupamento dos dados e em seguida é realizado o particionamento das classes por meio de um algoritmo de agrupamento interno. Os detalhes desta proposta são descritos nas seções a seguir.

3.4.1 Identificando a Tendência de Particionamento

Esta análise de tendência natural considera somente os vetores de características, ou seja, independentemente de categoria ou subcategoria (caso deste trabalho) da imagem. Isto se faz necessário porque segundo o trabalho de [Banerjee e Dave 2004] antes de submeter um conjunto de dados ao agrupamento, é preciso ter certeza se os dados exibem uma predisposição para agrupar-se em grupos naturais. Cada uma das classes (espécies de plantas) foi submetida ao algoritmo de análise estatística de Hopkins. Depois de obtidos os valores estatísticos de Hopkins para cada uma das classes, foi definido um limiar (valor mínimo de Hopkins) para a escolha de quais classes são submetidas ao particionamento através do algoritmo de agrupamento. Já as classes que tem seus valores de Hopkins abaixo do limiar definido não são particionadas e continuam fazendo parte da base de dados como originalmente.

3.4.2 Método de Particionamento

Para a desenvolvimento do método de particionamento foi utilizada uma implementação do algoritmo KMeans com restrição de tamanho mínimo para os *clusters* encontrados, aqui chamado de KMeans-MinSize, conforme proposto no trabalho de [Bradley et al. 2000]. Nesta etapa de execução, o algoritmo foi configurado com tamanho mínimo de 10 amostras por *clusters*. Isto foi feito com o objetivo de garantir, na etapa classificação, que ao executar o *10-fold Cross Validation* estratificado, cada *fold* teria ao menos uma amostra de cada subclasse.

No que refere-se à quantidade de *clusters*, aqui chamado de *k-clusters*, foi feita uma otimização paramétrica do valor de “k” com valores no intervalo 2 até 10. Primeiro verifica-se por meio de um cálculo a quantidade máxima de *k-clusters* em que uma dada classe pode ser particionada, isto por que o algoritmo KMeans MinSize é configurado para garantir no mínimo 10 amostras em cada *cluster*. Por exemplo, uma classe com 35 amostras só pode ser particionada em no máximo 3 *clusters* com no mínimo 10 amostras cada. Este número máximo de *clusters* é chamado de KMax. Após verificado o KMax para cada classe, então a escolha do melhor “k” é definido pelo coeficiente de Silhueta e executado o particionamento das classes com cada valor possível de “k” para descoberta da configuração que maximize o coeficiente.

3.5 Seleção de Features

Em virtude da alta dimensionalidade, elevado número de amostras e do custo computacional para execução dos dos diferentes classificadores, foi utilizada a técnica de Análise de Componentes Principais (PCA) [Ng 2017, Tipping e Bishop 1997] com 90% de variância sobre as *features* extraídas com as redes profundas. A Tabela 3.1 exhibe o número de dimensões das *features* e o número de dimensões resultantes da técnica de PCA aplicada sobre cada uma delas.

RNC	Dimensões	PCA
Inception V3	2048	340
VGG-16	4096	527
VGG-19	4096	526

Tabela 3.1: Número de dimensões resultantes da Análise de Componentes Principais (PCA)

3.6 Abordagens de Classificação

Para avaliar a eficácia da predição utilizando as diferentes *features*, foram utilizados classificadores SVM com diferentes configurações de *kernel* e métodos de decomposição para problemas multiclasse, sendo *One-vs-One* e *One-vs-Rest*. Também foram utilizados classificadores naturalmente multiclasse como o Softmax e a *Random Forest*. O método de validação implementado nos classificadores foi o *10-fold Cross Validation* estratificado. O quadro dos classificadores, suas respectivas configurações e método de validação podem ser observados na Figura 3.3.

Protocolo	Algoritmo	Configuração	Multiclasse
10-Fold Cross Validation Estratificado	SVM	Linear	One-vs-Rest
			One-vs-One
		RBF	One-vs-Rest
			One-vs-One
		Polynomial	One-vs-Rest
			One-vs-One
	Sigmoid	One-vs-Rest	
		One-vs-One	
	Softmax	Weights - ImageNet Optimizer - adam Epochs - 100 Batch size - 200	Probability
	Random Forest	100 Trees	Decision Tree

Figura 3.3: Quadro-Resumo dos Classificadores utilizados para avaliação da técnica de particionamento.

3.7 Modelo de Avaliação Comparativa de Eficácia

A eficácia dos classificadores pode mensurada por diferentes medidas de avaliação, e estas, calculadas com base nos valores presentes na matriz de confusão. A matriz de confusão é uma tabela que mostra as quantidades de ocorrências de erro e acertos de predição entre as classes. Cada linha da matriz representa as instâncias de uma classe predita, enquanto que cada coluna representa as instâncias de uma classe real (ou vice-versa).

Diante disso, com o objetivo de comparar adequadamente os resultados das medidas de avaliação da eficácia do classificador tanto para a base original quanto para a base com classes particionadas, foi desenvolvido um modelo de avaliação baseado no redimensionamento da matriz de confusão das bases particionadas.

Considerando que a Matriz de Confusão possui tamanho ($N \times N$), sendo N quantidade de classes, é preciso redimensionar a matriz de confusão da base particionada por meio da soma dos valores de Verdadeiros Positivos (TP), Falsos Positivos (FP), Verdadeiros Negativos (TN) e Falsos Negativos (FN) de todas as subclasses para torná-la equivalente à matriz de confusão da base original.

A fim de exemplificar o modelo de avaliação proposto, simulamos a classificação de uma base de dados fictícia com 100 amostras no total, sendo que 50 pertencem a classe “A” e 50 a classe “B”. Além disso, simulamos a execução do particionamento das classes resultando em 4 subclasses, apresentadas com seus respectivos números de amostras: “A1” - 40, “A2” - 10, “B1” - 30 e “B2” - 20.

A Figura 3.4 apresenta as matrizes de confusão com valores simulados da classificação da base de dados fictícia em sua forma original e particionada. Podemos observar que o redimensionamento da matriz de confusão da base particionada é feito somando os valores TP, FP, TN e FN das subclasses, pois quando uma amostra é de A1 e o método classifica como A2 é computado como acerto de A na matriz redimensionada.

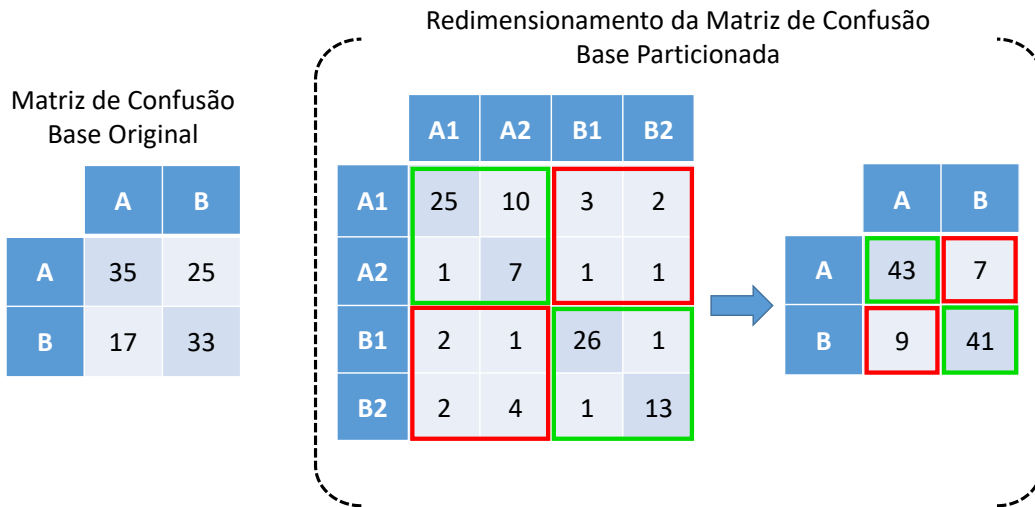


Figura 3.4: Exemplo de matrizes de confusão com valores simulados de classificação com adaptação da matriz de confusão para a base particionada.

3.7.1 Análise de Estabilidade e Significância Estatística

De modo a garantir uma análise adequada, detalhada e com rigor científico, os principais resultados de eficácia dos classificadores selecionados na etapa de classificação foram submetidos a 10 execuções com protocolo de validação *10-fold Cross Validation* estratificado. Em seguida, calculada a média dos resultados das 10 execuções, foi calculado o Intervalo de Confiança de 95% e realizado o teste *T-Student*, que verifica a hipótese nula de as médias serem ou não equivalentes.

3.8 Ferramentas Utilizadas

Para realizar os experimentos foram utilizadas a biblioteca de código aberto *Scikit-learn* [Pedregosa et al. 2011] que inclui uma ampla gama de algoritmos de aprendizado de máquina para problemas supervisionados e não supervisionados. A ferramenta Orange Data Mining [Demšar et al. 2013] que é um kit de ferramentas de código aberto para visualização de dados, aprendizado de máquina e mineração de dados, possuindo um *front-end* de programação visual para análise exploratória e visualização interativa de dados. Além disso, foi utilizada a ferramenta Keras [Chollet et al. 2015], uma API de alto nível para redes neurais sob plataforma online e gratuita com suporte a utilização de GPU do Google Colab [Carneiro et al. 2018].

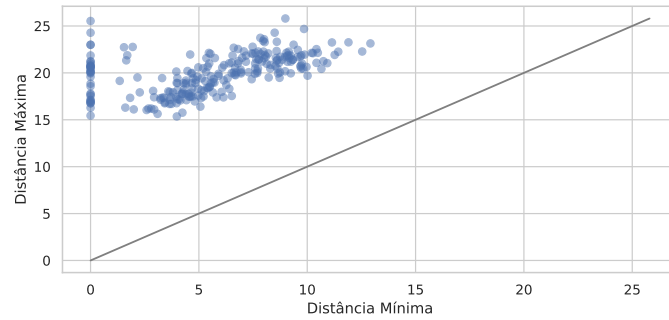
Capítulo 4

Resultados e Discussão

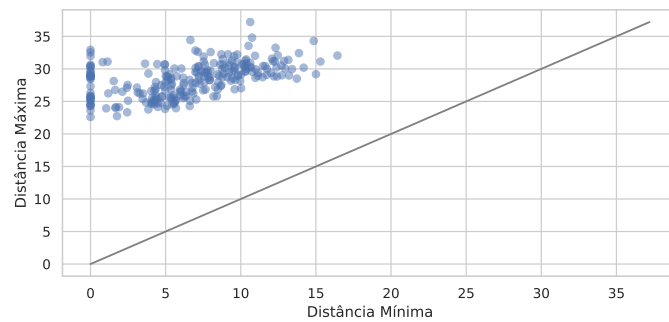
Neste capítulo são apresentados os resultados dos experimentos realizados no desenvolvimento, análise e avaliação dessa pesquisa. O capítulo está organizado em quatro principais seções. A Seção 4.1 faz uma caracterização do problema de variabilidade dos dados, a Seção 4.2 mostra a descoberta de subclasses, partindo da verificação da tendência natural de agrupamento das amostras contidas dentro de cada classe da coleção de imagens objeto do estudo. A Seção 4.3 trata da avaliação do impacto do particionamento no desempenho de diferentes classificadores sob cada uma das *features*, em seguida a Seção 4.5 apresenta amostras visuais do particionamento e das matrizes de confusão por mapa de calor do processo de classificação.

4.1 Caracterização do Problema de Variabilidade Intra-classe

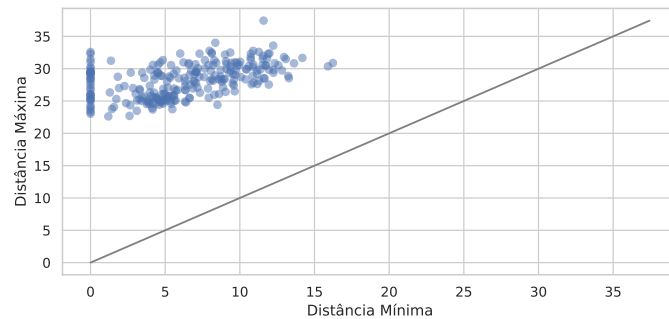
A Figura 4.1 apresenta os gráficos de dispersão de cada uma das três *features* utilizadas, considerando para cada classe, o valor mínimo e máximo de distância entre amostras de cada uma das 250 classes. Deste modo, observa-se que os pontos localizados no canto superior esquerdo indicam que existe uma variação entre a distância mínima e a distância máxima, o que sugere uma variedade de dissimilaridade entre as imagens daquela classe. As classes do canto superior esquerdo são aquelas em que algumas imagens são consideradas muito próximas em termos de distâncias, mas que ao mesmo tempo existem imagens dentro da mesma classe cujos valores de distância são altos.



(a) Dispersão dos valores mínimos e máximos de distância entre amostras com *features Inception V3*.



(b) Dispersão dos valores mínimos e máximos de distância entre amostras com *features VGG 16*.

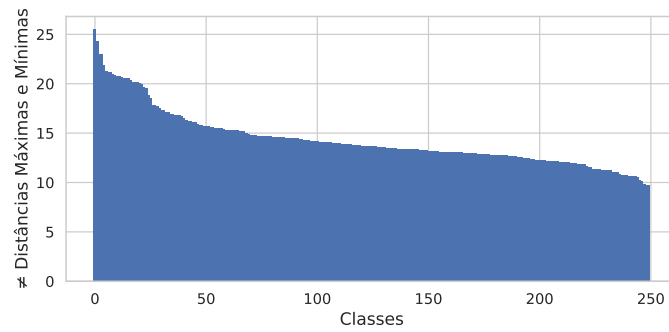


(c) Dispersão dos valores mínimos e máximos de distância entre amostras com *features VGG 19*.

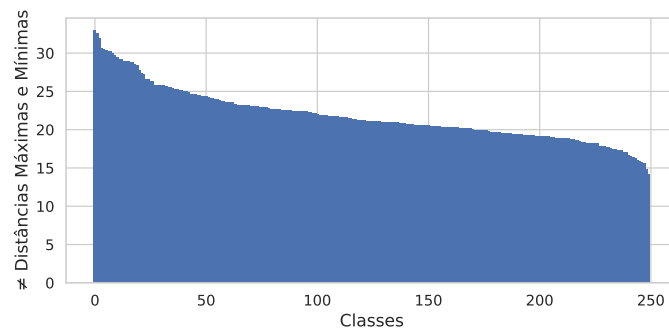
Figura 4.1: Dispersão dos valores mínimos e máximos de distância entre as amostras das classes considerando cada uma das *features*.

Tudo isso sugere uma variabilidade entre as imagens internas de cada classe, porém para quantificar essa variabilidade, a Figura 4.2 quantifica essa variabilidade considerando a diferença entre maior e o menor valor de distância dentro da classe. Neste caso, valores de diferença muito grande indicam que na mesma classe existem imagens muito parecidas e ao mesmo tempo muito diferentes. Nas gráficos (a), (b), (c) da Figura 4.2, podemos observar que para uma quantidade de classes grande, temos altos valores de diferenças entre os valores mínimos e máximos de distância,

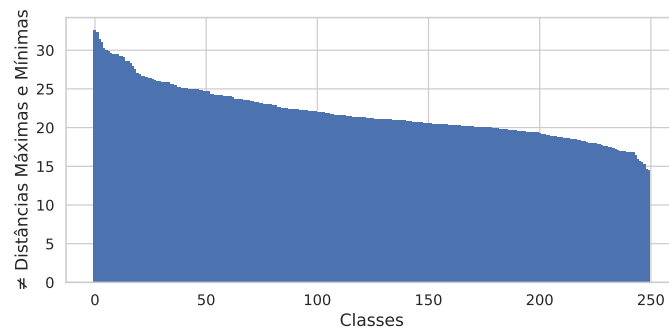
o que reforça a ideia de que existe variabilidade entre as amostras dessas classes.



(a) Diferença entre os valores mínimos e máximos de distância das amostras com *features Inception V3*.



(b) Diferença entre os valores mínimos e máximos de distância das amostras com *features VGG 16*.



(c) Diferença entre os valores mínimos e máximos de distância das amostras com *features VGG 19*.

Figura 4.2: Diferença entre os valores mínimos e máximos de distância das amostras de cada classe considerando cada uma das *features*.

