UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

REGINA ROSA PARENTE

Classificação de Uma Classe para Seleção de Conjuntos de Dados Sintéticos em Meta-Aprendizado

Recife

2022

REGINA ROSA PARENTE

Classificação de Uma Classe para Seleção de Conjuntos de Dados Sintéticos em Meta-Aprendizado

Tese de Doutorado apresentada ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Doutor em Ciência da Computação.

Área de Concentração: Inteligência Computacional

Orientador: Ricardo Bastos Cavalcante Prudêncio

Recife

2022

Catalogação na fonte
Bibliotecária Nataly Soares Leite Moro, CRB4-1722

P228c Parente, Regina Rosa
Classificação de uma classe para seleção de conjuntos de dados sintéticos em meta-aprendizado / Regina Rosa Parente. – 2022.
110 f.: il., fig., tab.

Orientador: Ricardo Bastos Cavalcante Prudêncio.
Tese (Doutorado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2022.
Inclui referências e apêndice.

1. Inteligência computacional. 2. Meta-aprendizado. 3. Seleção de algoritmo. 4. Seleção de conjuntos de dados sintéticos. 5. Classificação de uma classe. I. Prudêncio, Ricardo Bastos Cavalcante (orientador). II. Título

006.31 CDD (23. ed.) UFPE - CCEN 2022 – 77

Regina Rosa Parente

“Classificação de Uma Classe para Seleção de Conjuntos de Dados Sintéticos em Meta-Aprendizado”

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação. Área de Concentração: Inteligência Computacional

Aprovado em: 04/02/2022.

_____

Orientador: Prof. Dr. Ricardo Bastos Cavalcante Prudêncio

BANCA EXAMINADORA

_____

Prof. Dr. Tsang Ing Ren
Centro de Informática / UFPE

_____

Prof. Dr. Paulo Salgado Gomes De Mattos Neto
Centro de Informática / UFPE

_____

Profa Dra. Gisele Lobo Pappa
Departamento de Ciência da Computação / UFMG

_____

Prof. Dr. Péricles Barbosa Cunha de Miranda
Departamento de Computação / UFRPE

_____

Prof. Dr. Ronnie Cley de Oliveira Alves
Instituto de Ciências Exatas e Naturais / UFPA

Ao meu pai Francisco de Assis da Silva Parente (em memória), por ter sido a pedra fundamental do caminho trilhado para que hoje eu estivesse aqui. À minha irmã Conceição Rosa Parente (em memória) por todo amor e carinho a mim dedicados e todo apoio ao longo da minha trajetória.

AGRADECIMENTOS

Agradeço, primeiramente, a Deus pelo dom da vida, por tudo que tenho e sou e por ter sonhado comigo mais esse passo na minha trajetória e providenciado todos os meios necessários para que ele fosse possível. Obrigada pelas oportunidades a mim concedidas e, de modo muito especial, pelas pessoas que tem posto em meu caminho.

Sou muito grata a minha família, minha mãe por entender a ausência e me apoiar quando foi necessário. Agradeço aos sobrinhos que sempre me apoiaram e incentivaram nessa jornada.

Agradeço ao professor Ricardo por ter aceitado ser meu orientador e não ter desistido de mim, mesmo quando tudo parecia desfavorável. Sou imensamente grata por ter usufruído dos seus vastos conhecimentos que foram fundamentais para que esse projeto fosse concluído.

Agradeço a todos os meus amigos, de forma muito especial a Débora, Kamilla, Sandra Beatriz e Diogo Vinícius por toda paciência nos momentos desafiadores e também por se alegrarem comigo nas celebrações das vitórias. Obrigada Regina Machado, minha mãe de coração. Tudo começou com você há 29 anos, regando a semente do saber que existia e ainda existe dentro de mim. Obrigada! Você é responsável pela minha trajetória até aqui. Obrigada Mario Gomes, por me ouvir muitas vezes e compartilhar comigo seus conhecimentos.

Esses anos de doutorado foram muito desafiadores. Logo no início do curso, sofri uma perda que ainda luto para administrar. Minha irmã, minha amiga, às vezes minha mãe, Conceição, "Neca"para os íntimos, foi levada de junto de nós de uma forma dura, sofrida, ao ponto de eu achar que nada mais fazia muito sentido, nem mesmo essa jornada que agora encerra-se chamada "doutorado". Já seria uma jornada longa, mas começando assim, foi sentida muito mais. Nesse momento ímpar, recebi muito carinho, amizade, atenção. Presença. Obrigada Débora Rodrigues, Rosana Parente, Maria Júlia, Daniel, Daiane, Liz, Sandrinha, Camila Cristina, Tia Dulce, Ricardo, Kamilla Uhl.

Agradeço também a todos os colegas que tive o prazer de conhecer nos laboratórios da UFPE. Silas, Mario, Bruno Wagner, Vanessa, Halisson. O convívio com vocês foi muito enriquecedor e não poderia esquecê-los.

Obrigada a todos os professores e funcionários da UFPE por estarem sempre disponíveis para contribuir na construção de tudo que aprendi. Agradeço a todos os colegas do IFMA campus Bacabal pelo apoio de sempre.

RESUMO

A seleção de algoritmos é uma tarefa crucial e desafiadora no aprendizado de máquina em diferentes domínios do conhecimento. O meta-aprendizado trata a seleção de algoritmos como uma tarefa de aprendizado supervisionado. Exemplos de treinamento (ou seja, meta-exemplos) são gerados a partir de experimentos realizados com um conjunto de algoritmos candidatos em vários conjuntos de dados. Pode haver uma pequena disponibilidade de conjuntos de dados reais em alguns domínios, o que torna difícil gerar bons meta-exemplos. Portanto, confiar em conjuntos de dados sintéticos pode ser uma boa alternativa para gerar meta-exemplos. No entanto, não é garantido que todos os conjuntos de dados sintéticos sejam relevantes e representativos em comparação com os conjuntos de dados reais. Desta forma, o uso indiscriminado de muitos conjuntos de dados sintéticos aumenta o custo computacional da realização de experimentos sem melhorar significativamente a acurácia do meta-aprendizado. Nesta tese, lidamos com a seleção de conjuntos de dados sintéticos para meta-aprendizagem como um problema de classificação de uma classe (OCC). Em OCC, os classificadores são construídos assumindo a disponibilidade de exemplos de treinamento pertencentes a uma classe única de interesse (ou seja, a classe positiva), enquanto os rótulos de classe dos outros exemplos são desconhecidos. Na solução proposta, técnicas de OCC são utilizadas para selecionar os conjuntos de dados sintéticos mais relevantes (a classe desconhecida), considerando os conjuntos de dados reais (a classe positiva) disponíveis. Também conduzimos experimentos em dois estudos de caso, nos quais empregamos dois procedimentos diferentes de manipulação de dados para produzir conjuntos de dados sintéticos e duas técnicas de OCC para seleção de conjuntos de dados. Os resultados demonstraram que é possível usar um número reduzido de conjuntos de dados sintéticos selecionados, mantendo um bom desempenho do Meta-Aprendizado.

Palavras-chave: meta-aprendizado; seleção de algoritmo; seleção de conjuntos de dados sintéticos; classificação de uma classe.

ABSTRACT

Algorithm selection is a crucial and challenging task in machine learning in different domains of knowledge. Meta-learning treats algorithm selection as a supervised learning task. Training examples (i.e. meta-examples) are generated from experiments performed with a set of candidate algorithms in several datasets. There may be a small availability of real datasets in some domains, which makes it challenging to generate good meta-examples. Therefore, relying on synthetic datasets may be a good alternative to generate meta-examples. However, it is not guaranteed that all synthetic datasets are relevant and representative compared to real datasets. In this way, the indiscriminate use of too many synthetic datasets increases the computational cost of performing experiments without significantly improving meta-learning accuracy. In this thesis, we deal with the selection of synthetic datasets for meta-learning as a one-class classification (OCC) problem. In OCC, classifiers are built by assuming the availability of training examples belonging to a unique class of interest (i.e., the positive class), while the class labels of the other examples are unknown. In the proposed solution, OCC techniques are used to select the most relevant synthetic datasets (the unknown class), by considering the real datasets (the positive class) available. We also conducted experiments in two case studies, in which we employed two different data manipulation procedures to produce synthetic datasets and two OCC techniques for dataset selection. The results demonstrated that it is possible to use a reduced number of selected synthetic datasets while maintaining a good meta-learning performance.

Keywords: meta-learning; algorithm selection; selection of synthetic datasets; one-class classification.

LISTA DE FIGURAS

Figura 1 – Processo de meta-aprendizagem para seleção de algoritmos 26
Figura 2 – Categorização para o estudo de Técnicas de *One-Class Classification*(OCC) 42
Figura 3 – Uma descrição esquemática da solução proposta. 52
Figura 4 – Ilustração de meta-exemplo sintético que é aceito por *Nearest Neighbor Data Description* (NN-d). As instâncias sintética e real são muito próximas uma da outra. Os triângulos representam os meta-exemplos reais e os quadrados os meta-exemplos desconhecidos. 56
Figura 5 – Ilustração de meta-exemplo sintético que é rejeitado por NN-d. As instâncias sintética e real são muito distantes uma da outra. Os triângulos representam os meta-exemplos reais e os quadrados os meta-exemplos desconhecidos. . 56
Figura 6 – Geração de dois conjuntos de dados sintéticos de classificação a partir de uma base de dados com dois atributos simbólicos. 58
Figura 7 – Conjunto de dados Iris Original 62
Figura 8 – Conjunto de dados Iris manipulado 62
Figura 9 – Pipeline de Geração de Conjuntos de Dados Sintéticos. 64
Figura 10 – Medidas de complexidade melhor classificadas selecionadas pelo meta-classificador *Random Forest* (RF) 65
Figura 11 – 1114 conjuntos de dados sintéticos mapeados nas vinte e duas medidas de complexidade 68
Figura 12 – 78 conjuntos de dados originais mapeados nas vinte e duas medidas de complexidade 68
Figura 13 – A relação entre o número de instâncias negativas selecionadas (N_r) na primeira etapa consideradas verdadeiramente negativas (círculos) e o número de instâncias selecionadas (U_p) na segunda etapa de aprendizagem de PU consideradas relevantes, ou seja, pertencentes a classe positiva (quadrados). 73

Figura 14 – Acurácia média obtida por RF usando *Positive and Unlabeled Learning* (PU Learning) para selecionar meta-exemplos sintéticos. O conjunto de treinamento é inicializado com todos os meta-exemplos originais, deixando apenas meta-exemplos sintéticos para a seleção de PU Learning. As linhas horizontais tracejadas representam a acurácia obtida usando todos os meta-exemplos originais e meta-exemplos sintéticos (parte superior) ou apenas os meta-exemplos originais (parte inferior). 74
Figura 15 – Acurácia média obtida por *k-nearest Neighbors* (KNN) usando PU Learning para selecionar meta-exemplos sintéticos. O conjunto de treinamento é inicializado com todos os meta-exemplos originais, deixando apenas meta-exemplos sintéticos para a seleção de PU Learning. As linhas horizontais tracejadas representam a acurácia média observada usando todos os meta-exemplos originais e meta-exemplos sintéticos (parte superior) ou apenas os meta-exemplos originais (parte inferior). 75
Figura 16 – Acurácia média obtida por RF usando NN-d para selecionar meta-exemplos sintéticos. O conjunto de treinamento é inicializado com todos os meta-exemplos originais, deixando apenas meta-exemplos sintéticos para seleção. As linhas horizontais tracejadas representam a acurácia obtida usando todos os meta-exemplos originais e meta-exemplos sintéticos (parte superior) ou apenas os meta-exemplos originais (parte inferior). 77
Figura 17 – Acurácia média obtida por KNN usando NN-d para selecionar meta-exemplos sintéticos. O conjunto de treinamento é inicializado com todos os meta-exemplos originais, deixando apenas meta-exemplos sintéticos para seleção. As linhas horizontais tracejadas representam a acurácia obtida usando todos os meta-exemplos originais e meta-exemplos sintéticos (parte superior) ou apenas os meta-exemplos originais (parte inferior). 77
Figura 18 – Acurácia média obtida por RF usando PU Learning para selecionar conjuntos de dados sintéticos. O conjunto de treinamento é inicializado com todos os conjuntos de dados originais, deixando apenas conjuntos de dados sintéticos para seleção de PU. As linhas horizontais tracejadas representam a acurácia obtida usando todos os conjuntos de dados originais e sintéticos (parte superior) ou apenas os conjuntos de dados originais (parte inferior). 81

Figura 19 – Acurácia média obtida por KNN usando PU Learning para seleção de conjuntos de dados sintéticos. O conjunto de treinamento é inicializado com todos os conjuntos de dados originais, deixando apenas conjuntos de dados sintéticos para seleção de PU. As linhas horizontais tracejadas representam a acurácia obtida usando todos os conjuntos de dados originais e sintéticos (parte superior) ou apenas os conjuntos de dados originais (parte inferior). 81
Figura 20 – Acurácia média obtida pela Random Forest usando NN-d para selecionar conjuntos de dados sintéticos. O conjunto de treinamento é inicializado com todos os conjuntos de dados originais, deixando apenas conjuntos de dados sintéticos para seleção. As linhas horizontais tracejadas representam a acurácia obtida usando todos os conjuntos de dados originais e sintéticos (parte superior) ou apenas os conjuntos de dados originais (parte inferior). 82
Figura 21 – Acurácia média obtida por KNN usando NN-d para selecionar conjuntos de dados sintéticos. O conjunto de treinamento é inicializado com todos os conjuntos de dados originais, deixando apenas conjuntos de dados sintéticos para seleção. As linhas horizontais tracejadas representam a acurácia obtida usando todos os conjuntos de dados originais e conjuntos de dados sintéticos (parte superior) ou apenas os conjuntos de dados originais (parte inferior). 83

LISTA DE TABELAS

Tabela 1 – Distribuição de classes (%) dos meta-dados relacionados aos conjuntos de dados originais e sintéticos 59
Tabela 2 – Parâmetros usados para otimizar subconjuntos de dados 64
Tabela 3 – Medidas de Caracterização 67
Tabela 4 – Distribuição de classes (%) dos meta-dados correspondentes aos conjuntos de dados originais e sintéticos 69
Tabela 5 – Acurácia média (%) obtida pelos algoritmos RF e KNN e o número médio de meta-exemplos em conjuntos de treinamento formados pela seleção de PU Learning. 75
Tabela 6 – Acurácia média (%) obtida pelos meta-aprendizes RF e KNN e o número médio de exemplos no conjunto de treinamento que consiste na seleção NN-d e instâncias originais. 78
Tabela 7 – Acurácia média (%) obtida pelos algoritmos Random Forest e KNN e o número médio de exemplos em conjuntos de treinamento formados pela seleção de PU Learning 80
Tabela 8 – Acurácia média (%) obtida por meta-alunos Random Forest e KNN e o número médio de exemplos no conjunto de treinamento que consiste em seleção NN-d e meta-exemplos originais. 83
Tabela 9 – Descrição das Bases de Dados 99
Tabela 10 – Características das Bases de Dados 107

LISTA DE ABREVIATURAS E SIGLAS

AM	*Aprendizado de Máquina*
CMA-ES	*Covariance Matrix Adaptation Evolution Strategy*
EE	*Estratégia de Evolução*
KNN	*k-nearest Neighbors*
MA	*Meta-Aprendizado*
MD	*Mineração de Dados*
NB	*Naive Bayes*
NN	*Nearest Neighbor*
NN-d	*Nearest Neighbor Data Description*
OCC	*One-Class Classification*
OCRF	*One Class Random Forest*
OCSVM	*One Class Support Vector Machine*
PU Learning	*Positive and Unlabeled Learning*
RF	*Random Forest*
SVDD	*Support Vector Data Description*
SVM	*Support Vector Machine*
UCI	*Machine Learning Repository*

LISTA DE SÍMBOLOS

$\mathcal{P}$	Instâncias positivas
$\mathcal{U}$	Instâncias desconhecidas
$\mathcal{S}$	Conjunto de Espiões
$\mathcal{N}_r$	Instâncias considerados verdadeiramente negativas
$\mathcal{U}_r$	Demais instâncias do conjunto $\mathcal{U}$ após retiradas as instâncias de $\mathcal{N}_r$
$\mathcal{U}_p$	Instâncias classificadas como positivas, relevantes
$\mathcal{V}_n$	Instâncias consideradas irrelevantes
γ	Tamanho da População
Λ	Melhores indivíduos da geração
δ	Passo de mutação
ϵ	Erro de tolerância
$maxAval$	Máximo de avaliações de função
D	Conjunto de dados real
Q	Número de classes
P	Número de atributos do conjunto de dados D
p	valor de uma característica
q	valor de uma classe
μ_{pq}	Valor médio de cada característica em cada classe
μ'_{pq}	Valor médio das características do novo indivíduo
x_p	Instância de uma classe
σ_{pq}	Desvio padrão original da característica p na classe q
λ	Indivíduos mais adequados da geração atual

c_{target}	Valor de complexidade alvo
C_M	Medida de complexidade
D_{new}	Conjunto de dados manipulado
$f(D_{new})$	Função objetivo

SUMÁRIO

1 INTRODUÇÃO 17
1.1 MOTIVAÇÃO 20
1.2 OBJETIVOS 20
1.3 ORGANIZAÇÃO 20
2 META-APRENDIZADO 22
2.1 META-APRENDIZADO PARA SELEÇÃO DE ALGORITMOS 23
2.1.1 O Problema de Seleção de Algoritmos 24
2.1.2 Técnicas de Geração de Conjuntos de Dados 27
2.1.3 Caracterização dos Dados 31
2.1.3.1 Direta 32
2.1.3.2 Landmarking 32
2.1.3.3 Via modelos 33
2.1.3.4 Medidas de Complexidade 34
2.1.4 Medidas de Avaliação 36
2.1.5 Formas de Recomendação 37
2.2 CONSIDERAÇÕES FINAIS 38
3 CLASSIFICAÇÃO DE UMA CLASSE 39
3.1 ESTRATÉGIAS DE APRENDIZAGEM EM CLASSIFICAÇÃO DE UMA CLASSE 41
3.1.1 Disponibilidade de Dados de Treinamento 42
3.1.2 Metodologias/Algoritmos Usados 44
3.1.2.1 Algoritmo Individual 44
3.1.2.2 Comitês de Máquinas 45
3.1.3 Domínios de Aplicação 46
3.1.3.1 Domínios de Classificação de Texto/Documentos 46
3.1.3.2 Classificação de Imagem 47
3.1.3.3 Outros Domínios de Aplicação 47
3.2 CONSIDERAÇÕES FINAIS 48
4 CLASSIFICAÇÃO DE UMA CLASSE PARA SELEÇÃO CONJUNTOS DE DADOS SINTÉTICOS 49
4.1 ABORDAGEM PROPOSTA 49