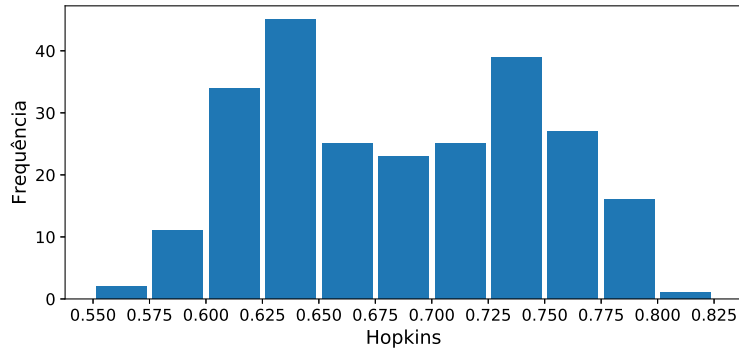
4.2 Avaliação da Descoberta de Subclasses

Nesta seção são apresentados os resultados da avaliação do particionamento, considerando os resultados da análise da estatística de Hopkins para o levantamento do número de classes com tendência de formação de grupos naturais e a análise da qualidade do particionamento em termos da eficácia de classificação para as subclasses geradas.

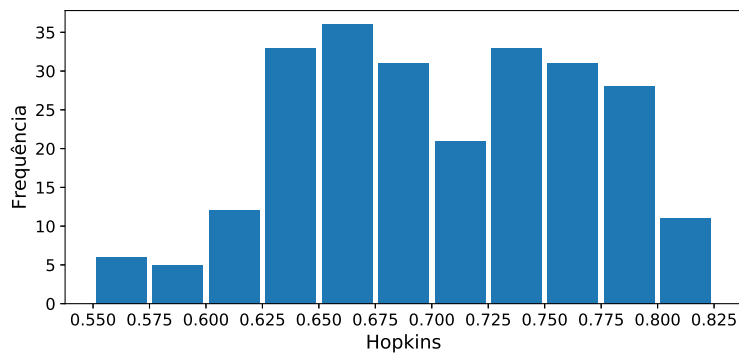
4.2.1 Tendência de Agrupamento

A análise estatística de Hopkins permite identificar quais classes do conjunto de dados estão propensas ao particionamento interno de suas amostras. Sendo assim, cada uma das classes (espécies de plantas) foi submetida ao algoritmo de análise estatística de Hopkins para avaliação da tendência natural de agrupamento. Seguindo aquilo indicado nos trabalhos de [Hopkins e Skellam 1954], conjuntos de dados cujos resultados da análise geram valores entre 0,7 e 1,0 indicam formação de grupos naturais. Por outro lado, esta análise também contribui para evitar o particionamento forçado de classes que não apresentam uma tendência de agrupamento significativa.

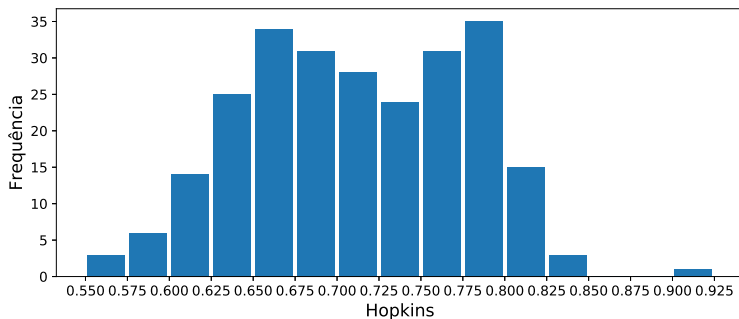
Diante disso, foi realizada a análise estatística de Hopkins em cada uma das classes considerando cada uma das *features* extraídas com as redes: (a) Inception V3; (b) VGG-16; e (c) VGG-19. Os resultados dessa análise são apresentados nos histogramas da Figura 4.3, onde são indicadas as frequências dos valores estatísticos em intervalos de 0,025.



(a) Análise Estatística de Hopkins - *Inception V3*.



(b) Análise Estatística de Hopkins - *VGG 16*.



(c) Análise Estatística de Hopkins - *VGG 19*.

Figura 4.3: Histogramas de frequência dos valores de estatística de Hopkins para cada uma das *features* utilizadas.

Após análise dos histogramas e considerando o intervalo cujos valores mostrados nos trabalhos de [Hopkins e Skellam 1954] indicam uma melhor formação de grupos naturais, definiu-se um limiar de 0,65 como critério mínimo para particionamento das classes em subclasses por meio do algoritmo de agrupamento. Deste modo, tem-se um maior aproveitamento em relação ao número de classes que apresentaram tendência de agrupamento e, conseqüentemente, uma pequena quantidade de classes com valores inferiores a este limiar.

4.2.2 Número de classes particionadas

Considerando a etapa de particionamento, após a análise estatística de Hopkins, foi feito um levantamento do número de classes “Não particionadas”, cujos valores na etapa da análise estatística de Hopkins ficaram abaixo do limiar 0,65 adotado como requisito de particionamento, e “Particionadas”, cujos valores ficaram acima do limiar adotado. O gráfico da Figura 4.4 permite a visualização dos resultados deste levantamento, bem como a observação da quantidade de classes que foram submetidas ao particionamento em relação às que não foram sob a ótica das três diferentes *features* utilizadas.

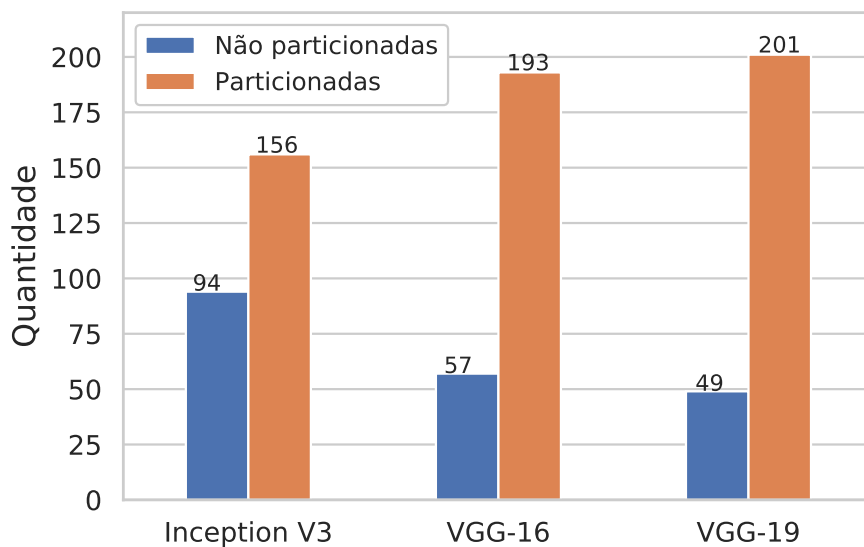


Figura 4.4: Levantamento do número de espécies agrupadas e não agrupadas para cada uma das *features* utilizadas.

Diante dos números apresentados na Figura 4.4, e considerando a variabilidade do método, observa-se que as duas *features* extraídas pelas redes VGG-16 e VGG-19 indicaram particionalmente mais classes, evidenciando o impacto direto da escolha da *feature* na etapa de particionamento.

4.2.3 Coeficiente de Silhueta

Com o objetivo obter um melhor particionamento para cada uma das classes, previamente selecionadas pela análise estatística de Hopkins, o número de *k-clusters* para cada uma das classes particionadas foi escolhido considerando o “k” com maior valor de silhueta. Lembrando que, o valor obtido pelo método de Silhueta permite avaliar o quão semelhante o objeto é ao seu grupo e o quão diferente são os objetos de um grupo quando comparado aos demais grupos formados. Podemos observar

nos histogramas das Figuras 4.5, 4.6 e 4.7 a frequência dos valores de silhueta para o melhor “k”, considerando a variabilidade das *features* utilizadas.

Na Figura 4.5 temos os valores de silhueta obtidos na análise determinante do número de grupos para o particionamento das classes utilizando as *features* da Inception V3, cujos valores dos intervalos de silhueta estão discretizados em intervalos de tamanho 0,025. Considerando que os valores de silhueta apresentados estão diretamente relacionados a escolha do melhor *k-cluster* no particionamento das classes, observa-se que os melhores “k” geraram valores de silhueta com maior frequência no intervalo [0.550,0.650], que segundo a literatura já indicam uma boa qualidade de particionamento.

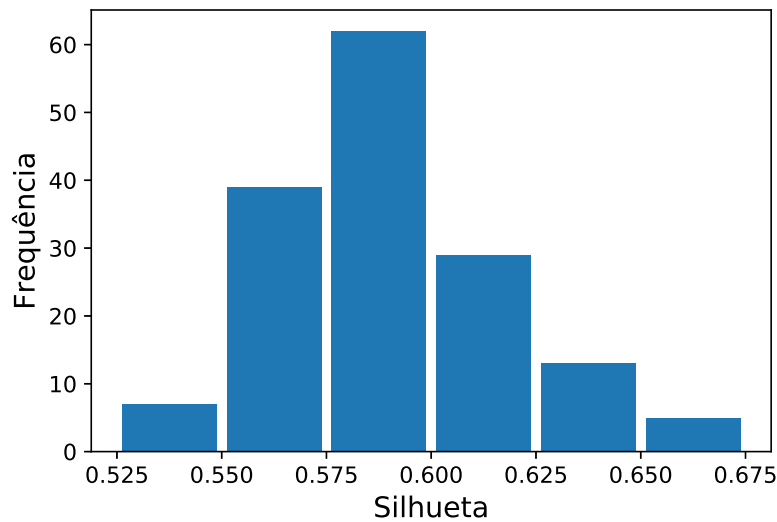


Figura 4.5: Frequência dos valores de silhueta utilizando *Features Inception V3*.

A Figura 4.6 apresenta os valores de silhueta obtidos na análise de determinação do número de grupos do particionamento das classes para as amostras com *features* extraídas pela VGG-16. Verifica-se valores mínimos de silhueta partindo de 0,500 a valores máximos de até 0,725. Porém, mesmo o intervalo partindo de um valor menor que o apresentado na Figura 4.5, ao observar a distribuição da frequência dos valores de silhueta, nota-se que a maior parte dos melhores “k” também geraram valores de silhueta com maior frequência no intervalo [0.550,0.650].

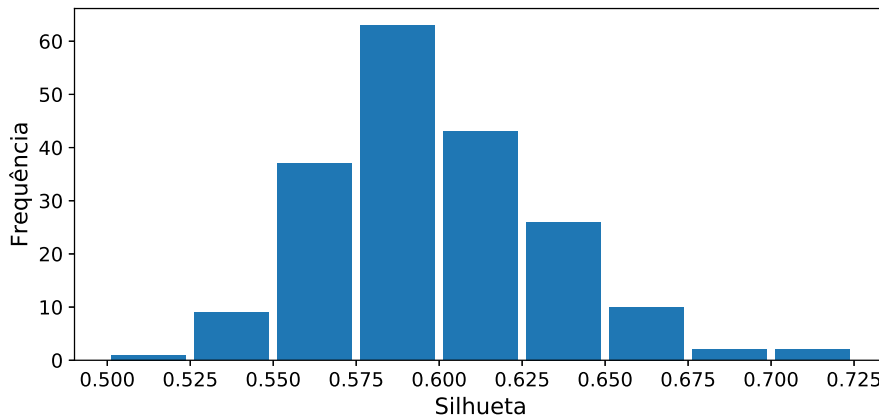


Figura 4.6: Frequência dos valores de silhueta utilizando *Features VGG 16*.

Por sua vez, a Figura 4.7 apresenta a frequência dos valores de silhueta dos melhores “k” utilizados no particionamento das classes com as amostras de *features* extraídas com a VGG-19. E ao observar o histograma, é possível notar um comportamento semelhante na distribuição das maiores frequências de silhueta no intervalo [0,550,0,650], assim como já visto nas Figuras 4.5 e 4.6.

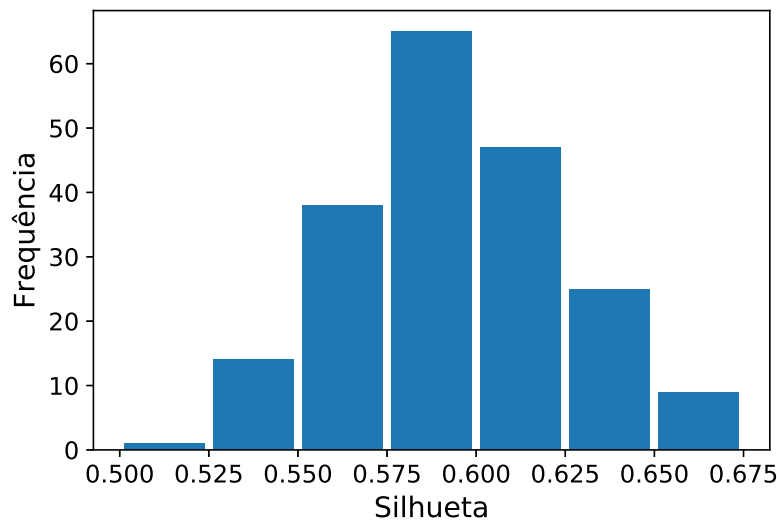


Figura 4.7: Frequência dos valores de silhueta utilizando *Features VGG 19*.

De modo geral, os valores de silhueta apresentados nos histogramas da Figuras 4.5, 4.6 e 4.7 apresentam valores de silhueta que partem de 0,500 até 0,725. Segundo o trabalho de [Rousseeuw 1987], uma vez que os valores da silhueta variam no intervalo $[-1,1]$, quanto mais próximo de 1, mais os objetos são compatíveis com seu próprio *cluster* e diferentes de seus *clusters* vizinhos. Diante disso, os valores de silhueta sugerem uma boa formação dos *clusters*.

4.2.4 Quantidade de Subclasses Descobertas

Considerando a metodologia já descrita na Seção 3.4.2, após a execução do particionamento, foi feita uma análise na quantidade de subclasses descobertas considerando a quantidade de classes pelo número de grupos(subclasses) escolhidos no particionamento. Esta análise considerou a variabilidade das *features* separadamente.

A Figura 4.8 apresenta a distribuição da quantidade de classes em relação ao número grupos para considerando as *features* extraídas pela Inception V3. Nesta distribuição podemos observar que a maioria das classes foi particionada em poucos grupos, sendo 83 classes com 2 grupos e 39 classes com 3 grupos, o que representa mais da metade das classes particionadas. Porém algumas classes foram particionadas em mais grupos, mas, percebe-se que o número de classes cai em média pela metade a partir de 4 grupos e nenhuma classe foi particionada com 10 grupos.

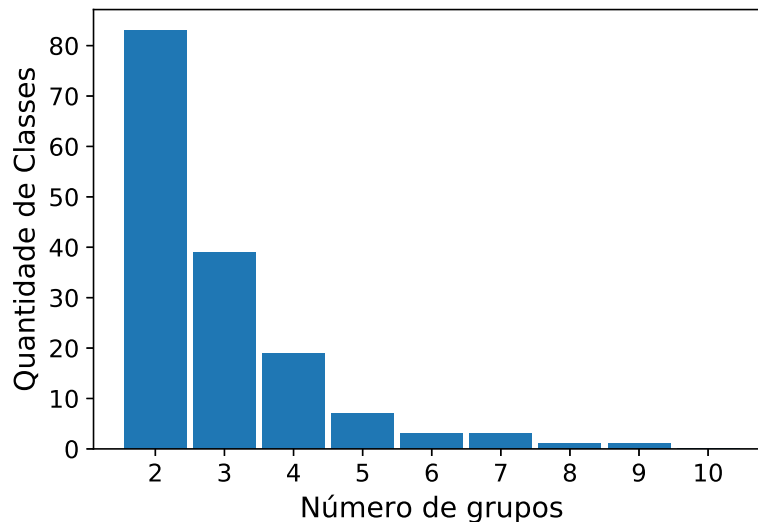


Figura 4.8: Quantidade de classes por número de grupos no particionamento das amostras com *Features Inception V3*.