4.2 *PU LEARNING* . . . 53
4.3 *ONE-CLASS NEAREST NEIGHBOR* . . . 55
4.4 META-EXEMPLOS . . . 57
4.4.1 Estudo de Caso 1 . . . **57**
4.4.1.1 Manipulação de Dados . . . *57*
4.4.1.2 Caracterização de Dados . . . *58*
4.4.1.3 Avaliação de Algoritmo . . . *59*
4.4.2 Estudo de Caso 2 . . . **60**
4.4.2.1 Manipulação de Dados . . . *60*
4.4.2.2 Caracterização de Dados . . . *66*
4.4.2.3 Avaliação de Algoritmo . . . *67*
4.5 META-APRENDIZAGEM . . . 69
4.6 METODOLOGIA DE AVALIAÇÃO . . . 70
4.7 CONSIDERAÇÕES FINAIS . . . 71
5 RESULTADOS EXPERIMENTAIS . . . 72
5.1 RESULTADOS - ESTUDO DE CASO 1 . . . 72
5.1.1 PU learning . . . **72**
5.1.2 NN-d . . . **76**
5.2 RESULTADOS - ESTUDO DE CASO 2 . . . 78
5.2.1 PU learning . . . **79**
5.2.2 NN-d . . . **82**
5.3 DISCUSSÃO DOS RESULTADOS . . . 83
5.4 CONSIDERAÇÕES FINAIS . . . 85
6 CONCLUSÕES . . . 86
6.1 CONTRIBUIÇÕES . . . 87
6.2 TRABALHOS FUTUROS . . . 87
REFERÊNCIAS . . . 89
APÊNDICE A – PROBLEMAS DE CLASSIFICAÇÃO . . . 99

1 INTRODUÇÃO

No universo cotidiano, estamos constantemente realizando ações. Duas delas são frequentes e de grande importância: classificar (distribuir por classes) e tomar decisão. Tanto no processo de classificação quanto em um processo de tomada de decisão, somos orientados pela consideração das características envolvidas no problema específico. No caso de escolher uma fruta na feira, por exemplo, a classificação se dá em considerá-la apta ou não apta a ser consumida. No caso das frutas, algumas características observadas podem ser aparência, cor, consistência, tamanho, preço, dentre outras. Ao decidir em qual candidato votar em uma eleição, geralmente, considera-se a proposta de governo do candidato, seus atos ao longo de sua vida pública e suas ideias, por exemplo. A caracterização de um conjunto de objetos faz parte do processo de tomada de decisão em geral. Na área de Inteligência Artificial, mais precisamente, em *Aprendizado de Máquina* (AM), a caracterização é aplicada aos conjuntos de dados[1], estes relacionadas aos mais diversos problemas encontrados no mundo real.

É um lugar-comum que atualmente estamos enfrentando um momento especial de aplicação massiva do AM em diferentes domínios da ciência e da indústria. O crescente interesse exige soluções para apoiar o *design* de sistemas de aprendizagem e abordar questões relevantes, como quais algoritmos usar, como ajustá-los e como processar dados de treinamento. Essas soluções são cruciais, especialmente para usuários não especialistas em técnicas de AM. Na literatura, há um grande número de algoritmos de aprendizagem disponíveis para análise de dados. Saber qual algoritmo empregar não é uma tarefa trivial.

O foco desta tese é o problema de seleção de algoritmos, que tem sido tratado com sucesso usando *Meta-Aprendizado* (MA) (BRAZDIL et al., 2009). O campo do MA estuda como os algoritmos de AM podem se tornar mais eficazes por meio da experiência. A ideia principal é aprender sobre o desempenho do algoritmo de aprendizagem. O conhecimento obtido com a experiência pode ser útil em muitos cenários diferentes. MA tem sido investigado nos últimos anos com o objetivo de aprender modelos de seleção de algoritmos a partir de experiências na resolução de problemas passados (BRAZDIL; GIRAUD-CARRIER, 2018). Em MA, cada exemplo de treinamento (ou meta-exemplo) é associado a um problema de interesse (por exemplo,

[1] Registros dispostos em estrutura regular que possibilita a reorganização dos mesmos e produção de informação.

uma tarefa de classificação) e geralmente armazena: (1) as características do problema, ou seja os meta-atributos (por exemplo, número de exemplos, número de instâncias, correlações entre atributos); (2) um atributo-alvo, usualmente indicando o algoritmo avaliado com melhor desempenho empírico para o problema dentre um conjunto de algoritmos candidatos. A partir de um conjunto de meta-exemplos gerados a partir de problemas diversos, um meta-aprendiz é construído (por exemplo, um classificador para seleção do melhor algoritmo) relacionando características dos problemas com desempenho dos algoritmos candidatos. O interesse pelo MA tem aumentado ao longo dos anos devido à sua ampla aplicabilidade, especialmente para seleção de algoritmos (PRUDÊNCIO; LUDERMIR, 2004; BRAZDIL et al., 2009; ROSSI et al., 2014; CUNHA; SOARES; de Carvalho, 2018; ROSA-RIVERA et al., 2021)

O sucesso do MA, como em qualquer outra tarefa de aprendizagem, depende da quantidade e qualidade dos exemplos de treinamento (em nosso caso, um bom conjunto de meta-exemplos). Uma grande número de trabalhos anteriores (KALOUSIS; HILARIO, 2001; FILCHENKOV; PENDRYAK, 2015; PEREIRA; SANTOS; CARVALHO, 2021) é focado na construção e seleção de meta-características relevantes, mas poucos os artigos estão preocupados com as instâncias (problemas) usados para gerar o conjunto de meta-exemplos. Idealmente, os meta-exemplos devem ser gerados a partir de um conjunto de problemas representativo e grande o suficiente para resultar em um bom desempenho de MA. No entanto, em diferentes domínios, há uma baixa disponibilidade de instâncias de problemas reais ou conjuntos de dados de *benchmark* para produzir um bom (valioso e grande) conjunto de meta-exemplos (BRAZDIL et al., 2009). Cada meta-exemplo é produzido a partir de um conjunto de dados, geralmente armazenado em repositórios de *benchmark*. Nesse contexto, dependendo da classe do problema, a dificuldade está no número de conjuntos de dados disponíveis nos repositórios. Portanto, o número de meta-exemplos produzidos nem sempre é suficiente para treinar um bom meta-aprendiz (BRAZDIL et al., 2009).

Para contornar a dificuldade acima, dados sintéticos podem ser utilizados, em combinação com dados reais, para aumentar o número de meta-exemplos (SOARES, 2009; KANDA et al., 2011; REIF; SHAFAIT; DENGEL, 2012; SOUSA; PRUDÊNCIO; LUDERMIR, 2016). O uso de dados sintéticos, entretanto, pode gerar outra dificuldade: muitos dos conjuntos de dados sintéticos podem ser irrelevantes quando se considera as características de problemas reais. A distribuição de meta-atributos (REIF; SHAFAIT; DENGEL, 2012; MORENO-TORRES et al., 2012; WEN; GREI-

NER; SCHUURMANS, 2015) observada nos exemplos sintéticos pode ser bastante diferente em comparação com problemas reais. Selecionar apenas os conjuntos de dados sintéticos mais relevantes economizaria custos computacionais de realização de experimentos ao mesmo tempo, mantendo (ou mesmo melhorando) o desempenho de MA.

Em nosso trabalho, propomos uma nova solução para a seleção de conjuntos de dados sintéticos relevantes para MA, através da adoção de técnicas de classificação de uma classe *One-Class Classification* (OCC) (KHAN; MADDEN, 2014). Em OCC, assume-se a disponibilidade de instâncias pertencentes a uma classe de interesse (o conjunto positivo ou alvo), enquanto as outras instâncias não estão rotuladas e são referidas como conjunto desconhecido. OCC é aplicável em cenários nos quais as instâncias na classe negativa são difíceis de coletar ou são caras para rotular. A seleção de conjuntos de dados sintéticos para MA é tratada diretamente em nosso trabalho como uma tarefa OCC. Os conjuntos de dados reais disponíveis (por exemplo, conjuntos de dados de referência em repositórios) são considerados representantes de problemas relevantes, formando assim o conjunto positivo. Conjuntos de dados sintéticos, por sua vez, formam o conjunto desconhecido de instâncias. Na verdade, embora seja relativamente fácil produzir uma grande base de conjuntos de dados sintéticos, não se sabe a *priori* quais podem ser adotados com segurança como problemas de referência relevantes em experimentos. Em nossa proposta, uma técnica OCC recebe como entrada o conjunto positivo de meta-exemplos e é então usada para selecionar os conjuntos de dados sintéticos relevantes na classe desconhecida.

Com a finalidade de testar a viabilidade da proposta, realizamos dois estudos de caso. No estudo de caso 1, a técnica de manipulação de dados proposta em (SOARES, 2009) foi usada para gerar 983 conjuntos de dados sintéticos a partir de 64 conjuntos de dados reais. Adotamos duas técnicas de OCC para a seleção do conjunto de dados: a *Positive and Unlabelled learning (PU learning)* proposta em (LIU et al., 2003) e *Nearest Neighbor Data Description (NN-d)* de (TAX, 2001). Já no estudo de caso 2, propomos uma nova técnica de manipulação de dados reais baseada em algoritmo evolucionário para a geração de conjuntos de dados sintéticos. Esta nova abordagem de geração deu origem a 1114 conjuntos de dados sintéticos de 78 conjuntos de dados reais. A seleção dos conjuntos de dados foi realizada utilizando as mesmas técnicas de OCC adotadas no primeiro estudo de caso. Além disso, adotamos diferentes repositórios para coleta de dados reais e as medidas de complexidade também foram diferentes para cada

estudo de caso.

1.1 MOTIVAÇÃO

A presente pesquisa é motivada por duas questões importantes de meta-aprendizagem para seleção de algoritmos: (1) aumento do número de conjuntos de dados relevantes para produção de meta-exemplos e (2) redução do custo computacional da geração de meta-exemplos.

A geração de dados, manipulados ou sintéticos, favorece o aumento do número de exemplos disponíveis para MA, causando assim um potencial impacto positivo no desempenho no nível meta. Mesmo resultando em meta-exemplos relevantes, é possível porém que o processo de geração de bases de dados produza também meta-exemplos ruidosos ou redundantes (SOARES, 2009; SOUSA et al., 2013; ACOSTA et al., 2018). Além do mais, em MA, o processo de rotulagem dos meta-exemplos é realizado via avaliação empírica dos algoritmos postulantes nos conjuntos de dados disponíveis, o que pode ser computacionalmente custoso. Sendo assim, poder eliminar meta-exemplos irrelevantes e/ou redundantes antes do processo de rotulagem é uma boa estratégia para produzir meta-exemplos.

1.2 OBJETIVOS

De maneira mais geral, esta tese tem como propósito contribuir para o aprimoramento do estado da arte da área de MA em contexto de seleção de algoritmos de classificação. Isto posto, em termos mais específicos e em concordância com as lacunas apresentadas anteriormente, os objetivos são: (1) aumentar a disponibilidade de conjuntos de dados para produção de meta-exemplos; (2) propor estratégias de OCC para selecionar dados de treinamento relevantes em MA; (3) demonstrar que meta-exemplos selecionados melhoraram o desempenho dos modelos de MA; (4) reduzir o custo computacional no processo de rotulagem relacionado à produção de meta-exemplos.

1.3 ORGANIZAÇÃO

Além deste capítulo, esta tese está organizada da seguinte maneira:

Capítulo 02: apresenta conceitos gerais sobre meta-aprendizado e sua aplicação no âmbito da seleção de algoritmos. Aborda temas como: geração de bases de dados, caracterização de dados, medidas de avaliação, formas de sugestão.

Capítulo 03: neste capítulo são apresentados conceitos básicos sobre Classificação unária e seu relacionamento com a aprendizagem convencional. São apresentadas estratégias de OCC e uma nova taxonomia é organizada com suas três grandes categorias de estudo: Disponibilidade de dados de treinamento, Metodologias e Domínios de Aplicação.

Capítulo 04: apresenta a descrição da proposta de trabalho da tese, incluindo a definição do problema abordado e a solução proposta. Logo após, os métodos de OCC adotados na seleção de conjuntos de dados são descritos. O trabalho desenvolvido visando testar a viabilidade da proposta é apresentado. Os estudos de caso propostos são detalhados. Em seguida, a MA usada é detalhada e a metodologia adotada é descrita.

Capítulo 05: apresenta os resultados experimentais observados. Os resultados são apresentados e dispostos na perspectiva dos estudos de caso desenvolvidos. Apresenta, ainda, uma análise e discussão dos resultados.

Capítulo 06: este capítulo trata das considerações finais deste documento de tese. Apresenta as conclusões, contribuições e as possibilidades de trabalhos futuros

2 META-APRENDIZADO

Este capítulo faz uma revisão da literatura sobre os principais conteúdos em MA que serão abordados ao longo desta tese, sobretudo em relação ao MA aplicado à seleção de algoritmos.

Os avanços das ferramentas em *Mineração de Dados* (MD) e AM são caracterizados por inúmeros algoritmos, além da falta de regras sobre como selecionar o método certo de acordo com a natureza do problema em análise (SERBAN et al., 2013). Os problemas reais de aprendizagem trazem pouca ou nenhuma informação adicional sobre o tipo de método mais apropriado para a tarefa em questão. Assim, os profissionais e pesquisadores tendem a usar apenas alguns algoritmos disponíveis para análise de dados, esperando que o conjunto de suposições embutidas nesses algoritmos corresponda às características dos dados - em algumas situações utilizam procedimento de tentativa e erro (BRAZDIL et al., 2009). Tal prática tem estimulado os pesquisadores a investigar se o aprendizado de dados é feito de uma única camada operacional - busca por um bom modelo que se ajuste aos dados - ou se existem de fato várias camadas operacionais, ou seja, conhecimento acumulado sobre diversas tarefas, que podem ser exploradas para produzir um aumento no desempenho ao longo do tempo (BRAZDIL et al., 2009).

De maneira geral, a pesquisa busca entender se o processo de aprendizado pode ocorrer em um nível de abstração em que é possível aprender sobre o próprio processo de aprendizado e gerar conhecimento a partir da experiência de execuções anteriores. Sendo assim, um sistema pode aprender a lucrar com a experiência anterior para gerar conhecimento adicional que pode simplificar a seleção automática de modelos eficientes (BRAZDIL et al., 2009). Essa solução é alcançada com a construção de um sistema de MA (BRAZDIL et al., 2009; VANSCHOREN, 2018). Um sistema de MA armazena conhecimento sobre o processo de aprendizado, o chamado meta-conhecimento.

Os algoritmos de AM funcionam adaptando seus parâmetros a um ambiente específico (MITCHELL, 1997), ou seja, o foco da aprendizagem de métodos tradicionais é na adaptação do modelo a uma determinada tarefa. Por outro lado, MA faz uso de modelos de aprendizado induzidos a partir de meta-conhecimento, com o ojetivo de acumular experiências

que resultam de várias aplicações de um sistema de MA (VILALTA; DRISSI, 2002).

O que distingue o MA da aprendizagem de base é o escopo de seu nível de adaptação. A aprendizagem no nível meta é baseada na escolha da polarização ou *bias* correto de forma dinâmica através das múltiplas aplicações de um algoritmo de aprendizado, em oposição à aprendizagem de base, onde o *bias* é fixado a *priori*, ou através de parâmetros, ou seja, o MA é baseado no acúmulo de experiência, enquanto o aprendizado convencional trabalha sobre um conjunto de dados por vez.

MA ainda pode ser associado a métodos de aprendizado com a capacidade de: avaliar e comparar seus resultados; medir o benefício de aprendizados anteriores em aprendizados subsequentes e comparar técnicas de aprendizado para selecionar a melhor entre elas (SCHMIDHUBER; WIERING, 1996; MIRANDA et al., 2014; MIRANDA; PRUDÊNCIO; PAPPA, 2017).

MA tem sido aplicada a diversas categorias de problemas. Dentre as aplicações mais comuns, tem-se o problema de gerar regras que tenham a capacidade de relacionar o desempenho de algoritmos de AM com as propriedades das bases de dados (BRAZDIL et al., 2009). Levando em consideração esta aplicação, podemos utilizar tal conceito como forma de ajuda na criação de sistemas que forneçam sugestões sobre quais algoritmos utilizar em situações específicas. No caso da presente tese, auxiliar na escolha do melhor modelo para um determinado conjunto de dados de teste. Neste contexto, alguns conceitos e tópicos relacionados à seleção de algoritmos são relevantes. Estes serão abordados nas seções seguintes deste capítulo.

2.1 META-APRENDIZADO PARA SELEÇÃO DE ALGORITMOS

MA busca descobrir como os algoritmos de AM podem aumentar sua eficiência mediante a experiência. O MA (BRAZDIL et al., 2009; BRAZDIL; GIRAUD-CARRIER, 2018) utiliza técnicas de AM para obter modelos preditivos que associam características de tarefas ao respectivo desempenho de algoritmos.

Na última década, os progressos no campo do AM têm contribuído para a proposição de diversos métodos e algoritmos na literatura, conforme mencionado anteriormente. O principal desafio, porém, consiste em escolher um método específico ou saber qual ferramenta utilizar

para um novo problema de classificação, que é representado por um determinado conjunto de dados, de forma que as soluções oferecidas sejam adequadas e eficientes.

Cada problema tem suas particularidades, levando em consideração a natureza do conjunto de dados representado. Além disso, cada algoritmo de aprendizagem possui a chamada superioridade seletiva, o que pode tornar seu desempenho melhor do que os demais em uma classe de tarefas específicas (BRODLEY, 1995). Para esses casos, o uso do MA pode ser uma solução eficiente, pois o modelo de aprendizagem pode usar suas experiências anteriores de aprendizado para selecionar algoritmos de forma automática.

2.1.1 O Problema de Seleção de Algoritmos

Diante dos avanços em AM, uma grande variedade de algoritmos que resolvem um mesmo problema está disponível. algoritmos de classificação, como redes neurais artificiais, árvores de decisão e máquinas de vetores de suporte, podem ser aplicados com o objetivo de auxiliar a tomada de decisão em diagnóstico médico ou análise de crédito, por exemplo. Em AM, o problema de seleção de algoritmo é claramente um problema de aprendizado.