Para as amostras com *features* extraídas pela VGG-16, a distribuição da quantidade de classes em relação ao número grupos é mostrado na Figura 4.9. Verifica-se que a maior parte das classes foram particionadas com apenas 2 grupos, um total de 136 classes, o que representa mais da metade da quantidade total de classes particionadas. Já a quantidade de classes particionadas com 3 grupos é de 28 classes, e esse número diminui aproximadamente pela metade conforme aumentam-se o número de grupos, como pode ser observado quando comparadas as quantidades de classes particionadas com 3, 4, 5 e 6 grupos. No mais, foram poucas classes particionadas a partir de 7 grupos e nenhuma classe com 10 grupos.

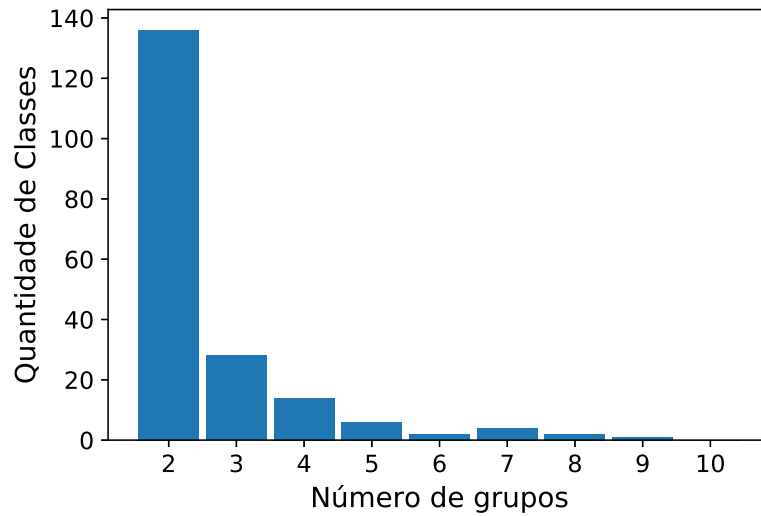


Figura 4.9: Quantidade de classes por número de grupos no particionamento das amostras com *Features VGG 16*.

A Figura 4.10 mostra a distribuição da quantidade de classes particionadas de acordo com o número de grupos das amostras com as *features* extraídas pela VGG-19. Observa-se que nesta distribuição, assim como nas anteriores, os valores da quantidade de classes também apresentam-se decrescentes em relação ao aumento do número de grupos, isto significa que a maior parte das classes foram particionadas com 2 grupos e que menos classes foram particionadas com 3 e até o limite máximo de 10 grupos. Porém, nesta distribuição podemos observar ao menos um classe particionada com 10 grupos, originando assim, 10 subclasses.

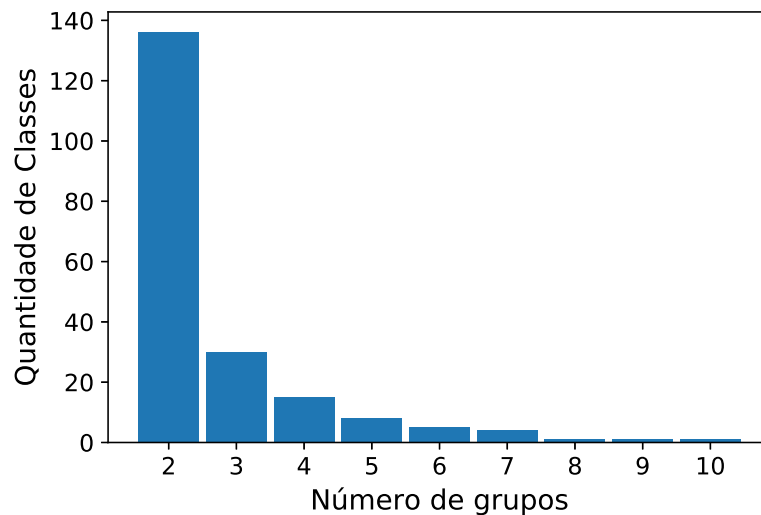


Figura 4.10: Quantidade de classes por número de grupos no particionamento das amostras com *Features VGG 19*.

De modo geral, percebe-se que a maior parte das classes foram particionadas com poucos grupos, apresentando em sua maioria a formação de apenas 2 subclasses para as três diferentes *features* utilizadas. Diante disso, viu-se a necessidade de avaliar se houve correção entre o número de grupos formados em relação ao número de amostras por classe, que será explanado na seção 4.2.5.

4.2.5 Relação entre o número grupos e quantidade de amostras presentes nas classes

Após análise da quantidade de classes particionadas com os diferentes números de grupos, observamos a necessidade de verificar a relação do número de grupos formados no particionamento das classes em relação à quantidade de amostras. O objetivo é verificar se houve correlação do número de amostras com o número de grupos uma vez que há um desbalanceamento no número de amostras nas classes.

A Figura 4.11 apresenta um gráfico dispersão do número de grupos pela quantidade amostras considerando o particionamento com as *features* Inception V3. A distribuição dos pontos não indica uma correlação direta. Observa-se que tanto classes com poucas amostras quanto classes com muitas amostras, em sua maioria, tiveram a formação em quantidades variadas de grupos, sendo majoritariamente particionadas de 2 a 4 grupos.

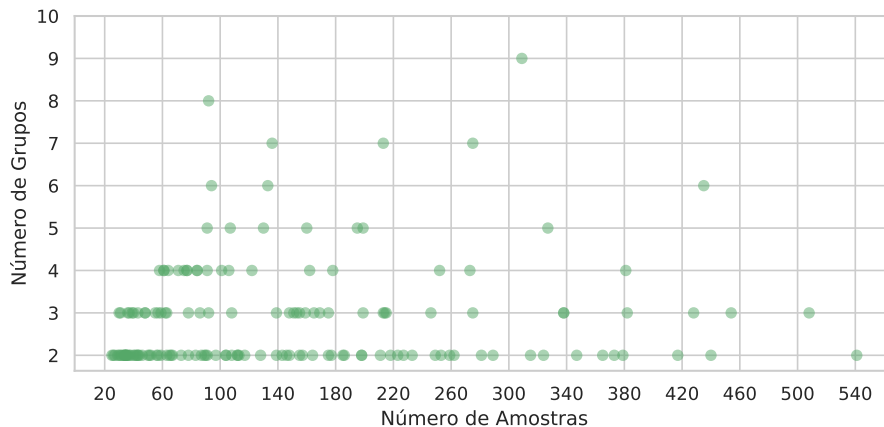


Figura 4.11: Relação do número de grupos pela quantidade de amostras considerando o particionamento com as *features* Inception V3.

A Figura 4.12 apresenta a relação do número de grupos pela quantidade de amostras considerando o particionamento com as *features* VGG-16. Assim, como já observado na Figura 4.11, os dados apresentados aqui também não indicam correlação do número de grupos gerados no particionamento com quantidade de amostras das classes. Mesmo assim, percebe-se uma maior concentração de classes particionadas com poucos grupos (2 ou 3 grupos) quando comparada com a distribuição na Figura 4.11.

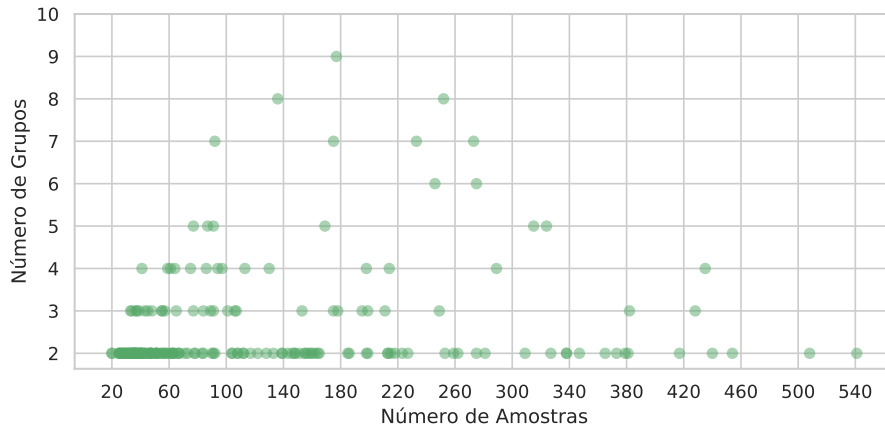


Figura 4.12: Relação do número de grupos pela quantidade de amostras considerando o particionamento com as *features* VGG-16.

A Figura 4.13 apresenta o gráfico de dispersão do número de grupos pelo número de amostras considerando o particionamento com as *features* VGG-19. Os resultados são similares àqueles apresentados para as *features* da Inception V3 e VGG-16.

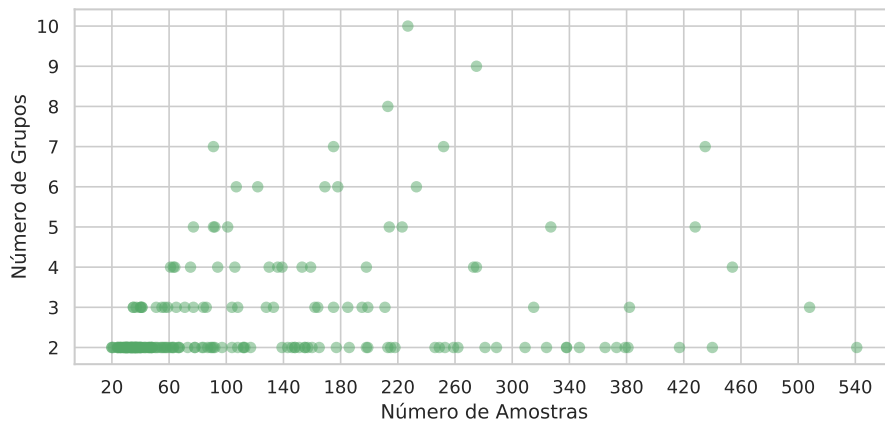


Figura 4.13: Relação do número de grupos pela quantidade de amostras considerando o particionamento com as *features* VGG-19.

De modo geral, os resultados nas Figuras 4.11, 4.12 e 4.13 indicaram que não houve correlação do número de grupos formados em relação à quantidade de amostras presentes nas classes.

4.2.6 Quantidade de subclasses geradas no particionamento

Após análise da distribuição da quantidade de classes particionadas por número de grupos, foi feito um levantamento do número de subclasses geradas no particiona-

mento para cada uma das *features* utilizadas. A Figura 4.14 apresenta este levantamento e nos permite discutir o impacto da escolha das *features* para o processo de particionamento em questão.

Observando as classes particionadas das amostras com *features* extraídas pela Inception V3, percebemos um número de 156 classes particionadas do total de 250 da coleção de imagens PlantCLEF 2013. As 156 classes particionadas na etapa de *particionamento* geraram 450 subclasses, que somadas às 94 classes não particionadas compõem a base PlantCLEF 2013 - Particionada com um total de 544 classes.

Para as amostras com *features* VGG-16, o número de classes particionadas foi maior que para as amostras com *features* Inception V3, com 193 classes. Estas classes particionadas originaram 507 subclasses, que somadas às 57 classes não particionadas, resultaram em outra base PlantCLEF 2013 - Particionada com um total de 564 classes.

Já para as amostras com *features* VGG-19 é possível observar um número maior de classes particionadas quando considerados os números de classes particionadas das com *features* Inception V3 e VGG16. Neste caso, foram particionadas 201 classes, das quais originaram-se 547 subclasses. Visto que para compor a nova base PlantCLEF 2013 - Particionada com amostras de *features* VGG-19, as 547 subclasses assumem os valores das suas classes originárias, que somadas as 49 classes não particionadas totalizam um total de 596.

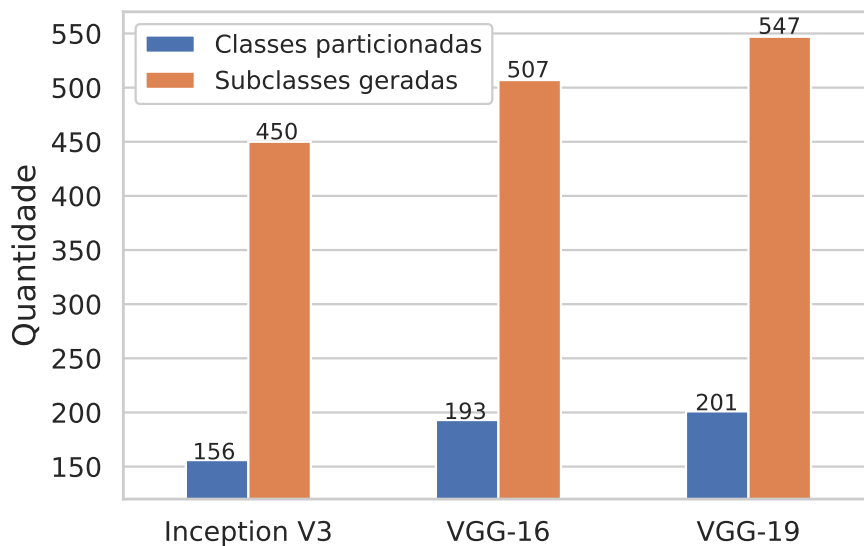


Figura 4.14: Levantamento do número de subclasses geradas no particionamento para cada uma das *features* utilizadas.

Após análise e discussão do levantamento do número de subclasses geradas no particionamento para cada uma das *features* mostrado na Figura 4.14, foi possível afirmar

que a escolha das *features* utilizadas na etapa de particionamento impactam diretamente na quantidade de classes sujeitas ao particionamento e conseqüentemente na quantidade de subclasses geradas. Em seguida analisaremos o impacto do particionamento considerando o tamanho inicial das classes e o problema de desbalanceamento na Seção 4.2.7.

4.2.7 Impacto do particionamento considerando o tamanho inicial das classes e desbalanceamento de amostras

Considerando o desbalanceamento das classes, ou seja, o desequilíbrio do número de amostras nas classes, verificamos as distribuições dos números de amostras nas subclasses descobertas.

A Figura 4.15 apresenta a distribuição do número de amostras para cada uma das 250 classes originais da coleção de imagens. Observa-se uma grande variação do número de amostras das classes, o que evidencia o problema de desbalanceamento das classes.

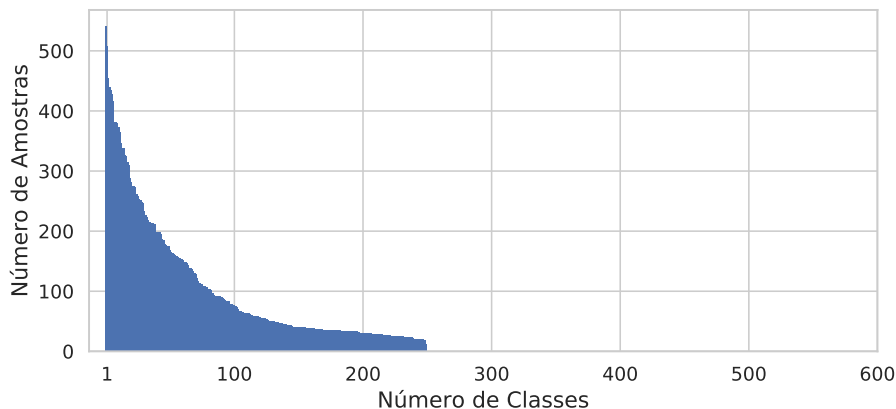


Figura 4.15: Distribuição do número de amostras para cada uma das 250 classes originais da coleção de imagens ImageCLEF 2013.

A Figura 4.16 apresenta a distribuição do número de amostras para cada uma das classes da base particionada com as *features* Inception V3. Observa-se que a distribuição do número de amostras em relação ao número de classes apresenta uma curva menos acentuada quando comparada a distribuição apresentada na Figura 4.15. Contudo, isso não permitiu solucionar o problema do desbalanceamento entre classes, mas a distribuição das amostras em mais classes, atenuando a diferença do número de amostras entre elas.