O termo Meta-aprendizado, aprender sobre aprender, foi usado pela primeira vez neste contexto por Aha (1992), seguido pelo projeto europeu StatLog, cujo objetivo era relacionar o desempenho de algoritmos a características ou medidas de conjuntos de dados de classificação. Apesar disso, Rice (1976) já trazia em suas ideias conexão e semelhança de objetivos com o MA (SMITH-MILES, 2008). Ao apresentar métodos para determinar qual algoritmo deve ser selecionado, os pesquisadores do projeto StatLog (MICHIE et al., 1994) reconheceram o problema de seleção de algoritmo de Rice (1976). Em Rice (1976), o problema de seleção de algoritmos foi primeiramente caracterizado formalmente definindo os seguintes espaços de busca:

- o espaço de problemas P, representando o conjunto de instâncias de um problema;
- o espaço de características F, contendo características mensuráveis das instâncias geradas por um processo de extração de características aplicado a uma instância de P;

- o espaço de algoritmos A, que contém todos os algoritmos capazes de resolver um determinado problema;
- o espaço de desempenho Y, que representa todas as medidas de desempenho possíveis.

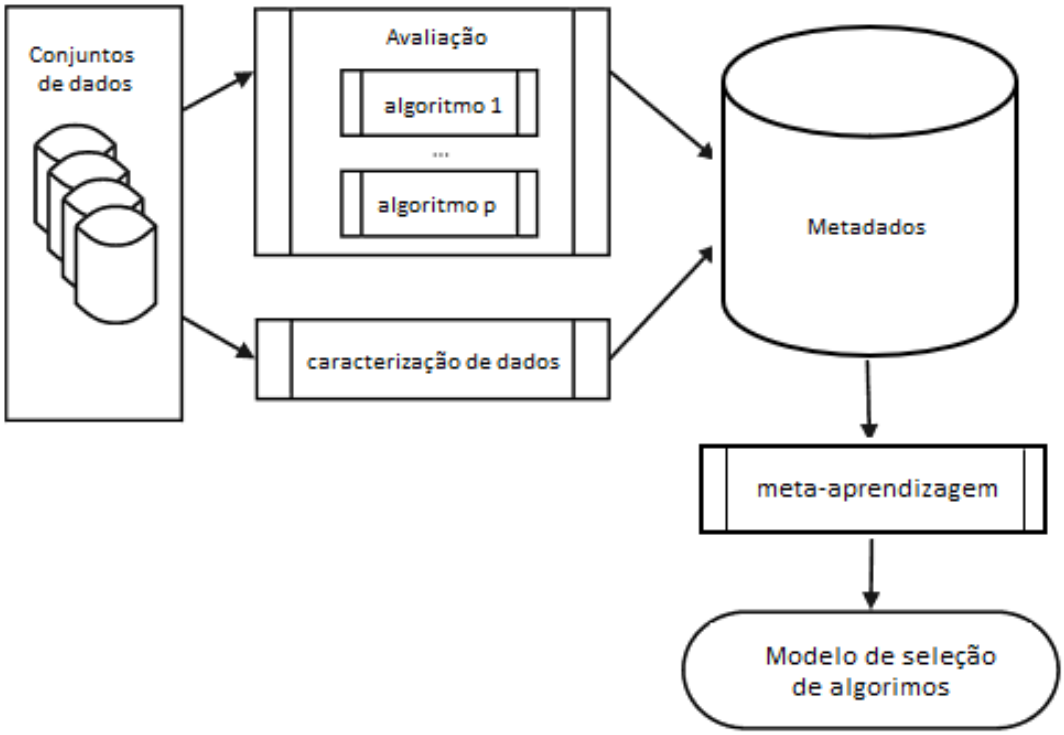
Embora raramente a literatura de MA apresente referências ao modelo de Rice (1976) do problema de seleção de algorítmos, é interessante considerar a notação e o modelo que ele introduziu como parte de uma linguagem comum para descrever as diversas tentativas de diferentes áreas de selecionar o algoritmo mais adequado para diferentes tarefas. Pensando MA à luz do modelo de Rice (1976), na comunidade de AM há interesse em problemas de classificação $x \in P$, com características $f(x) \in F$, resolvidos usando uma variedade de algoritmos de aprendizado A, com medidas de desempenho geralmente relacionadas à acurácia de classificação Y. Em Smith-Miles (2009) é apresentada uma estrutura unificada que considera o problema de seleção de algoritmo como um problema de aprendizagem e ilustra o desenvolvimento interdisciplinar no tratamento do problema de seleção de algoritmo.

O problema da seleção de algoritmos, ao ser reconhecido como uma tarefa de aprendizado no início dos anos 90, a comunidade de AM desenvolveu o campo do MA, focado em aprender sobre o desempenho de algoritmos de aprendizagem em problemas de classificação, elevando o nível de abstração do processo de aprendizado e fazendo com que o modelo aprenda sobre o próprio processo de aprendizagem em si. A Figura 1 ilustra o processo geral de MA para seleção de algoritmos.

Em contexto geral, o processo de MA para seleção de algoritmos inicia-se com a aquisição de um conjunto de problemas que represente, apropriadamente, os problemas para os quais a seleção será feita. O próximo passo é, a partir de cada base de dados, avaliar o algoritmo empregado e extrair as características, de acordo com as métricas pré-estabelecidas. Obtendo-se essas duas informações (desempenho do algoritmo e características do conjunto de dados) de cada problema, gera-se um meta-exemplo, o qual é formado pelas meta-características de entrada e pela meta-classe[1], respectivamente. Ao conjunto dos meta-exemplos disponíveis dá-se o nome de meta-dados. Com a finalidade de induzir o mapeamento entre as meta-características de entrada e meta-classes, é aplicado um algoritmo de AM, o qual é chamado de meta-aprendiz. Por meio dele, é possível utilizar o meta-conhecimento obtido do processo de aprendizagem e realizar seleção de algoritmos no contexto de MA.

[1] Rótulo indicando o melhor algoritmo para o problema

Figura 1 – Processo de meta-aprendizagem para seleção de algoritmos



Fonte: Produzido pela autora baseado em (BRAZDIL et al., 2009)

O processo de aquisição automática de conhecimento (ou meta-conhecimento) em MA é é realizado a partir de uma avaliação empírica de algoritmos de treinamento em diferentes problemas de aprendizagem (conjuntos de dados). Um sistema de MA realiza a seleção de modelos efetivamente por meio de um mapeamento entre um determinado problema e um modelo preditivo (ou uma combinação de modelos). A função preditiva é modelada na forma de um problema de aprendizagem supervisionada.

A geração de conhecimento em MA (meta-conhecimento) é composto basicamente por qualquer informação extraída a partir da execução de um modelo preditivo em uma tarefa de aprendizado. Esse meta-conhecimento pode ser utilizado para gerar um modelo que relacione características de problemas a uma classe de saída. Dessa maneira, há uma estreita ligação entre a qualidade de um modelo em MA e o meta-conhecimento adquirido durante a etapa de aprendizagem. Em recomendação de algoritmos, por exemplo, o meta-conhecimento é utilizado para construir a função de mapeamento entre características de problemas e desempenho de algoritmos (SOARES, 2004).

Na literatura, diversos trabalhos têm investigado MA de maneira mais profunda em tarefas de classificação e regressão e tem sido aplicada à recomendação de algoritmos em vários domínios de aplicações, como seleção de modelos de séries temporais (PRUDÊNCIO; LUDERMIR, 2004), análise de expressão gênica (SOUZA; SOARES, 2009; NASCIMENTO et al., 2009), configuração de medidas de caracterização de comitês de classificadores homogêneos (PARENTE; CANUTO; XAVIER-JUNIOR, 2013), algoritmos de otimização para o problema de seleção de parâmetros do *Support Vector Machine* (SVM) (MIRANDA et al., 2014), hiper-heurística híbrida para projeto de algoritmo (MIRANDA; PRUDÊNCIO; PAPPA, 2017), seleção de algoritmo para resolver problemas de cronogramas educacionais (ROSA-RIVERA et al., 2021).

Uma questão importante no MA está relacionada à disponibilidade de conjuntos de dados em número suficiente para permitir a indução confiável do meta-aprendiz. Além disso, o processo de MA é fortemente dependente de instâncias de problemas relevantes. Na literatura é possível encontra um número significativo de trabalhos que tratam da geração de conjuntos de dados para MA (SOARES, 2009; MACIÀ; BERNADÓ-MANSILLA, 2014; LORENA et al., 2018; MELO; LORENA, 2018), contudo a qualidade dos meta-exemplos (SOUSA et al., 2013; SOUSA; PRUDÊNCIO; LUDERMIR, 2016) gerados a partir desses conjuntos de dados não tem recebido muita atenção em trabalhos anteriores sobre MA. A próxima seção trata da questão da geração de conjuntos de dados em MA.

Maiores informações sobre MA para seleção de algoritmos podem ser encontradas em (BRAZDIL et al., 2009; BRAZDIL; GIRAUD-CARRIER, 2018; VANSCHOREN, 2018) e referências citadas.

2.1.2 Técnicas de Geração de Conjuntos de Dados

Uma questão a ser considerada ao gerar conjuntos de dados com o objetivo de obter meta-conhecimento geral é a relevância. A questão é se os conjuntos de dados gerados são representativos da população de conjuntos de dados do mundo real, ou pelo menos uma parte interessante deles. O meta-conhecimento gerado a partir de um conjunto de conjuntos de dados só é útil na prática se eles forem representativos de problemas do mundo real (SOARES, 2009).

A disponibilidade de conjuntos de dados em MA é uma questão extremamente importante, uma vez que o processo de MA é fortemente dependente de instâncias de problemas relevantes e em número suficiente, como dito anteriormente, pois o desempenho de um modelo depende da qualidade dos dados de entrada na etapa de treinamento. Intuitivamente, buscar dados em repositórios como *Machine Learning Repository* (UCI) (LICHMAN, 2013) ou OpenML (VANSCHOREN et al., 2013), por exemplo, pode ser uma boa alternativa a ser considerada. Embora intuitivo e conveniente, dependendo da classe do problema, pode ser difícil encontrar um número elevado de instâncias de problemas para geração de meta-dados significativos para obtenção de um modelo confiável. Sendo assim, a maioria das pesquisas em MA é realizada com uma quantidade pequena de exemplos, o que pode comprometer a confiabilidade dos meta-aprendizes (SOARES, 2009).

Nos últimos anos, a temática da geração de bases de dados para MA tem recebido uma crescente atenção e tem sido resolvida de forma satisfatória com técnicas de geração de dados. Na literatura, abordagens de geração de problemas sintéticos são consideradas para contornar o assunto tratado anteriormente (MACIÀ; ORRIOLS-PUIG; BERNADÓ-MANSILLA, 2008; KANDA et al., 2011; MACIÀ; BERNADÓ-MANSILLA, 2014; LORENA et al., 2018; MELO; LORENA, 2018; ACOSTA et al., 2018). Basicamente, são duas as formas tradicionais para gerar novos conjuntos de dados e aumentar sua disponibilidade em MA: usar um modelo de geração de dados sintéticos (REIF; SHAFAIT; DENGEL, 2012; MACIÀ; ORRIOLS-PUIG; BERNADÓ-MANSILLA, 2008; LIANG et al., 2013) ou manipular conjuntos de dados reais existentes por meio de alguma técnica de manipulação de dados (HILARIO; KALOUSIS, 2000; BLOCKEEL; VANSCHOREN, 2007; KANDA et al., 2011).

De maneira geral, a abordagem de geração de conjuntos de dados que envolve um modelo de geração de dados permite produzir novas bases com meta-atributos específicos, definidos previamente ou fornecidos pelo usuário. Em Reif, Shafait e Dengel (2012) é apresentado um gerador de conjuntos de dados com caraterísticas fornecidas pelo usuário, fazendo uso de uma abordagem genética e também é capaz de incorporar meta-características arbitrárias. Os valores de meta-atributos são fornecidos pelo usuário. Cada conjunto de dados é representado como um cromossomo. A definição da função *fitness* do modelo leva em conta a similaridade entre os meta-atributos obtidos a partir dos conjuntos de dados produzidos e os valores desejados. Já em Macià, Orriols-Puig e Bernadó-Mansilla (2008), um método é proposto para gerar

conjuntos de dados sintéticos com uma complexidade específica, com base no comprimento do limite da classe. A técnica ainda é utilizada em Liang et al. (2013) para geração de dados de ventos para séries temporais usando Cadeias de Markov. Por outro lado, Melo e Lorena (2018) apresentam um método para gerar conjuntos de dados sintéticos de classificação trocando os rótulos dos conjuntos de dados feitos sinteticamente, a fim de atingir determinados valores de medidas de complexidade (LORENA et al., 2018). Uma desvantagem da abordagem é que ela não leva em consideração nenhuma informação para realizar a troca de etiquetas. Garante apenas a troca de etiquetas diferentes.

Embora essas técnicas de geração tenham um bom êxito, gerando uma grande quantidade de dados, apresentam uma limitação geral, que é a necessidade de possuir conhecimento prévio ou modelo sobre como os dados são produzidos, criando, assim, uma forte dependência de decisões humanas ou mesmo possuir um grau de aleatoriedade elevado.

A manipulação de conjuntos de dados reais para produzir novos conjuntos de dados sintéticos não é uma ideia recente, como pode ser visto em Hilario e Kalousis (2000) e em Blockeel e Vanschoren (2007). As mudanças em abordagens típicas podem ocorrer adicionando atributos independentes redundantes (HILARIO; KALOUSIS, 2000) ou adicionando ruído ao atributo alvo (BLOCKEEL; VANSCHOREN, 2007). Nesse caso, o meta-conhecimento que pode ser obtido a partir de tais conjuntos de dados é centrado em um aspecto específico do comportamento dos algoritmos, como resiliência a atributos redundantes, por exemplo. Mesmo útil, é muito específico para fornecer diferentes *insights* sobre o desempenho geral dos algoritmos. Uma técnica de manipulação de conjuntos de dados para geração de novas bases de dados proposta em Soares (2009), lança mão de instâncias reais de problemas (ou seja, uma base de dados de classificação) para produzir novas bases por meio de uma alteração nos dados. O método proposto gera novos conjuntos de dados (os chamados *Datasetoids*) permutando o atributo alvo na base de dados real com um atributo de entrada simbólico.

O método que manipula conjuntos de dados reais para criação de novas bases de dados proposto em Soares (2009) foi amplamente usado, como pode ser visto em demonstrações experimentais em trabalhos anteriores envolvendo problemas de MA (SOARES, 2009; PRUDÊNCIO; SOARES; LUDERMIR, 2011; SOUSA; PRUDÊNCIO; LUDERMIR, 2016). Os conjuntos de dados sintéticos produzidos pelo método (SOARES, 2009) podem ser úteis em situações em

que é necessário realizar experimentos em um grupo de conjuntos de dados para obter meta-conhecimento, ou seja, relacionar propriedades (meta-características) dos dados com o desempenho (relativo) dos algoritmos. A técnica tem como motivação a baixa disponibilidade de instâncias de problemas em MA e a necessidade de aumentar o número de conjuntos de dados disponíveis. Ela cria novos elementos na base, e assim um aumento efetivo dos conjuntos de dados iniciais, os quais proporcionam melhorias na qualidade da predição dos modelos de MA. É possível utilizar a técnica em outros domínios de aplicação.

Experimentos realizados em Soares (2009) com o intuito de decidir quando podar uma árvore de decisão utilizando meta-dados contendo conjuntos de dados sintéticos apresentaram resultados mais promissores que aqueles realizados apenas com bases de dados reais. Nesses experimentos, o modelo de MA obteve ganho de desempenho que variou entre 7% e 17%, quando utilizou dados sintéticos em relação a utilização apenas de dados do UCI. Apesar dos resultados expressivos, o método de geração apresenta alguns aspectos negativos que devem ser considerados. Os conjuntos de dados sintéticos gerados podem conter anomalias, como por exemplo, valores faltantes no atributo simbólico usado como atributo alvo no novo conjunto de dados de classificação. Nesse caso uma possível solução seria descartar esse conjunto de dados ou simplesmente remover os exemplos com dados faltantes no atributo. Adotando-se a segunda solução, deve-se observar o número de exemplos que restaram no conjunto de dados sintético. Caso não sejam exemplos suficientes para indução do modelo, a primeira solução será a mais viável. Outros detalhes sobre a abordagem podem ser vistos em Soares (2009).

Em Macià e Bernadó-Mansilla (2014), após uma avaliação experimental do repositório UCI, foi identificado que a maioria dos problemas do repositório são fáceis de aprender, o que significa que 97% deles oferecem uma estrutura de fácil aprendizagem para os classificadores. Conjuntos de dados sintéticos visando cobrir o espaço de complexidade são sugeridos por (MACIÀ; BERNADÓ-MANSILLA, 2014) a fim de aumentar a diversidade do repositório UCI. Um conjunto de dados semente é calibrado pela evolução de cada observação realizada por um algoritmo genético multi-objetivo. Uma questão a ser tratada na abordagem proposta é que conforme o número de observações aumenta, o processo de evolução se torna intratável. Em Lorena et al. (2018), dados sintéticos são usados para auxiliar na análise de meta-características empregadas para quantificar a dificuldade em problemas de regressão.

Em nosso segundo estudo de caso, Uma *Estratégia de Evolução* (EE) (HANSEN, 2006) é adotada para evoluir conjuntos de dados reais com o intuito de obter conjuntos de dados sintéticos com os níveis desejados de complexidade (LORENA et al., 2018). Os detalhes nossa proposta de manipulação de conjuntos de dados reais são fornecidos no capítulo que trata das configuraçcoes dos experimentos. A próxima subseção apresenta uma revisão das técnicas de caracterização de dados.

2.1.3 Caracterização dos Dados

Dado que os algoritmos de AM têm seu desempenho influenciado pelas características do problema considerado, o processo de caracterização consiste em identificar e extrair características destes dados que sejam relevantes para o desempenho desses algoritmos. Além disso, o processo de caracterização de dados deve apresentar baixo custo computacional(SOARES; BRAZDIL; KUBA, 2004). Uma vez de posse das características contendo informações dos dados, técnicas de MA são aplicadas, levando em consideração o conhecimento a *priori* do comportamento dos algoritmos, sob um aspecto em particular.

Considerando-se o fato de que cada algoritmo tem um comportamento específico diante da natureza dos dados a ele apresentados, faz-se necessário explorar as propriedades que caracterizam os dados com o intuito de compreender o desempenho dos algoritmos de classificação. Isso é possível através de algumas métricas usadas para explorar a natureza dos dados. Segundo Soares, Brazdil e Kuba (2004), tais medidas apresentam informações relevantes para determinar o desempenho relativo entre os algoritmos de classificação e não apenas gerar um custo computacional. A pesquisa em caracterização de dados tem apresentado ao longo dos anos diversas categorias e medidas de caracterização. Em Vilalta, Giraud-Carrier e Brazdil (2005) a pesquisa em caracterização divide-se em 3 categorias (VILALTA; GIRAUD-CARRIER; BRAZDIL, 2005): caracterização direta; caracterização baseada em *landmarking* e caracterização via modelos, as quais serão apresentadas de maneira breve nas subseções seguintes.