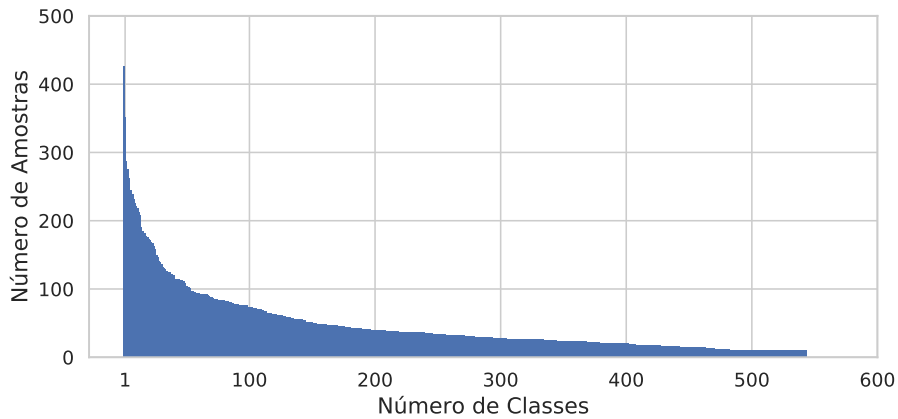


Figura 4.16: Distribuição do número de amostras nas classes da base particionada com amostras de *features* Inception V3.

A Figura 4.17 apresenta a distribuição do número de amostras para cada uma das classes da base particionada com as *features* VGG-16. Assim como na Figura 4.16, observa-se uma diferença menos acentuada na distribuição do número de amostras entre as classes.

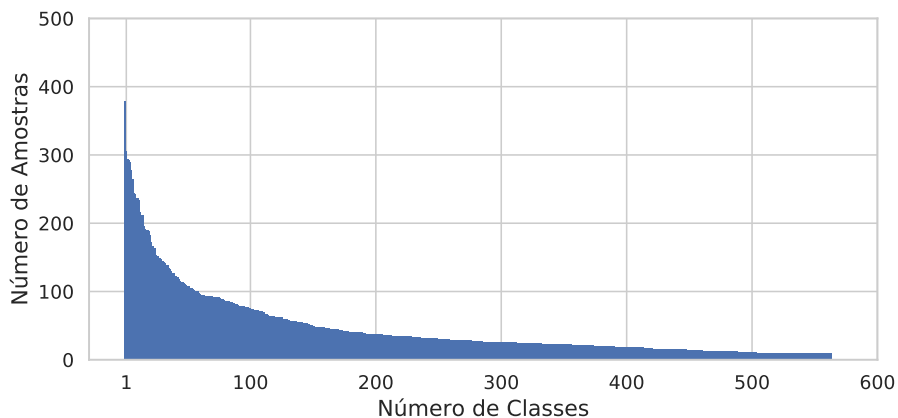


Figura 4.17: Distribuição do número de amostras para cada uma das classes da base particionada com amostras de *features* VGG-16

A Figura 4.18 apresenta a distribuição do número de amostras para cada uma das classes da base particionada com as *features* VGG-19. Percebe-se que, assim como nas Figuras 4.16 e 4.17, a maior parte das classes da base particionada apresentam poucas amostras, porém ainda há cerca de 50 classes cujo número de amostras é superior a 100. Assim, foi possível o problema de desbalanceamento foi apenas atenuado.

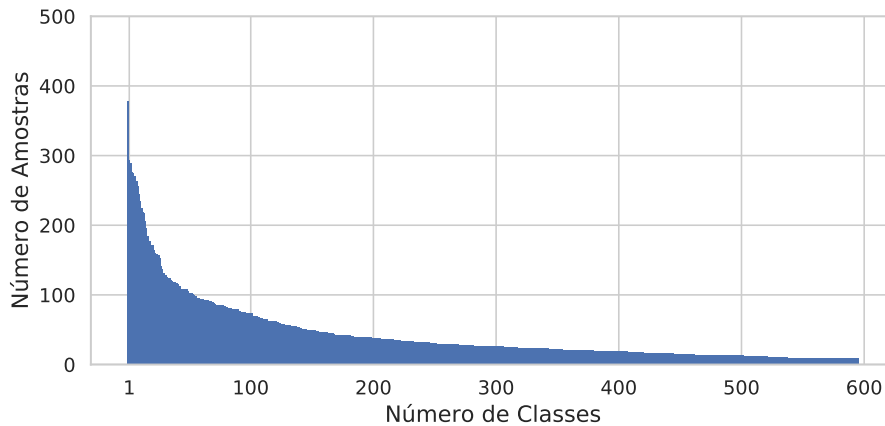


Figura 4.18: Distribuição do número de amostras para cada uma das classes da Base Particionada com amostras de *features* VGG-19.

Pelo exposto, observa-se que, mesmo que o problema do desequilíbrio entre as classes não tendo sido completamente sanado, as bases particionadas apresentaram uma melhor distribuição do número de amostras entre as classes.

4.2.8 Capacidade de diferenciação entre as subclasses descobertas

Além da análise qualitativa do particionamento pelo método de silhueta descrito na Seção 4.2.3 e com o objetivo de validar o método de particionamento e avaliar a capacidade do classificador de diferenciar entre as subclasses geradas, foram executados experimentos de classificação interna em cada uma das classes particionadas. A fim de verificar esta capacidade, foram realizados experimentos de classificação com duas variações do algoritmo SVM e um algoritmo *Random Forest* para cada uma das *features*. Os classificadores SVM utilizados para esta avaliação são de *kernel* linear com métodos de decomposição para problemas multiclasse diferentes, um *One vs. One* e outro *One vs. Rest*. Já a *Random Forest*, cujo o algoritmo de classificação é naturalmente multiclasse, foi executado com 100 árvores e demais configurações padrão de acordo com a documentação da biblioteca do *Scikit-learn* [Pedregosa et al. 2011]. O protocolo de validação utilizado foi o *10-fold Cross Validation* estratificado.

A Figura 4.19 apresenta os resultados da medida F1, computadas sobre *Precision* e *Recall* do tipo *micro*, relativas à classificação interna das subclasses para os três diferentes algoritmos utilizados considerando cada uma das *features*. Analisando a eficácia de classificação dos algoritmos quando utilizadas as *features* Inception V3, temos valores médios de F1 de 0,939 para o algoritmo *Random Forest*, 0,945 para SVM (linear, ovo) e 0,938 para o SVM (linear, ovr). Apesar de serem valores ligeiramente diferentes, a análise dos limites do intervalo de confiança de 95% nos permite inferir resultados estatisticamente equivalentes.

Ainda na Figura 4.19, podemos observar os resultados de F1 da classificação interna das subclasses com *features* VGG-16. Nesta análise, temos valores de 0,956, 0,954 e 0,948 para os classificadores *Random Forest*, SVM (linear, ovo) e SVM (linear, ovr) respectivamente. Neste caso, também observamos valores estatisticamente equivalentes após análise do intervalo de confiança de 95%.

Já os resultados de F1 da classificação interna das subclasses com *features* VGG-19, também apresentadas na Figura 4.19, nos permite observar resultados de 0,952 para os classificadores *Random Forest* e SVM (linear, ovo). E, ainda que a média de F1 para o SVM (linear, ovr) seja de 0,946, o intervalo de confiança de 95% para as médias dos três classificadores também indicam estabilidade desses resultados.

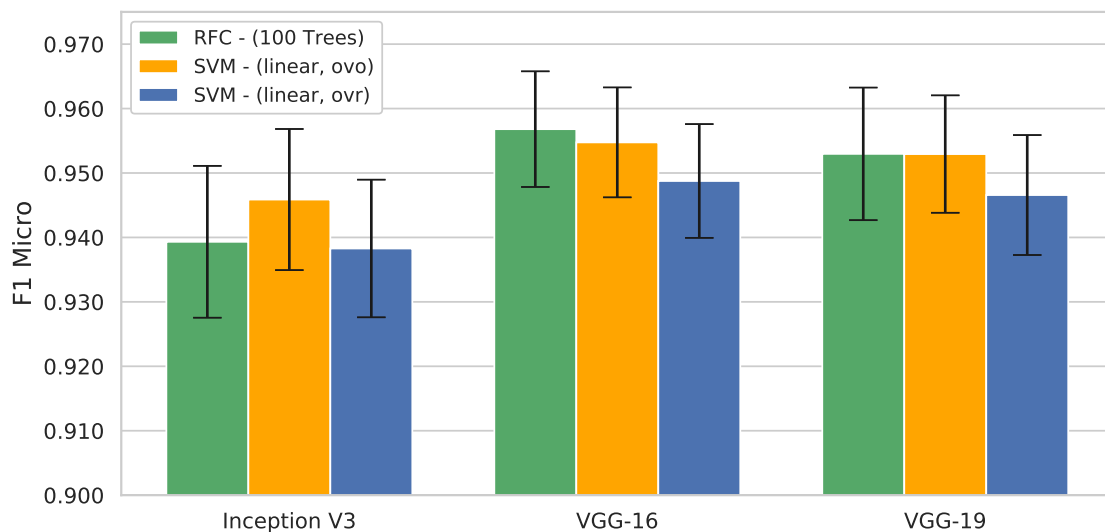


Figura 4.19: Resultados de F1 com intervalo de confiança de 95% da classificação interna das particionadas considerando as três *features* utilizadas.

Pelo exposto, considerando os resultados de todos os classificadores para as três diferentes *features*, podemos observar que todos os resultados são estatisticamente equivalentes. Além disso, assumindo que os classificadores tiveram resultados gerais acima de 93%, podemos concluir que é possível uma boa diferenciação entre as subclasses e que estas são homogêneas o suficiente para se alcançar bons resultados de classificação quanto consideradas no contexto geral do problema em estudo.

4.3 Avaliação da Classificação

Nesta seção são apresentados os resultados dos múltiplos classificadores em suas diversas configurações, a fim de verificar quais classificadores apresentam a maior eficácia de classificação no processo de reconhecimento automático de imagens de

plantas, tanto para coleção de imagens PlantCLEF 2013 (Base Original), quanto para PlantCLEF 2013 - Particionada (Base Particionada). Foram adotadas as métricas *Precision*, *Recall* e F1 (média harmônica entre *Precision* e *Recall*) do tipo *Micro*. Além disso, as médias apresentadas tanto para a Base Original quanto para a Base Particionada, tiveram seus resultados tabulados adequadamente, conforme descrito na Seção 3.7.

4.3.1 Resultados Gerais de Eficácia

Nesta seção é feita uma avaliação comparativa da eficácia de predição de todos os classificadores descritos na Seção 3.6 nas coleções de imagens PlantCLEF 2013 (Base Original) e PlantCLEF 2013 - Particionada (Base Particionada) considerando cada uma das *features*. Nesta seção não são apresentados valores médios de desvio padrão ou intervalo de confiança, já que busca-se identificar o classificador que apresenta maior eficácia em termos de F1 para cada uma das *features* e então executar o 10 *fold* - *Cross Validation* estratificado 10 vezes para análise de estabilidade e significância estatística.

No gráfico da Figura 4.20, são apresentados os resultados gerais de classificação com a medida F1 das Bases Original e Particionada, ambas com *features* extraídas pela Inception V3. De modo geral, o gráfico apresenta valores médios de F1 acima de 0,500 para maioria dos classificadores, com exceção do classificador SVM (polynomial, ovr) que apresentou resultados de 0,275 para a Base Original e 0,244 para a Base Particionada, ou seja, bastante inferiores em relação aos demais classificadores. Por outro lado, o classificador SVM (linear, ovo) apresentou as maiores médias de classificação tanto para a Base Original com 0,690, quanto para a Base Particionada com 0,716. Além disso, percebe-se que 7 dos 10 classificadores utilizados no experimento apresentaram melhores resultados de classificação para a Base Particionada em relação a Base Original.

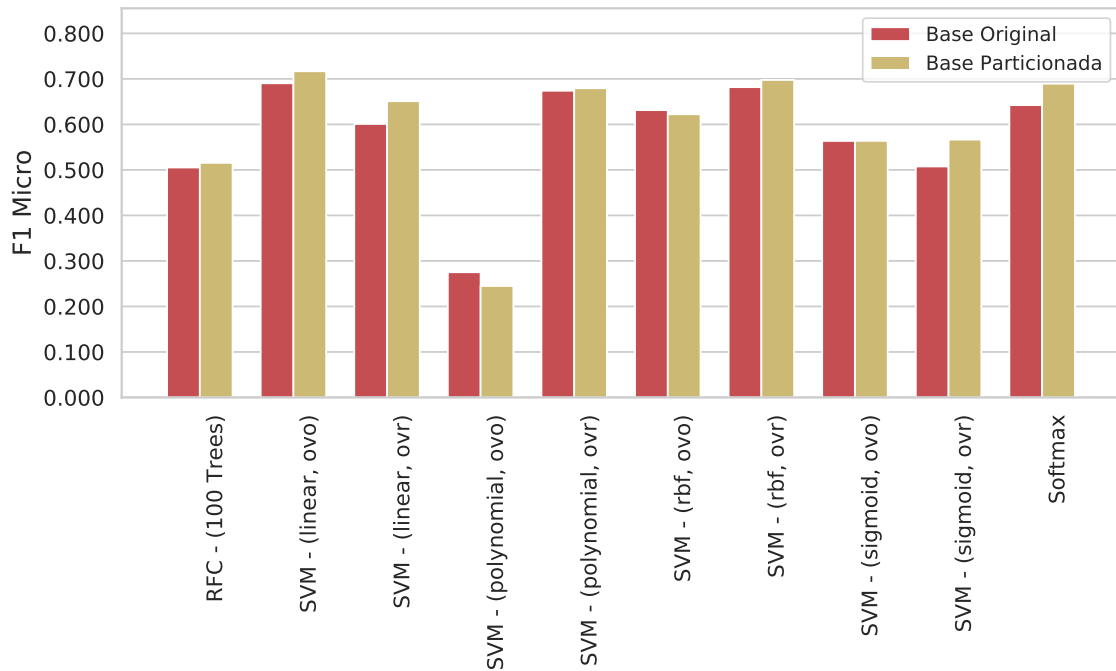


Figura 4.20: Resultados gerais de classificação (medida F1) para as Bases Original e Particionada com *features* extraídas pela Inception V3.

Na Figura 4.21 temos os resultados gerais de classificação das bases Original e Particionada com *features* extraídas pela VGG-16. Em geral, assim como nos resultados de classificação das bases com *features* da Inception V3, a maior parte dos classificadores obteve médias acima de 0,500 na classificação das Bases Original e Particionada. Utilizando-se das *features* VGG-16, o classificador que apresentou maior valor de F1 para as duas bases foi o SVM (polynomial, ovr), com 0,702 para Base Original e 0,700 para a Base Particionada. Porém o classificador SVM (sigmoid, ovr) apresentou médias extremamente baixas de 0,012 para a Base Original e 0,015 para a Base Particionada. Para o classificador SVM (sigmoid, ovo) os resultados também apresentaram-se relativamente baixos, onde a média para a Base Original foi de 0,103 e para a Base Particionada foi de 0,197, o que vale ressaltar é que mesmo se tratando de valores baixos houve uma melhora na eficácia do classificador quando aplicada a técnica de particionamento.

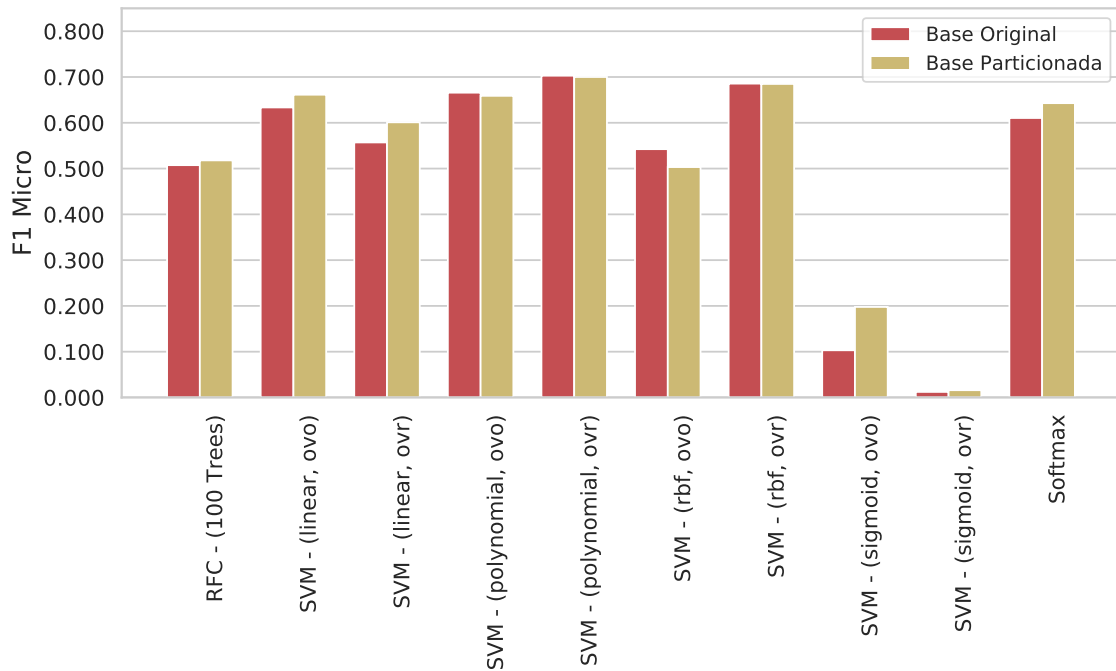


Figura 4.21: Resultados gerais de classificação com a medida F1 das Bases Original e Particionada com *features* extraídas pela VGG-16.

Por sua vez, a Figura 4.22 apresenta os resultados gerais de classificação das Bases Original e Particionada com *features* extraídas pela VGG-19. É possível observar neste gráfico, que assim como os resultados de classificação geral com *features* das Inception V3 e VGG-16, a maioria dos resultados possuem média acima de 0,500. E assim como ocorreu com os classificadores SVM (sigmoid, ovo) e SVM (sigmoid, ovr) utilizando *features* VGG-16, estes mesmos classificadores utilizando as *features* VGG-19 também apresentaram valores consideravelmente baixos em relação aos demais. Já os maiores valores de F1 para a classificação com as *features* VGG-19 foram obtidos pelo classificador SVM (polynomial, ovr) com 0,696 para a Base Original e 0,695 para a Base Particionada.

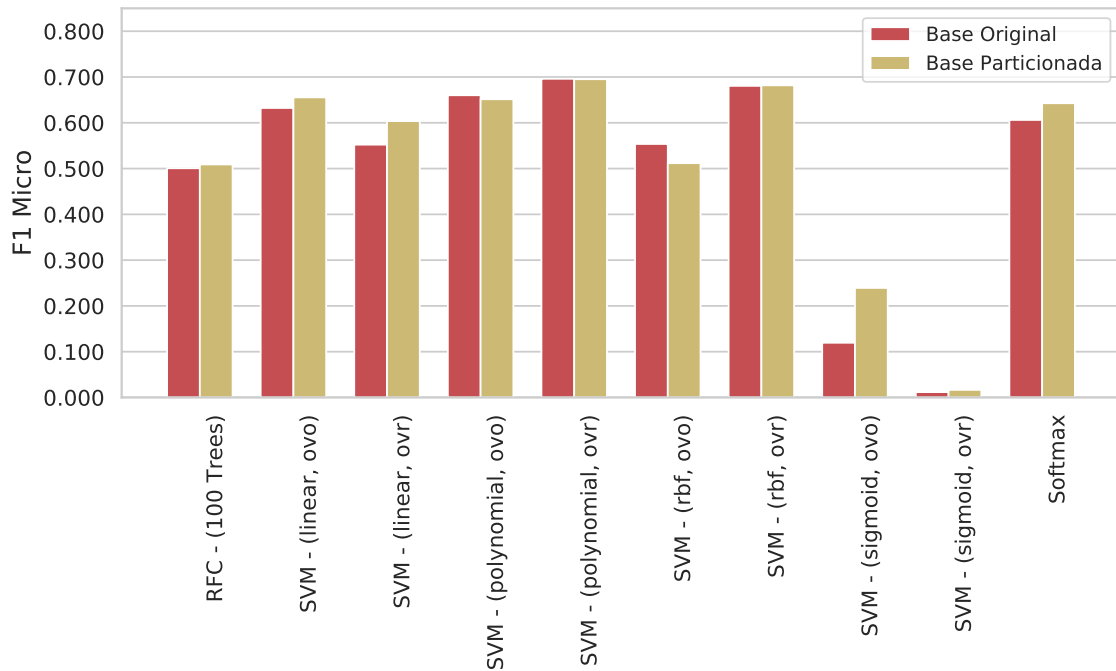


Figura 4.22: Resultados gerais de classificação com a medida F1 das Bases Original e Particionada com *features* extraídas pela VGG-19.

4.3.2 Estabilidade e significância estatística

Com o objetivo de verificar a eficácia de classificação quando utilizada a técnica de particionamento no reconhecimento automático de imagens de plantas, dentro do escopo deste trabalho e considerando a coleção de imagens objeto do estudo, foram feitas análises de estabilidade e significância estatística dos resultados para o classificador de maior média F1 para cada uma das *features* consideradas. As Figuras 4.23, 4.24 e 4.24 apresentam a análise de estabilidade e significância estatística dos melhores classificadores para cada uma das *features* com intervalo de confiança de 95%. A metodologia adotada para obtenção das médias e realização do teste estatístico é a descrita na Seção 3.7.1.

O gráfico da Figura 4.23 exibe os resultados comparativos da eficácia de classificação do classificador SVM (linear, ovo) para as Bases Original e Particionada com *features* Inception V3. A média (micro) de F1, calculada por meio de *Precision* e *Recall*, foi de 0,687 para a Base Original e de 0,712 para a Base Particionada. Considerando o intervalo de confiança de 95% e o resultado do teste *t de Student* que permitiu rejeitar a hipótese de que as médias são iguais, podemos concluir que o resultado da eficácia de classificação do classificador SVM (linear, ovo) foi superior quando o método de particionamento foi utilizado.

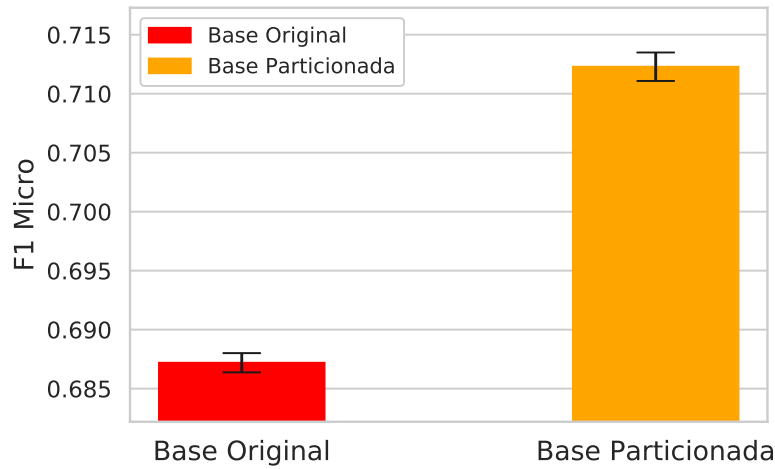


Figura 4.23: SVM - Linear - *One vs One* - Intervalo de Confiança 95% utilizando *Features* Inception V3.

No gráfico da Figura 4.24, podemos observar os resultados comparativos da eficácia de classificação do classificador SVM (polynomial, ovr), tanto para a Base Original quanto para a Base Particionada com as *features* extraídas pela VGG-16. Neste caso, o resultado de F1 (*Micro*) para a Base Original foi de 0,699, já para a Base Particionada foi de 0.695. Mesmo apresentando valores muito próximos, por meio do intervalo de confiança de 95%, é possível atestar que o classificador SVM (polynomial, ovr) obteve melhor resultado na classificação da Base Original em comparação a classificação da Base Particionada. Além disso, o teste *t de Student* rejeita a hipótese de que as médias são iguais.

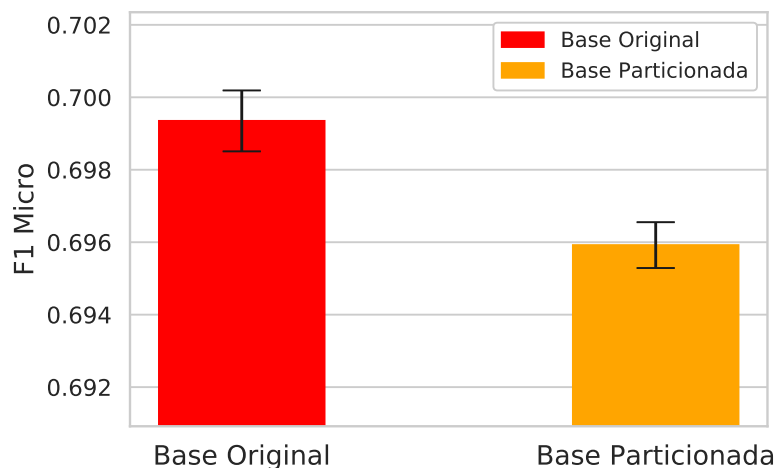


Figura 4.24: SVM - Polynomial - *One vs Rest* - Intervalo de Confiança 95% utilizando *Features* VGG-16.

Por sua vez, o gráfico da Figura 4.25 apresenta os resultados comparativos da eficácia de classificação do classificador SVM (polynomial, ovr) para as Bases Original e Particionada cujas *features* foram extraídas pela VGG-19. Verifica-se que o resultado da classificação da Base Original resultou em um valor de F1 de 0,693, enquanto que o resultado da classificação da Base Particionada foi de 0,691, ou seja, um pouco mais baixo. Apesar de ser uma diferença pequena, através do intervalo de confiança de 95%, podemos afirmar que os valores não são equivalentes. Além disso, o teste *t de Student* rejeita a hipótese de que as médias são iguais.

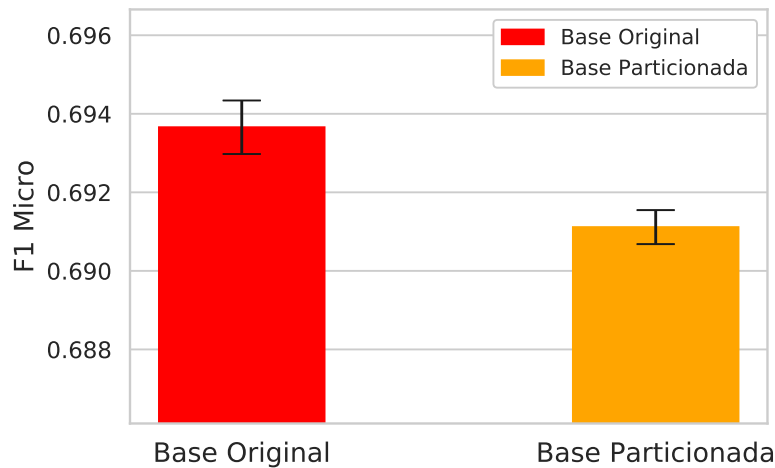


Figura 4.25: SVM - Polynomial - *One vs Rest* - Intervalo de Confiança 95% utilizando *Features* VGG-19.

Após a análise dos gráficos presentes nas Figuras 4.23, 4.24 e 4.24, conclui-se que a escolha das *features* impacta diretamente nos resultados de classificação tanto para a Base Original quanto para a Base Particionada. Considerando os resultados de classificação das bases com *features* da Inception V3, podemos concluir que a técnica de pré-agrupamento foi capaz aumentar a eficácia de classificação do classificador SVM (linear, ovo). Porém, quando utilizadas as *features* da VGG-16 e VGG-19 a técnica de pré-agrupamento não melhorou a eficácia do classificador SVM (polynomial, ovr).

4.4 Avaliação do Impacto da Seleção das *features* para a Classificação Baseada em Softmax

Esta seção apresenta uma análise comparativa do impacto da redução de dimensionalidade das *features* sobre a classificação usando o Softmax. Tendo em vista que o classificador Softmax integra a última camada *fully-connected* das Redes Neurais Convolucionais Profundas, e que seus resultados mostrados na Seção 4.3 foram

obtidos com *features* de dimensionalidades reduzidas pelo método de PCA, foram realizados experimentos de classificação com as *features* completas, o que não pôde ser realizado utilizando os classificadores clássicos devido ao seu custo computacional.

As Figuras 4.26, 4.27 e 4.28 apresentam os gráficos com os resultados comparativos da classificação da Base Original e da Base Particionada “Com PCA” e “Sem PCA” usando o classificador Softmax.

O gráfico da Figura 4.26 apresenta os resultados de F1 da classificação das bases Original e Particionada, ambas com as *features* da Inception V3 e usando o classificador Softmax. Percebe-se um ganho em relação às médias de classificação das bases Original e Particionada quando utilizadas as *features* sem o PCA. Os valores de F1 para a Base Original foram de 0,642 com PCA e de 0,699 sem PCA. Já os valores para a Base Particionada foram de 0,689 com PCA e de 0,721 sem PCA.

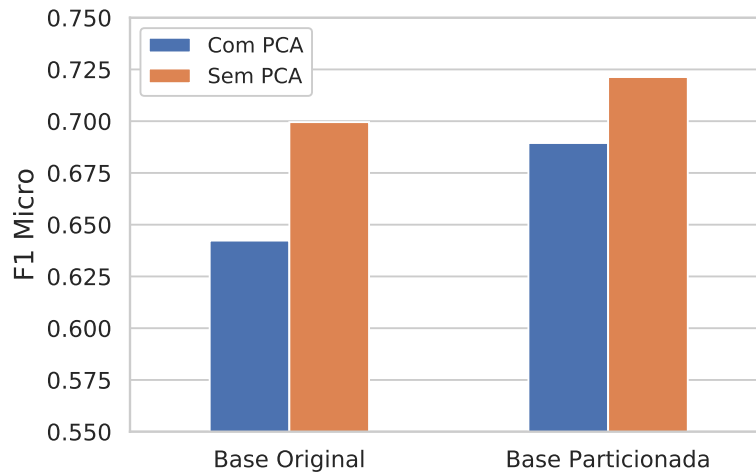


Figura 4.26: Resultados comparativos da classificação “Com PCA” e “Sem PCA” usando o Softmax - Inception V3.

O gráfico da Figura 4.27 apresenta os valores de F1 das bases Original e Particionada também usando o Softmax, porém com as *features* VGG-16. Também é possível observar valores mais altos de F1 para as bases Original e Particionada quando utilizadas as *features* sem redução de dimensionalidade pelo método de PCA. Neste caso, para a Base Original os valores de F1 são de 0,610 com PCA e de 0,651 sem PCA. Já os valores para a Base Particionada foram de 0,642 com PCA e de 0,664 sem PCA.

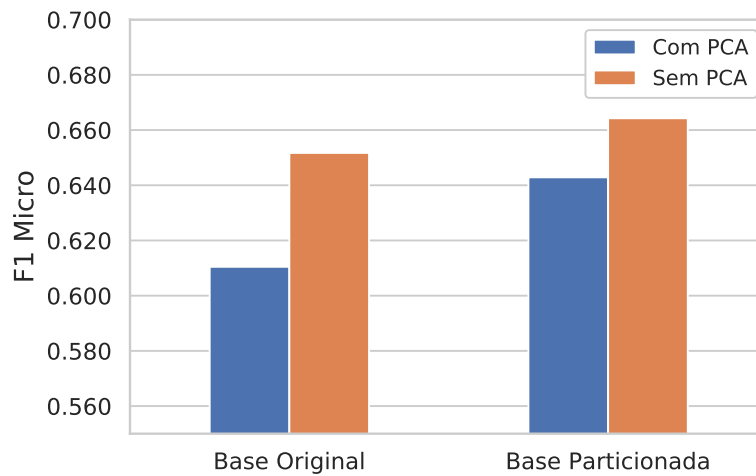


Figura 4.27: Resultados comparativos da classificação “Com PCA” e “Sem PCA” usando o Softmax - VGG-16.

Já o gráfico da Figura 4.28 apresenta os valores de F1 da classificação das Bases Original e Preclustering com *features* VGG-19 também usando o Softmax. Para estes experimentos, os valores de eficácia do classificador também foram mais altos quanto utilizadas as *features* sem redução de dimensionalidade. Especificamente os valores de F1 da classificação da Base Original foram de 0,606 com PCA e 0.651 sem PCA, já para a Base Particionada, os valores foram de 0,642 com PCA e 0,663 sem PCA.