2.1.3.1 Direta

O Projeto STATLOG (MICHIE et al., 1994) foi uma tentativa de gerar relacionamento entre as métricas que caracterizam as bases de dados e o desempenho dos algoritmos. Um dos seus principais objetivos era saber porque alguns algoritmos classificavam bem em alguns domínios e apenas regularmente em outros. É assumido que em problemas semelhantes, o perfil de desempenho dos algoritmos também deverão ser semelhantes. Foram utilizados 23 algoritmos e 21 bases de dados nos experimentos empíricos. Neste projeto as medidas utilizadas na caracterização dos dados foram divididas em três categorias:

- ***Simples***: incluem medidas gerais, como número de atributos, números de instâncias, número de classes e número de atributos binários;

- ***Estatísticas***: envolvem conceitos estatísticos, como razão média entre desvio padrão dos atributos, correlação média absoluta entre atributos por classe, primeira correlação canônica, proporção de variância explicada pelo 1^o discriminante canônico, assimetria média absoluta dos atributos e curtose média dos atributos;

- ***Baseadas na teoria da informação***: utilizadas para caracterizar os atributos nominais e sua relação entre si e com o atributo classe. São elas: entropia normalizada das classes, entropia média dos atributos, informação mútua média entre classe e atributos, razão sinal/ruído.

2.1.3.2 Landmarking

Outra fonte de caracterização se enquadra no conceito de *landmarking* (PFAHRINGER; BENSUSAN; GIRAUD-CARRIER, 2000). Cada algoritmo tem um tipo de dados em que ele tem um desempenho razoavelmente bom, isso é denominado área de competência ou área de especialização da classe de um algoritmo. A ideia básica da abordagem *landmarking* é que o desempenho de um algoritmo em uma tarefa descobre informações sobre a natureza da

tarefa. Assim, uma tarefa pode ser descrita pela coleção das áreas de conhecimento a que pertence. Desta forma, por meio da aplicação de algoritmos simples, os *landmarkers*, seria possível obter informação importante sobre a natureza do domínio em que eles são aplicados.

O *landmarking* é utilizado para determinar a proximidade de uma base de dados em relação a outras, por meio da similaridade entre os desempenho dos algoritmos. Forma-se uma vizinhança de áreas de competência, onde bases de dados podem ser representadas. Espera-se que bases de dados de natureza semelhante pertençam às mesmas áreas de competência e, consequentemente, os mesmos algoritmos de classificação sejam adequados à elas. O papel dos algoritmos de meta-aprendizagem é explorar quão bem as informações dos *landmarkers* podem ser utilizadas para localizar as bases de dados no espaço de competências. Em Bensusan e Giraud-Carrier (2000) é encontrado um exemplo de utilização de *landmarking* como técnica de caracterização de dados em MA.

2.1.3.3 Via modelos

A abordagem de caracterização via modelos (BENSUSAN; GIRAUD-CARRIER; KENNEDY, 2000) difere da chamada *landmarking* (PFAHRINGER; BENSUSAN; GIRAUD-CARRIER, 2000) pelo fato de não considerar as medidas de desempenho do classificador induzido e sim a estrutura do próprio classificador, conhecida como hipótese induzida ou modelo. Esta é uma forma alternativa de representar bases de dados fazendo uso de algoritmos de classificação. Existem várias vantagens na caracterização via modelos, dentre as quais se destacam (VILALTA; GIRAUD-CARRIER; BRAZDIL, 2005): a base de dados é representada por uma estrutura que contém informações sobre a complexidade e desempenho do modelo e; a representação dos dados nessa forma pode servir de base para explicar o desempenho do algoritmo de aprendizagem.

A utilização de modelos para a caracterização de bases de dados realiza uma mudança no espaço de busca do algoritmo de meta-aprendizagem, passando do espaço de exemplos para o espaço de hipóteses do algoritmo de caracterização (BENSUSAN; GIRAUD-CARRIER; KENNEDY, 2000). Uma vez que, o algoritmo, idealmente, é capaz de realizar busca eficiente em seu rico espaço de hipóteses, espera-se que a utilização da hipótese induzida comprima a base de dados original de maneira a proporcionar meta-características que contribuam mais com informações

ao algoritmo de meta-aprendizagem.

O algoritmo mais utilizado, dentre os algoritmos de classificação, para realizar a caracterização de bases de dados via modelos é a árvore de decisão. Evidências empíricas justificam este fato, apontando para a existência de fortes relações entre as propriedades das bases de dados e as estruturas de árvores de decisão sem podas (BENSUSAN; GIRAUD-CARRIER; KENNEDY, 2000). Diversas medidas de uma árvore de decisão podem ser utilizadas como meta-características, tais como (VILALTA; GIRAUD-CARRIER; BRAZDIL, 2005): o número de nós por atributo, a profundidade máxima da árvore, o grau de balanceamento, dentre outras.

2.1.3.4 Medidas de Complexidade

Segundo LORENA et al. (2018), os descritores de dados geométricos e estatísticos estão entre aqueles mais utilizados na caracterização da complexidade dos problemas de classificação. Esses descritos são referidos na literura como medidas de complexidade.

Em MA, soluções de problemas anteriores produzem meta-conhecimento, o qual é usado para auxiliar na solução de um novo problema (VILALTA; DRISSI, 2002). A meta-base usada em MA é composta por meta-características extraídas dos conjuntos de dados para os quais as soluções são conhecidas. Sendo assim, as medidas de complexidade (LORENA et al., 2018) são usadas como meta-características em MA.

Um trabalho fundamental na análise da complexidade de um problema de classificação usando descritores pode ser encontrado em Ho e Basu (2002). A complexidade de um problema de classificação pode ser atribuída a uma combinação de três fatores principais, segundo Ho e Basu (2002): (i) a ambiguidade das classes; (ii) a dispersão e dimensionalidade dos dados; e (iii) a complexidade da fronteira que separa as classes.

Ho e Basu (2002) apresentam medidas de caracterização com ênfase nas características geométricas das distribuições de classes. Tais medidas são capazes de descrever como as classes são separadas, ou seja, a complexidade da fronteira, em problemas de classificação binária. Trabalhos posteriores, como (MOLLINEDA; SÁNCHEZ; SOTOCA, 2005; MOLLINEDA;

SÁNCHEZ; SOTOCA, 2006; HO; BASU; LAW, 2006; ORRIOLS-PUIG; MACIÀ; HO., 2010) estendem essas medidas a problemas de classificação multiclasse. Ho e Basu (2002) dividem suas medidas em três categorias principais: (a) medidas de sobreposição de atributos; (b) medidas de separabilidade de classes; e (c) medidas de geometria, topologia e densidade de *manifolds*.

Na literatura, diversos trabalhos já fizeram uso de medidas de complexidade para caracterização de dados em MA. Mollineda, Sánchez e Sotoca (2006) apresenta uma estrutura geral de MA baseada em conjunto de medidas de complexidade de dados. Cavalcanti, Ren e Vale (2012) usam sete medidas de complexidade de dados para prever o comportamento do classificador *Nearest Neighbor* (NN). Garcia, Carvalho e Lorena (2016) apresentam um sistema de recomendação de MA capaz de prever o desempenho esperado de filtros de ruído em tarefas de identificação de dados ruidosos. Para tal, é criada uma meta-base, contendo meta-características extraídas de vários conjuntos de dados corrompidos juntamente com o desempenho de alguns filtros de ruído quando aplicados a esses conjuntos de dados. Uma proposta e avaliação de uma arquitetura para recomendação de algoritmos de seleção de atributos baseados no uso de MA é apresentada em Parmezan, Lee e Wu (2017).

Em LORENA et al. (2018) as medidas de complexidade são categorizadas de uma maneira mais ampla, como segue:

1. **Medidas de sobreposição de atributos:** essas medidas avaliam o poder discriminativo dos atributos, caracterizando o quão informativos são os atributos disponíveis para a separação das classes;

2. **Medidas de linearidade:** essas medidas procuram identificar o quanto as classes são linearmente separáveis, ou seja, se é possível separar as classes por um hiperplano;

3. **Medidas de vizinhança:** caracterizam a presença e densidade de classes iguais ou diferentes nas vizinhanças;

4. **Medidas de rede:** extraem informações estruturais do conjunto de dados modelando-o como um gráfico.

5. **Medidas de dimensionalidade:** avaliam a dispersão dos dados baseando-se no número de amostras em relação à dimensionalidade dos dados.

6. **Medidas de equilíbrio de classes:** consideram a razão do número de exemplos entre as classes.

As descrições de dados levantadas em LORENA et al. (2018) são disponibilizadas na ferramenta ECoL [2](GARCIA; LORENA; LEHMANN, 2019) que fornece um conjunto de 22 medidas de complexidade de dados.

Em Pimentel e de Carvalho (2019) uma nova abordagem para caracterização de dados é proposta. Uma novo conjunto de meta-características é proposto combinando correlação e dissimilaridade. Este conjunto de meta-características não usa nenhuma informação sobre os rótulos de classe das instâncias para extrair as medidas. A principal motivação para combinar as medidas de correlação e dissimilaridade é coletar mais informações dos dados, melhorando assim o desempenho do sistema de recomendação do(s) algoritmo(s) de AM mais adequado(s) para um novo conjunto de dados.