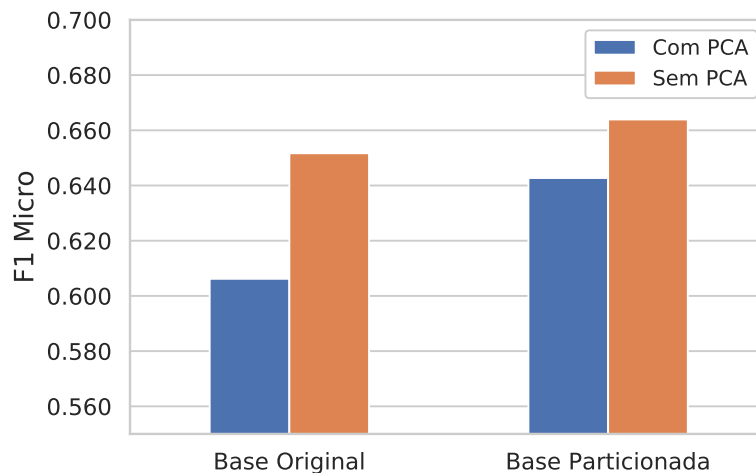


Figura 4.28: Resultados comparativos da classificação “Com PCA” e “Sem PCA” usando o Softmax - VGG-19.

Os gráficos da Figuras 4.26, 4.27 e 4.28 não apresentam análise de estabilidade ou teste de significância estatística, isto acontece porque o principal objetivo era

visualizar, de modo geral, se havia impacto na utilização das diferentes *features* com e sem PCA quando utilizado o Softmax. E a partir de então, de acordo com os resultados de maiores médias de F1, foi feita a análise de estabilidade e significância estatística com as *features* sem a redução de dimensionalidade usando o Softmax, o que é discutido na Seção 4.4.1.

4.4.1 Estabilidade e significância estatística do classificador Softmax sem redução de dimensionalidade

Após visualizados os resultados comparativos da eficácia do classificador Softmax sobre as *features* com e sem redução de dimensionalidade apresentados na Seção 4.4, optou-se pela execução do método 10 rodadas de 10-*fold Cross Validation* estratificado com as *features* sem redução de dimensionalidade.

O gráfico da Figura 4.29 apresenta os resultados de F1 do classificador Softmax com as *features* Inception V3 para a Base Original e Particionada. No que tange a análise comparativa de classificação das bases, percebe-se que o resultado da eficácia de classificação na Base Particionada foi melhor que na Base Original. Em termos quantitativos, a média de F1 para a Base Original foi de 0,698 enquanto que para a Base Particionada foi maior com valor de 0,722. Considerando-se a análise dos intervalos de confiança de 95% das médias em questão e o resultado do teste *t de Student* que rejeitou a hipótese de que as médias são equivalentes, podemos inferir que a técnica de particionamento de fato aprimorou a eficácia do classificador.

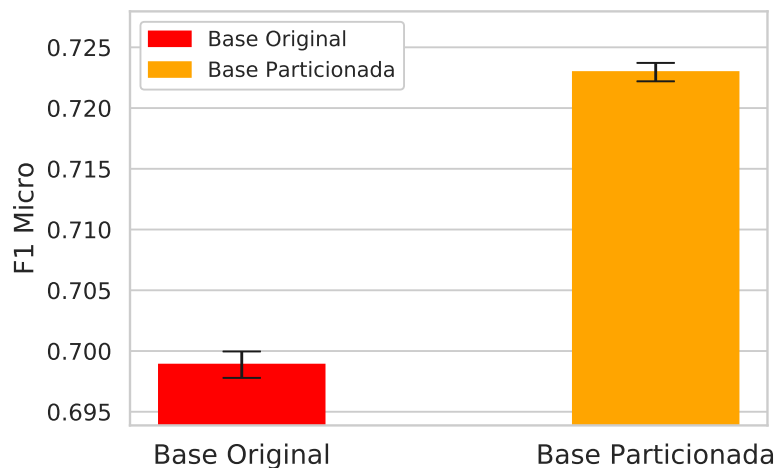


Figura 4.29: Eficácia do classificador Softmax com *features* sem redução de dimensionalidade - Intervalo de Confiança de 95% - Inception V3.

O gráfico da Figura 4.30 apresenta os resultados médios de F1 da eficácia do classificador Softmax com as *features* VGG-16 para a Base Original e Particionada. Neste

caso, o valor da média F1 para o resultado da classificação da Base Original foi de 0,660, enquanto que para a Base Particionada foi de 0,658, o que seria menor em comparação ao resultado da Base Original. Porém, quando analisados os intervalos de confiança de 95% de ambas as médias e o resultado do teste *t de Student* que aceitou a hipótese de que as médias são equivalentes, não se pode afirmar que de fato a eficácia do classificador foi melhor para a Base Original, mesmo esta apresentando valor da média F1 um pouco maior que a para a Base Particionada.

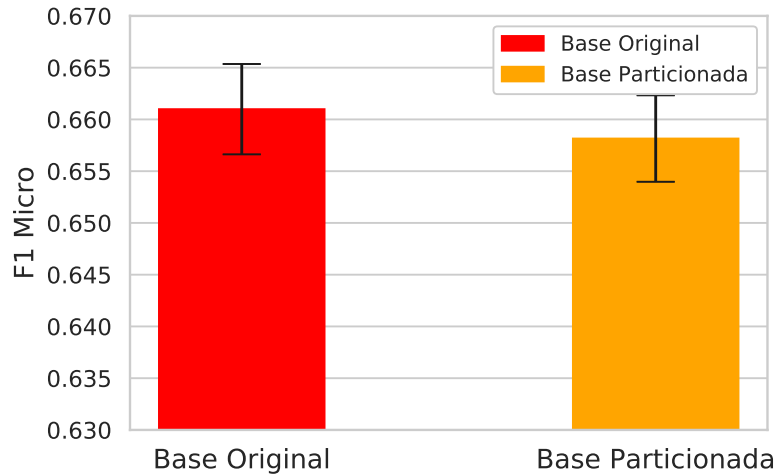


Figura 4.30: Eficácia do classificador Softmax com *features* sem redução de dimensionalidade - Intervalo de Confiança de 95% - VGG-16.

Já o gráfico da Figura 4.31 apresenta os resultados das médias de F1 do classificador Softmax com as *features* da VGG-19. Neste cenário, podemos observar que o valor médio de F1 na classificação da Base Original foi de 0,660, já o valor de F1 para a classificação da Base Particionada foi de 0,655. Mesmo apresentando valores distintos, ao analisar o intervalo de confiança de 95% não se pode afirmar que os valores são estatisticamente diferentes, o que se confirma com o resultado do teste *t de Student* que aceita a hipótese de que as médias são equivalentes.

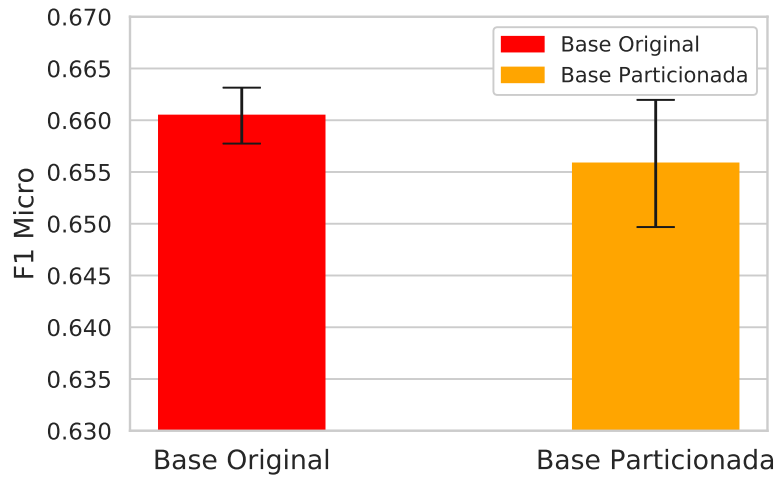


Figura 4.31: Eficácia do classificador Softmax com *features* sem redução de dimensionalidade - Intervalo de Confiança de 95% - VGG-19.

De modo geral, quando comparados os resultados de classificação das bases Original e Particionada, percebe-se que o particionamento efetivamente aumentou a eficácia de predição do classificador Softmax quando utilizadas as *features* Inception V3. Este ganho pode ser verificado pelo intervalo de confiança de 95% e pelo resultado do teste *t de Student* que rejeita a hipótese de que as médias são equivalentes. Porém, no que diz respeito aos resultados comparativos de ambas as bases, quando consideradas as *features* VGG-16 e VGG-19, a análise de estabilidade e significância estatística apontam para médias estatisticamente equivalentes.

4.5 Análise Visual dos Resultados

4.5.1 Ilustração Visual do processo de particionamento

Esta seção ilustra visualmente o processo de particionamento de duas classes da coleção de imagens PlantCLEF2013 utilizada neste trabalho. Os particionamentos mostrados visualmente nesta seção considerou as amostras com *features* Inception V3, além disso, a taxa de acerto mostrada nas Figuras 4.32 e 4.33 são baseadas na matriz de confusão resultante do processo de classificação utilizando o Softmax sem redução de dimensionalidade das *features*.

A Figura 4.32 apresenta uma ilustração visual do processo de particionamento da espécie (classe) *Prunus avium*, cujo número total de amostras é de 151 imagens, porém, para esta ilustração foram utilizadas somente 47 imagens selecionadas aleatoriamente. Em seu particionamento o método definiu 3 subclasses, sendo: *Prunus avium-C0* com 57 amostras, *Prunus avium-C1* com 70 amostras e *Prunus avium-C2* com 24 amostras. Observa-se que os grupos (subclasses) apresentam-se bastante

homogêneos, ou seja, as imagens de um grupo tem características semelhantes as do seu próprio grupo, porém suas características são diferentes das demais imagens dos outros grupos formados. Isto permitiu ao classificador melhorar sua taxa de acerto positivamente em 13%. Isso significa que para essa classe em específico, o método de particionamento minimizou o impacto da variabilidade intra-classe no processo de classificação.

A Figura 4.33 ilustra visualmente o processo de particionamento da espécie (classe) *Epilobium hirsotum*, cujo número total de amostras é de somente 35 imagens. A classe em questão foi particionada em 2 grupos (subclasses): *Epilobium hirsotum-C0* com 14 amostras e *Epilobium hirsotum-C1* com 21 amostras de imagens. Neste caso, foi possível mostrar todas as imagens da classe devido ao seu baixo número de amostras. Percebe-se pelo conjunto total de imagens da classe, que as imagens possuem pouca variabilidade entre si, apresentando em sua maioria imagens de apenas um órgão da espécie, sua flores. Isso fez com que seu particionamento formassem grupos com imagens visualmente bastante semelhantes, tanto entre imagens do próprio grupo quanto entre imagens do grupo seguinte. Observa-se pela taxa de acerto, que a similaridade entre as amostras das subclasses piorou a eficácia de predição do classificador em 20%. Isso significa que ao particionar esta classe com amostras homogêneas, acentuou-se o problema de similaridade-intraclasse.

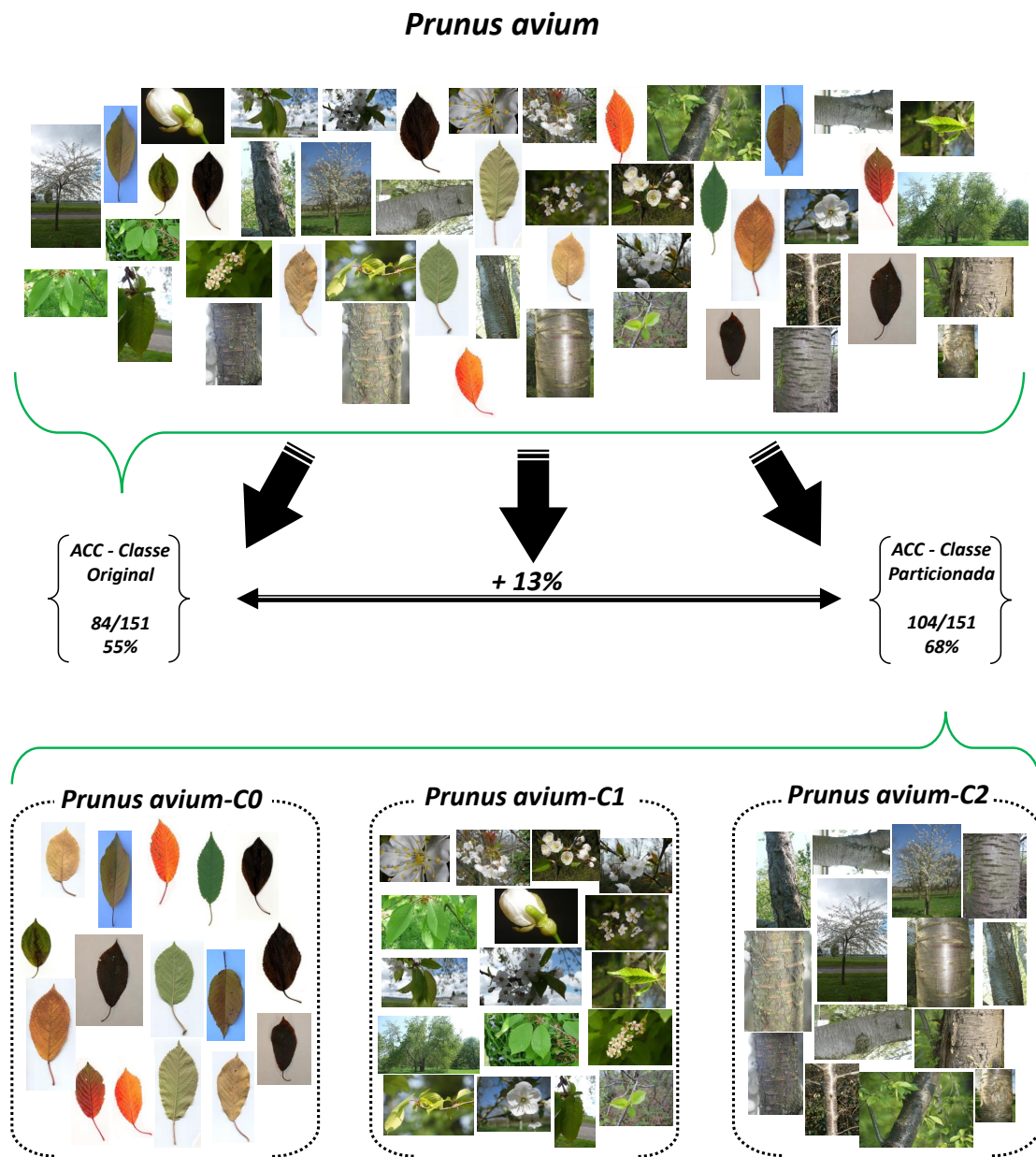


Figura 4.32: Análise visual do particionamento da espécie (classe) *Prunus avium* com *features* extraídas pela RNC Inception V3 e classificadas com Softmax.

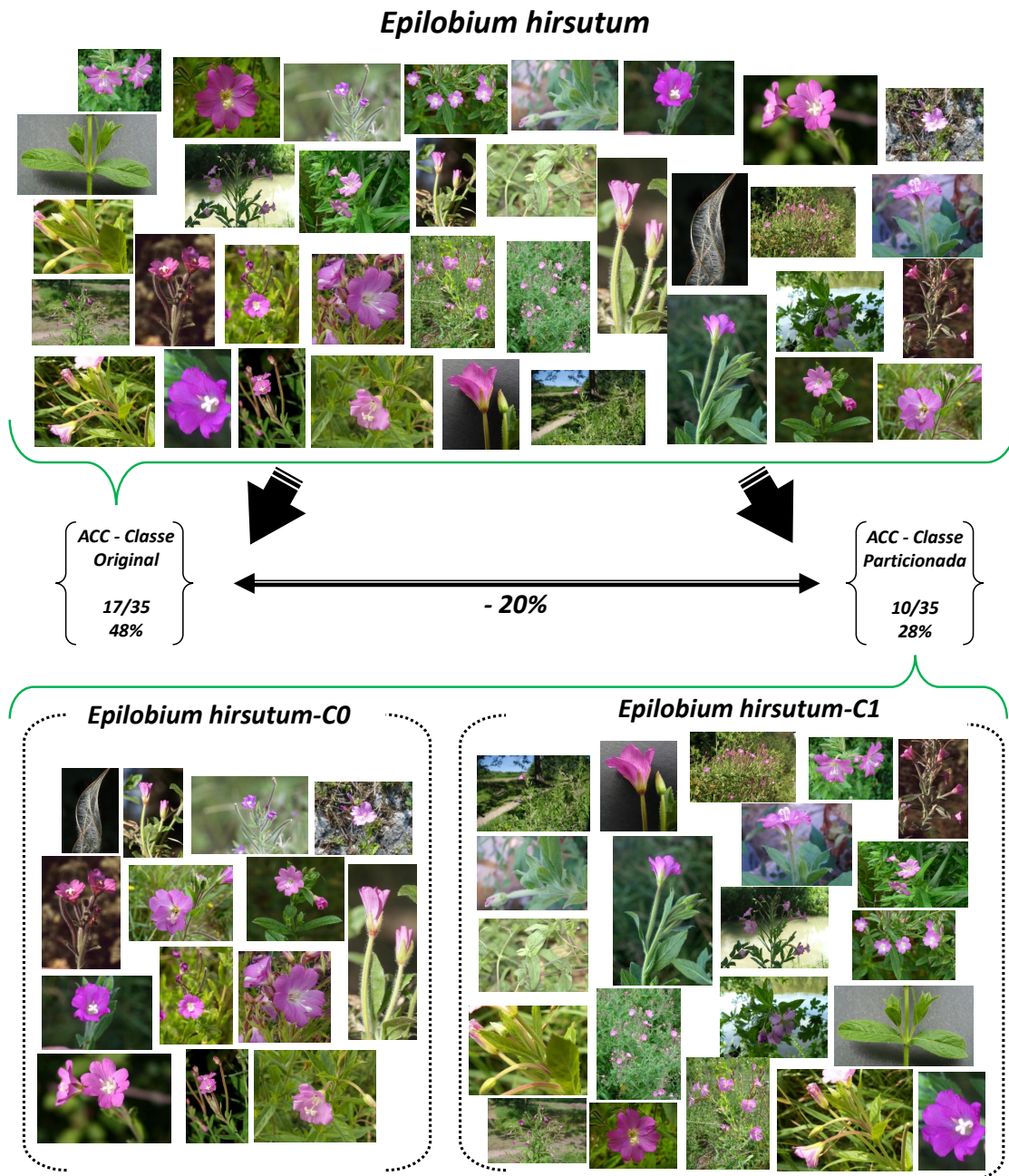


Figura 4.33: Análise visual do particionamento da espécie (classe) *Epilobium hirsutum* com *features* extraídas pela RNC Inception V3 e classificadas com Softmax.

4.5.2 Análise Visual das Matrizes de Confusão por Mapa de Calor

Esta seção trás uma análise visual das matrizes de confusão do resultado da eficácia de predição do classificador Softmax para as bases Original e Particionada com *features* Inception V3. As matrizes de confusão aqui apresentadas, são expostas em forma de mapa de calor. A escala de cores das matrizes das Figuras 4.34 e 4.35, variam entre amarelo e vermelho, sendo que quando mais amarelo mais próximo de zero é o valor na matriz, e conseqüentemente, quanto mais vermelho maiores são os valores presentes na matriz.

A Figura 4.34 apresenta os resultados de predição da Base Original. Visualmente, os pontos em vermelho indicam valores positivos que variam em uma escala de tons mais claros para valores mais baixos e mais escuro para valores mais altos. Na diagonal estão os valores verdadeiros positivos (TP), cuja tonalidade escura indica uma grande quantidade de acertos na maioria das classes. Outros pontos em vermelho distribuídos na matriz, indicam valores de confusão entre as classes na predição do classificador, porém, devido a alta proporção da matriz de 250x250 fica inviável uma identificação específica das classes com maior confusão.

A Figura 4.35 exhibe o mapa de calor dos resultados de predição da Base Particionada, porém, já redimensionada para equivalência com as proporções da matriz original de 250x250. Assim como na matriz anterior, a escala de cores é representada em amarelo e vermelho. Aqui também são observados os valores mais altos na diagonal da matriz, representados em vermelho escuro, que indicam os valores verdadeiros positivos (TP) para cada classe. Contudo, nota-se que os pontos circulosados apresentam tons de vermelho mais claros quando comparados com os mesmos pontos presentes na matriz de confusão da Base Original mostrada na Figura 4.34. Isto significa que houve uma diminuição dos valores de confusão entre essas classes quando aplicado o método de particionamento.

A partir das matrizes de confusão das bases Particionada e Original foi computada uma nova matriz de confusão com os valores da diferença entre ambas. A Figura 4.36 mostra valores representados em uma escala de três cores onde preto indica 0, vermelho os valores positivos e verde valores negativos. De modo geral, observa-se mais pontos em vermelho na diagonal da matriz cujos valores representam uma diferença positiva nos valores verdadeiros positivos (TP) para cada classe, o que significa um aumento da eficácia de predição do classificador na Base Particionada. É possível perceber ainda, onde o método corroborou no aumento da confusão entre as classes, porém há uma predominância de pontos na cor preta, o que mostra a neutralidade do método entre classes que não eram confundidas entre si.

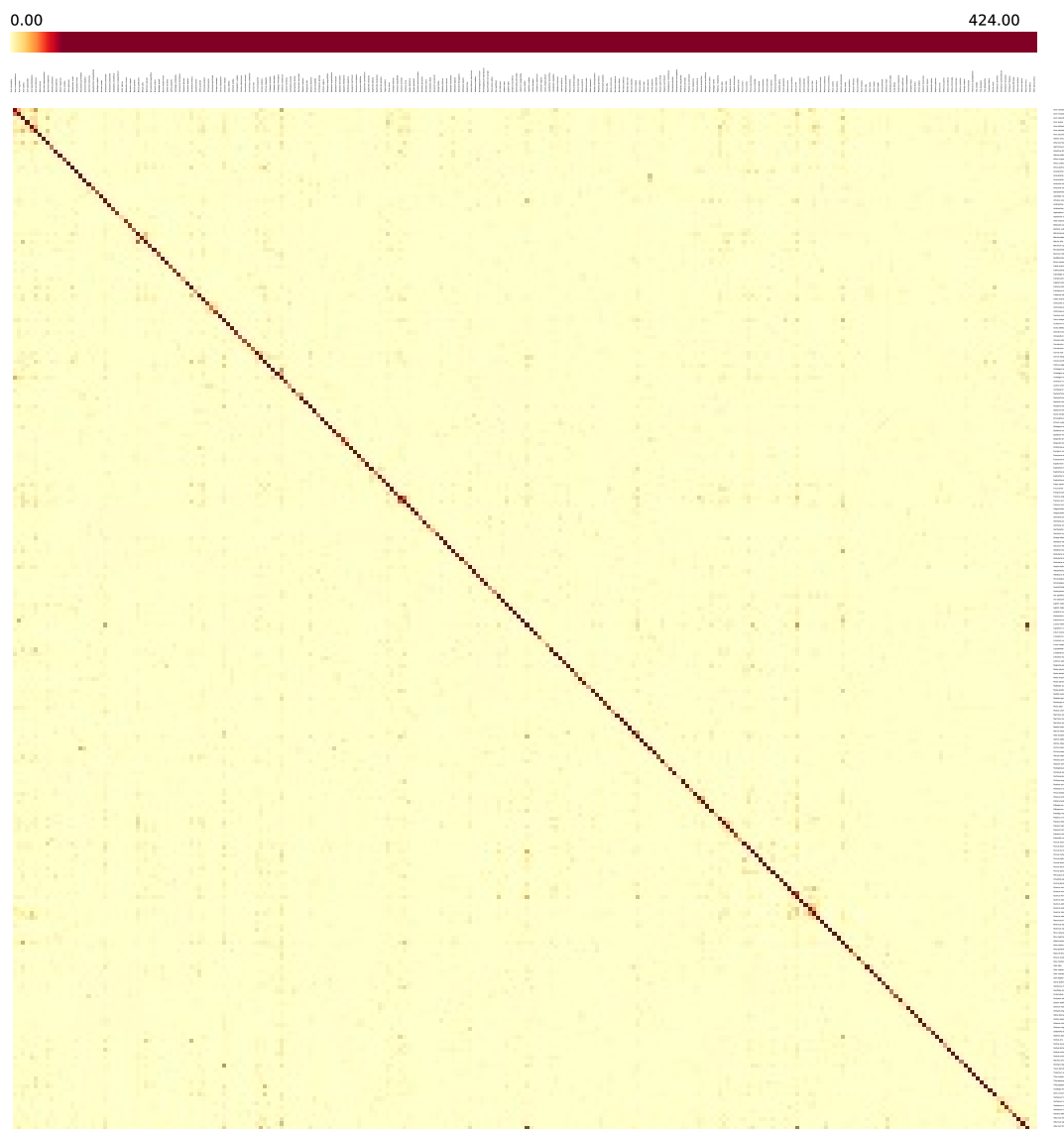


Figura 4.34: Mapa de calor da Matriz de Confusão da Base Original com *features* extraídas com a RNC Inception V3 e classificada com Softmax.

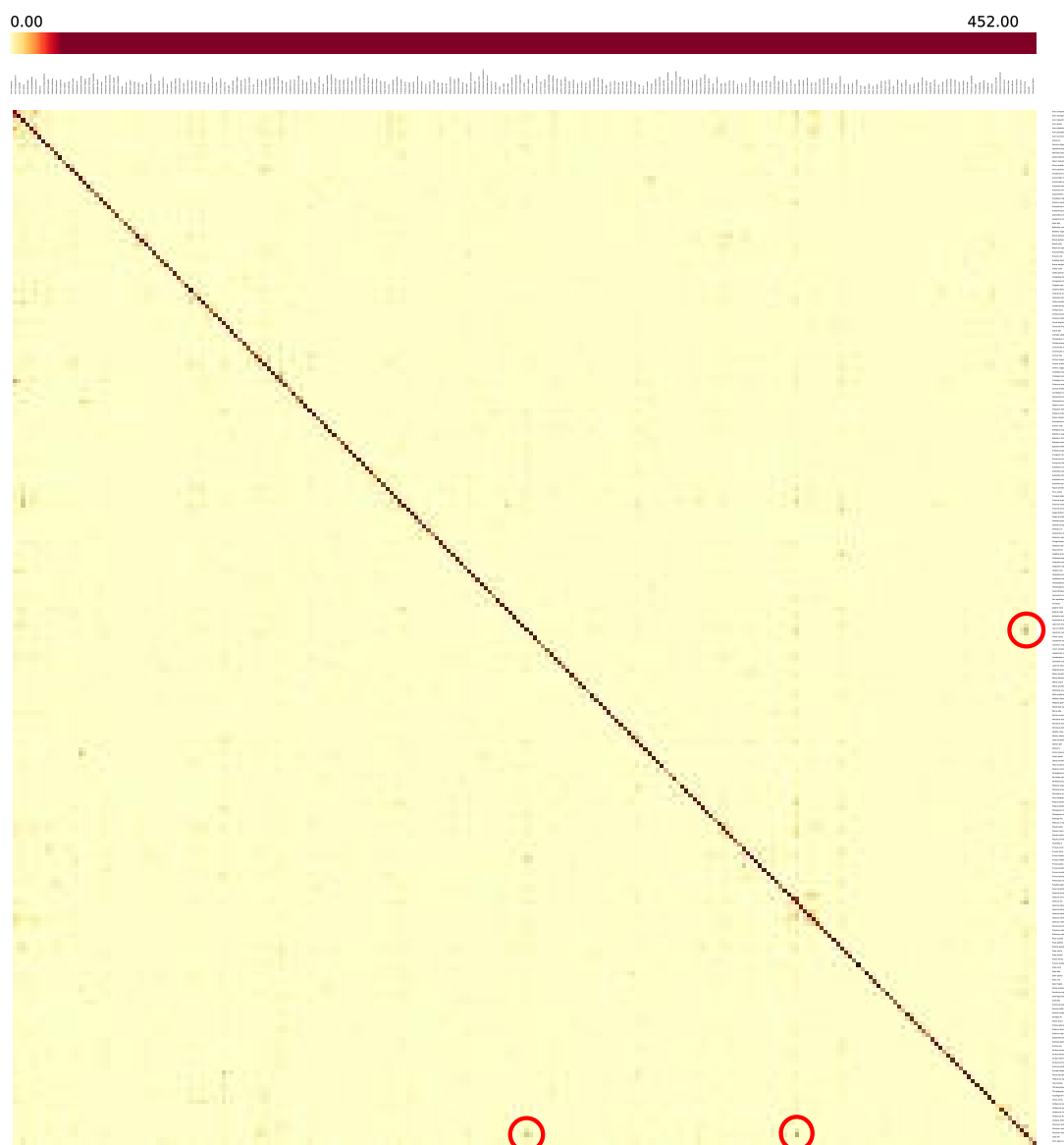


Figura 4.35: Mapa de calor da Matriz de Confusão da Base Particionada com *features* extraídas com a RNC Inception V3 e classificada com Softmax.

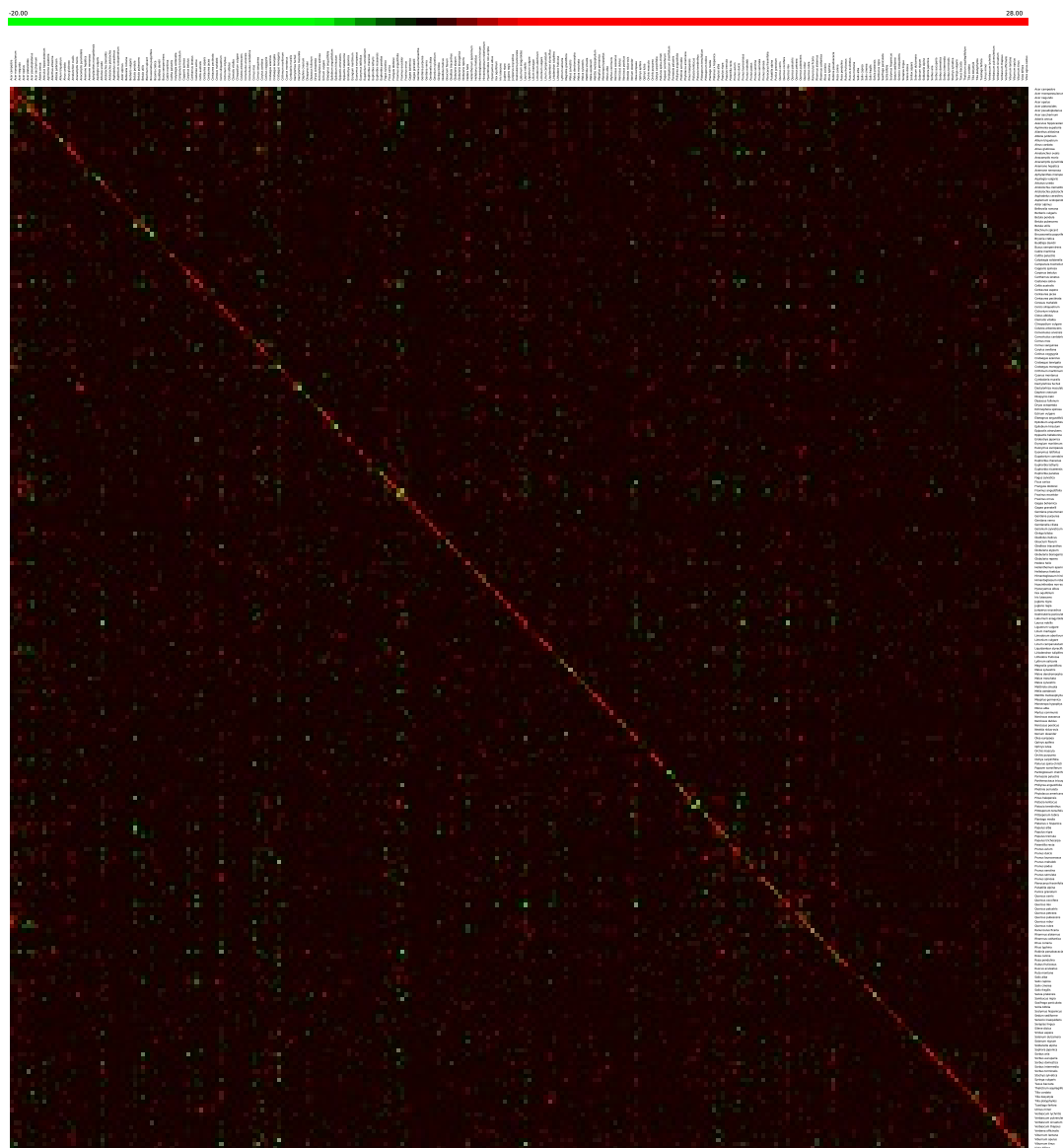


Figura 4.36: Mapa de calor da diferença entre as Matrizes de Confusão da Base Original e Particionada com *features* extraídas pela RNC Inception V3 e classificadas com Softmax.

Capítulo 5

Considerações Finais

A identificação automática de espécies de plantas vem se tornando uma solução muito promissora para auxiliar diversos e importantes campos de pesquisa, como ecologia, biodiversidade, farmacologia e áreas afins. A fim de contribuir para a solução de alguns dos principais desafios que envolvem o reconhecimento automático de plantas, este trabalho teve como principal objetivo explorar técnicas de aprendizado híbrido para minimizar os impactos da variabilidade intra-classe no processo de classificação de imagens de plantas.

Neste trabalho foi explorada a utilização de *features* extraídas por RNC. Como objeto de estudo, utilizou-se uma coleção de imagens composta 26.077 imagens de 250 espécies de plantas. Foram aplicadas técnicas de pré-agrupamento (particionamento) com base em diferentes *features*. Além disso, foram exploradas abordagens de classificação mais abrangentes, com uso métodos clássicos como o algoritmo *Random Forest* e oito variações do classificador SVM com *features* extraídas pelas RNC *Inception V3*, VGG-16 e VGG-19. Além disso, considerou-se também o classificador Softmax de cada uma das RNC a fim de se verificar o impacto do particionamento no processo de reconhecimento de imagens de espécies de plantas.

Sendo assim, considerando cada um dos objetivos específicos desta pesquisa, a partir dos experimentos realizados foram possíveis algumas conclusões:

- Considerando o objetivo de avaliar diferentes RNC's para extração de *features*, observou-se que a escolha das *features* extraídas pelas RNC's impactam diretamente nos resultados de classificação, tanto da Base Original quanto da Base Particionada. Isso fica evidente quando verificados os resultados por meio de teste de estabilidade e significância estatística.
- No que tange o objetivo de minimização do impacto dos problemas de variabilidade, observou-se que o uso de técnicas de aprendizado não-supervisionado, aqui chamada de particionamento, teve impacto significativo no reconhecimento automático de espécies de plantas da coleção de imagens objeto deste estudo. Percebemos que ao particionar as classes propensas à formação de

grupos naturais, aumentou-se o problema do classificador em relação ao número de saídas possíveis na predição, porém, isso fez o classificador aprender mais ao treinar com subclasses de amostras com características mais homogêneas ao invés de treinar com uma única classe com amostras de características heterogêneas.

- Em relação ao objetivo de avaliar a eficácia de múltiplos classificadores, percebemos que o uso de métodos clássicos de classificação tem seu custo computacional elevado quando utilizadas as *features* das RNC's completas, ou seja, sem redução de dimensionalidade. Levando-se em conta a utilização do classificador Softmax, podemos inferir que o ganho nos resultados da eficácia do classificador, quando aplicado o particionamento, se deu de fato quando utilizadas as *features* extraídas pela RNC Inception V3.

Espera-se que os resultados aqui apresentados possam trazer de fato uma contribuição para as pesquisas relacionadas ao desenvolvimento e aperfeiçoamento de sistemas de reconhecimento automático de espécies de plantas.

5.1 Trabalhos Futuros

Este estudo nos remete a outras possibilidades de experimentos futuros com a finalidade de aperfeiçoar a metodologia proposta e conseqüentemente aprimorar seus resultados, e assim, contribuir ainda mais com o desenvolvimento de novas técnicas de reconhecimento automático de espécies de plantas. Dentre as possibilidades de trabalhos futuros, podemos destacar:

- Utilizar a base de dados da competição PlantCLEF 2018 com cerca de 1 milhão de imagens de 10 mil espécies diferentes.
- Avaliar experimentalmente o método utilizando outras *features* de RNC's, como: *AlexNet* e *Inception V4*.
- Variabilizar os experimentos com diferentes tipos de algoritmos de *clustering* para realizar o particionamento das classes, por exemplo: i) *k-medoides* do tipo particional; ii) aglomerativo hierárquico (*bottom-up*) e iii) baseado em densidade (*DBSCAN*).
- Utilizar outros critérios de qualidade para avaliação do particionamento, como: i) Índice *Davis-Bouldin* e ii) Taxa Intra/Intercluster.
- Realizar uma análise mais profunda com otimizações paramétricas para os diferentes algoritmos de *clustering*.
- Fazer uma análise de otimização paramétrica para cada um diferentes classificadores utilizados.

Referências Bibliográficas

- [Anfeng He e Xinmei Tian 2016] Anfeng He e Xinmei Tian (2016). Multi-organ plant identification with multi-column deep convolutional neural networks. *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 2020–2025.
- [Banerjee e Davé 2004] Banerjee, A. e Davé, R. N. (2004). Validating clusters using the Hopkins statistic. *IEEE International Conference on Fuzzy Systems*, 1:149–153.
- [Banerjee e Dave 2004] Banerjee, A. e Dave, R. N. (2004). Validating clusters using the hopkins statistic. In *2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No.04CH37542)*, volume 1, pp. 149–153 vol.1.
- [Barré et al. 2017] Barré, P., Stöver, B. C., Müller, K. F., e Steinhage, V. (2017). LeafNet: A computer vision system for automatic plant species identification. *Ecological Informatics*.
- [Begue et al. 2017] Begue, A., Kowlessur, V., Singh, U., Mahomoodally, F., e Pudaruth, S. (2017). Automatic Recognition of Medicinal Plants using Machine Learning Techniques. *IJACSA) International Journal of Advanced Computer Science and Applications*, 8(4).
- [Biolab 2018] Biolab (2018). Orange3 Image Analytics Documentation Biolab.
- [Biot e Academy 1977] Biot, M. a. e Academy, R. (1977). Data Mining and Analysis: Fundamental concepts and Algorithms. 22:183–198.
- [Bradley et al. 2000] Bradley, P. S., Bennett, K. P., e Demiriz, A. (2000). Constrained k-means clustering. *Microsoft Research*, pp. 1–8.
- [Bramer 2007] Bramer, M. (2007). *Principles of Data Mining*.
- [Breiman 2001] Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5–32.
- [Carneiro et al. 2018] Carneiro, T., Medeiros Da Nóbrega, R. V., Nepomuceno, T., Bian, G., De Albuquerque, V. H. C., e Filho, P. P. R. (2018). Performance analysis of google colab as a tool for accelerating deep learning applications. *IEEE Access*, 6:61677–61685.

- [Chen et al. 2017] Chen, T., Xu, R., He, Y., e Wang, X. (2017). Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Systems with Applications*, 72:221–230.
- [Choi 2015] Choi, S. (2015). Plant identification with deep convolutional neural network: SNUMedinfo at LifeCLEF plant identification task 2015. *CEUR Workshop Proceedings*, 1391:2–5.
- [Chollet et al. 2015] Chollet, F. et al. (2015). Keras. <https://keras.io>.
- [Concetto et al. 2015] Concetto, S., Fisher, B., e Boom, B. (2015). LifeCLEF Fish Identification Task 2015. *CLEF working notes 2015*.
- [Corrigan 2007] Corrigan, M. (2007). *Pattern Recognition in Biology*. Nova Science Publishers.
- [Demšar et al. 2013] Demšar, J., Curk, T., Erjavec, A., Hočevar, T., Milutinovič, M., Možina, M., Polajnar, M., Toplak, M., Starič, A., Stajdohar, M., Umek, L., Zagar, L., Zbontar, J., Zitnik, M., e Zupan, B. (2013). Orange: Data Mining Toolbox in Python. *Journal of Machine Learning Research*, 14:23492353.
- [Dyrmann et al. 2016] Dyrmann, M., Karstoft, H., e Midtiby, H. S. (2016). Plant species classification using deep convolutional neural network. *Biosystems Engineering*.
- [Fabrizio Sebastiani 2002] Fabrizio Sebastiani (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1—47.
- [Fidalgo et al. 1989] Fidalgo, O., Bononi, V., e de Botânica, S. P. B. . S. I. (1989). *Técnicas de coleta, preservação e herborização de material botânico*. Série Documentos. Governo do Estado de São Paulo, Secretaria do Meio Ambiente, Instituto de Botânica.
- [Forman e Scholz 2010] Forman, G. e Scholz, M. (2010). Apples-to-apples in cross-validation studies. *ACM SIGKDD Explorations Newsletter*, 12(1):49.
- [Fürnkranz 2002] Fürnkranz, J. (2002). Round Robin Classification. *Journal of Machine Learning Research*, 2(4):721–747.
- [Ghazi et al. 2015] Ghazi, M. M., Yanikoglu, B., e Aptoula, E. (2015). Open-set Plant Identification Using an Ensemble of Deep Convolutional Neural Networks.
- [Gianessi e Reigner 2007] Gianessi, L. P. e Reigner, N. P. (2007). The Value of Herbicides in U.S. Crop Production. *Weed Technology*, 21(2):559–566.
- [Goëau et al. 2016] Goëau, H., Bonnet, P., e Joly, A. (2016). Plant identification in an open-world (LifeCLEF 2016). *CEUR Workshop Proceedings*, 1609(LifeCLEF):428–439.
- [Goeau et al. 2017] Goeau, H., Bonnet, P., e Joly, A. (2017). Plant identification based on noisy web data: the amazing performance of deep learning (LifeCLEF 2017). *CLEF working notes*, 2017(LifeCLEF).

- [Goëau et al. 2012] Goëau, H., Bonnet, P., Joly, A., Yahiaoui, I., Barthelemy, D., Boujemaa, N., e Molino, J. F. (2012). The ImageCLEF 2012 plant identification task. *CEUR Workshop Proceedings*, 1178.
- [Goëau et al. 2013] Goëau, H., Joly, A., Bonnet, P., Bakic, V., Barthélémy, D., Boujemaa, N., e Molino, J.-F. (2013). The imageCLEF Plant Identification Task 2013. *Proceedings of the 2Nd ACM International Workshop on Multimedia Analysis for Ecological Data*, (i):23–28.
- [Goodfellow et al. 2016] Goodfellow, I., Bengio, Y., e Courville, A. (2016). *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.
- [Gorunescu 2011] Gorunescu, F. (2011). *Data Mining: Concepts and Techniques*, volume 12.
- [Grinblat et al. 2016] Grinblat, G. L., Uzal, L. C., Larese, M. G., e Granitto, P. M. (2016). Deep learning for plant identification using vein morphological patterns. *Computers and Electronics in Agriculture*.
- [Hamuda et al. 2016] Hamuda, E., Glavin, M., e Jones, E. (2016). A survey of image processing techniques for plant extraction and segmentation in the field.
- [Hang et al. 2016] Hang, S. T., Tatsuma, A., e Aono, M. (2016). Bluefield (KDE TUT) at LifeCLEF 2016 plant identification task. *CEUR Workshop Proceedings*, 1609:459–468.
- [Hopkins e Skellam 1954] Hopkins, B. e Skellam, J. G. (1954). A new method for determining the type of distribution of plant individuals. *Annals of Botany*, 18(2):213–227.
- [Jia et al. 2014] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., e Darrell, T. (2014). Caffe: Convolutional Architecture for Fast Feature Embedding.
- [Joly et al. 2014] Joly, A., Goëau, H., Bonnet, P., Bakić, V., Barbe, J., Selmi, S., Yahiaoui, I., Carré, J., Mouysset, E., Molino, J. F., Boujemaa, N., e Barthélémy, D. (2014). Interactive plant identification based on social image data. *Ecological Informatics*.
- [Judd 2008] Judd, W. (2008). *Plant Systematics: A Phylogenetic Approach*. W. H. Freeman.
- [Kohavi e John 1997] Kohavi, R. e John, G. H. (1997). Wrappers for feature subset selection. *Artif. Intell.*, 97(1-2):273–324.
- [Krizhevsky et al. 2012] Krizhevsky, A., Sutskever, I., e Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances In Neural Information Processing Systems*, pp. 1–9.
- [Kumar et al. 2012] Kumar, N., Belhumeur, P. N., Biswas, A., Jacobs, D. W., Kress, W. J., Lopez, I. C., e Soares, J. V. (2012). Leafsnap: A computer vision system

- for automatic plant species identification. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*.
- [Lecun et al. 2015] Lecun, Y., Bengio, Y., e Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.
- [Lee et al. 2017] Lee, S. H., Chan, C. S., Mayo, S. J., e Remagnino, P. (2017). How deep learning extracts and learns leaf features for plant classification. *Pattern Recognition*.
- [Li et al. 2004] Li, Y., Wang, Z., e Zeng, H. (2004). Correlation filter: An accurate approach to detect and locate low contrast character strings in complex table environment. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(12):1639–1644.
- [Lin et al. 2013] Lin, M., Chen, Q., e Yan, S. (2013). Network In Network. pp. 1–10.
- [Louridas e Ebert 2016] Louridas, P. e Ebert, C. (2016). Machine Learning. *IEEE Software*, 33(5):110–115.
- [Manning et al. 2008] Manning, C. D., Raghavan, P., e Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.
- [Mehdipour Ghazi et al. 2017] Mehdipour Ghazi, M., Yanikoglu, B., e Aptoula, E. (2017). Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing*.
- [Ng 2017] Ng, S. C. (2017). Principal component analysis to reduce dimension on digital image. *Procedia Computer Science*, 111:113–119.
- [Pedregosa et al. 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., e Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- [PIMENTA 2005] PIMENTA, E. M. C. (2005). Abordagens para decomposição de problemas multi-classe: os Códigos de Correção de Erros de Saída. *Faculdade de Ciências do Porto*, Mestrado:103.
- [Psgr e Selvadoss Thanamani 2012] Psgr, A. C. e Selvadoss Thanamani, A. (2012). A Survey on Species Recognition System for Plant Classification. *INTERNATIONAL JOURNAL OF COMPUTER APPLICATIONS IN TECHNOLOGY*, 3:1132 – 1136.
- [Rifkin e Klautau 2004] Rifkin, R. e Klautau, A. (2004). In defense of one-vs-all classification. *Journal of machine learning research*, 5(Jan):101–141.
- [Ripley 2007] Ripley, B. (2007). *Pattern Recognition and Neural Networks*. Cambridge University Press.

- [Rousseeuw 1987] Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20(C):53–65.
- [Russakovsky et al. 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., e Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.
- [Schikora e Schikora 2014] Schikora, M. e Schikora, A. (2014). Image-based Analysis to Study Plant Infection with Human Pathogens.
- [Simonyan e Zisserman 2014] Simonyan, K. e Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. pp. 1–14.
- [Sladojevic et al. 2016] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., e Stefanovic, D. (2016). Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Computational Intelligence and Neuroscience*.
- [Sun et al. 2017] Sun, Y., Liu, Y., Wang, G., e Zhang, H. (2017). Deep Learning for Plant Identification in Natural Environment. *Computational Intelligence and Neuroscience*, 2017:1–6.
- [Szegedy et al. 2015] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., e Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June-2015:1–9.
- [Tipping e Bishop 1997] Tipping, M. E. e Bishop, C. M. (1997). Mixtures of Probabilistic Principal Component Analysers.
- [Torres e Falcão 2006] Torres, R. D. S. e Falcão, A. X. (2006). Content-Based Image Retrieval: Theory and Applications. *Revista de Informática Teórica e Aplicada RITA*, 13(2):161–185.
- [Witten et al. 2011] Witten, I. H., Frank, E., e Hall, M. a. (2011). *Data Mining: Practical Machine Learning Tools and Techniques (Google eBook)*.
- [Wu et al. 2007] Wu, S. G. G., Bao, F. S. S., Xu, E. Y. Y., Wang, Y.-X., Chang, Y.-f., e Xiang, Q.-l. (2007). A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network. *International Symposium on Signal Processing and Information Technology*, pp. 1–6.