2.1.4 Medidas de Avaliação

No processo de decisão, seja ele qual for, é necessário tomar algumas medidas que orientem o caminho a seguir, ou qual decisão tomar. No processo que determinará quais algoritmos utilizar em quais bases de dados, também se faz necessário especificar quais medidas de desempenho serão consideradas, a fim de que se possa estabelecer uma lista de preferência de algoritmos. Existem diversas medidas que podem ser consideradas, tais como: acurácia/erro, a área sob as curvas ROC, $F - Score$, erro médio quadrático, dentre outras, como visto em Caruana e Niculescu-Mizil (2006). Embora a utilização de apenas uma medida de avaliação seja a prática mais comum em meta-aprendizagem, é possível utilizar abordagens multi-objetivas, em que duas ou mais medidas são combinadas. A utilização de múltiplas medidas pode permitir a exploração de diversos aspectos do classificador. Em alguns domínios, outras medidas podem ser interessantes como, por exemplo, o tempo requerido para o algoritmo construir um classificador, o tempo requerido para o classificador rotular um exemplo, a quantidade de memória requerida pelo algoritmo e a simplicidade e interpretabilidade dos classificadores construídos, dentre outras. Em Soares e Brazdil (2000) foi utilizada como critério uma medida

[2] Disponível em: https://github.com/SmartDataAnalytics/ECoL. Acessado em 25/10/2021.

chamada *Adjusted Ratio of Ratios*, que combina tempo de execução e acurácia. Neste trabalho, usaremos como medida de avaliação a média simples da taxa de acurácia de cada método.

2.1.5 Formas de Recomendação

Segundo Soares (2004), de maneira geral, há quatro diferentes abordagens para sugestão de algoritmos em meta-aprendizagem: um algoritmo único, um conjunto de algoritmos, um *ranking* de algoritmos e a performance estimada dos algoritmos.

A primeira consiste em informar o melhor algoritmo, ou seja, aquele que gera, hipoteticamente, o melhor modelo para uma dada tarefa ou base de dados, de acordo com algum critério. Exemplos podem ser encontrados em Koepf, Taylor e Keller (2000). Uma limitação seria a eventual indisponibilidade do algoritmo em um dado momento, não sendo possível sua aplicação.

Uma abordagem mais flexível seria a segunda, que informa, dentre os algoritmos considerados, o conjunto de algoritmos que apresentam melhor estimativa de desempenho na base de dados. Além do melhor algoritmo, estão presentes neste conjunto também os algoritmos que não possuam desempenho estatisticamente inferior ao melhor. No caso em que o usuário esteja disposto a experimentar mais de um algoritmo, esse tipo de recomendação não fornece nenhuma orientação quanto à ordem em que os experimentos devem ser realizados. Como exemplo, citamos o projeto StatLog, no qual as sugestões eram apresentadas dessa forma.

Uma terceira abordagem mostra os algoritmos em ordem de preferência com relação à base de dados. A ordem de preferência pode ser estabelecida tomando por base a acurácia dos classificadores, por exemplo, como em Bensusan e Kalousis (2001), ou medidas mais complexas, tais como tempo de execução do algoritmo ou o quão fácil o modelo gerado pode ser interpretado.

Por fim, um algoritmo de regressão pode ser utilizado no meta-nível. Com essa abordagem, o método prevê o desempenho dos algoritmos. O desempenho estimado pode ser utilizado para gerar *ranking* dos algoritmos (SOUZA, 2010; KALOUSIS, 2002).

Nesta teste, a sugestão será construída utilizando a primeira forma de sugestão, oferecendo a recomendação do melhor modelo para um determinado conjunto de dados de teste.

2.2 CONSIDERAÇÕES FINAIS

Este capítulo discorreu sobre os principais tópicos relacionados à MA abordados nesta tese. Inicialmente, foram apresentados conceitos gerais sobre MA e sua aplicação no âmbito da seleção de algoritmos. Foi realizada uma introdução do conceito de seleção de algoritmos a partir do modelo sugerido por Rice (1976), destacando a importância das suas ideias. Um modelo do processo geral de MA para seleção de algoritmos também é apresentado. Em seguida, foi realizado um levantamento bibliográfico sobre as principais técnicas para geração de conjuntos de dados aplicadas ao MA. Conceitos abordados, tais como: geração de conjuntos de dados, caracterização de dados, MA para seleção de algoritmos, medidas de avaliação, formas de recomendação, são importantes para o entendimento da aplicação dos mesmos ao longo desta de tese. O capítulo seguinte trata da revisão da literatura em OCC.

3 CLASSIFICAÇÃO DE UMA CLASSE

Este capítulo faz uma revisão da literatura dos principais tópicos associados à técnica OCC que serão abordados ao longo desta tese, bem como os métodos relacionados.

A classificação de uma classe OCC (TAX, 2001; KHAN; MADDEN, 2014; PERERA; OZA; PATEL, 2021) é uma área específica do AM. Tem como objetivo distinguir uma determinada classe única de um conjunto mais amplo de classes. Preocupa-se em aprender um classificador quando todos os exemplos rotulados disponíveis pertencem a uma única classe. Normalmente, a classe conhecida por meio dos dados de treinamento é chamada de classe positiva (ou classe alvo), enquanto as demais amostras são negativas (ou classe desconhecida). Esse problema também pode ser abordado usando uma mistura de dados positivos e não rotulados (LIU et al., 2003; WANG et al., 2019; BEKKER; DAVIS, 2020).

O objetivo da classificação binária é aprender um modelo que seja capaz de distinguir entre exemplos positivos e negativos. Para fazer isso, um algoritmo tem acesso aos dados de treinamento. Na configuração mais tradicional, esses dados contêm exemplos positivos e negativos e são totalmente rotulados, ou seja, o valor da classe não está ausente em nenhum exemplo de treinamento. Este é um dos problemas mais amplamente estudados em AM. Aprender com dados positivos e não rotulados ou PU Learning é uma variante dessa configuração clássica, em que os dados disponíveis consistem em exemplos positivos e exemplos sem rótulos. A suposição é que cada exemplo não rotulado pode pertencer à classe positiva ou negativa (alvo ou desconhecida). PU Learning é considerado em OCC por não requerer exemplos de treinamento representativos de todos as classes.

Para ilustrar e esclarecer melhor o conceito de OCC, consideramos um problema específico, como a concessão de cartões de crédito a clientes. Neste exemplo, as organizações que fornecem cartões de crédito precisam avaliar os requerimentos de novos clientes ou o comportamento dos clientes existentes para aceitá-los ou rejeitá-los. Como a maioria dos clientes paga suas contas e poucas pessoas ficam inadimplentes, não temos uma porção aceitável de inadimplentes. Outros exemplos semelhantes podem ser citados para explicar o uso e a importância da OCC.

Exemplos de treinamento para todas as classes são necessários no processo tradicional de classificação de duas ou múltiplas classes. No entanto, em problemas do mundo real, pode ser difícil obter exemplos de uma ou mais classes (HEMPSTALK; FRANK; WITTEN, 2008). Por exemplo, em problemas de autenticação de sujeito biométrico (MAUCERI et al., 2017) é relativamente fácil coletar alguns dados da classe do "sujeito-alvo", no entanto, é menos simples coletar dados que são representativos da classe "sujeito não-alvo". Outras aplicações relacionadas são: detecção de fraude (PHUA et al., 2010), diagnóstico médico (SCHLEGL et al., 2017), verificação de assinatura (GUERBAI; CHIBANI; HADJADJI, 2015). OCC é diferente da classificação convencional, uma vez que apenas exemplos de uma classe estão disponíveis para o aprendizado.

Em tarefas de classificação multiclasse, os algoritmos têm como objetivo classificar um objeto não conhecido dentre várias categorias pré-definidas. Em OCC, as instâncias de uma das classes (denominada classe positiva ou classe alvo) são bem caracterizadas nos dados de treinamento. Já os demais dados de treinamento que representam a segunda classe (denominada negativa ou não-alvo) existem em pequena quantidade, não existem ou não formam uma amostra representativa, estatisticamente, do conceito negativo. Assim, apenas um lado do limite de classificação pode ser descrito usando somente dados positivos. Isso torna o problema de classificação unária, OCC mais difícil que problema de classificação convencional de duas ou multiclasses. Problemas encontrados no processo de classificação tradicional como a estimação do erro de classificação, a complexidade da solução, a generalização do método de classificação, também aparecem em OCC.

Classificadores convencionais assumem que as classes de dados possuem um certo grau de equilíbrio e não funcionam bem quando qualquer classe é fortemente sub-amostrada ou ausente por completo. Uma dificuldade que pode ser encontrada em OCC é o fato de decidir quais atributos devem ser usados para encontrar a melhor separação dos exemplos de classe positivas e não positivas (KHAN; MADDEN, 2010).

Pesquisadores têm usado diferentes termos para designar conceitos semelhantes ao de OCC (KHAN; MADDEN, 2010), como Detecção de *Outlier* (SWERSKY et al., 2016; YOU; ROBINSON; VIDAL, 2017; CHEN et al., 2017) ou Detecção de Novidades (PIDHORSKYI et al., 2018;

ABATI et al., 2019; PERERA; NALLAPATI; XIANG, 2019), Detecção de Anomalias (SCHLEGL et al., 2017; LAI; ZOU; LERMAN, 2019), por exemplo. Isso ocorre devido ao uso do conceito original de OCC em diferentes aplicações.

3.1 ESTRATÉGIAS DE APRENDIZAGEM EM CLASSIFICAÇÃO DE UMA CLASSE

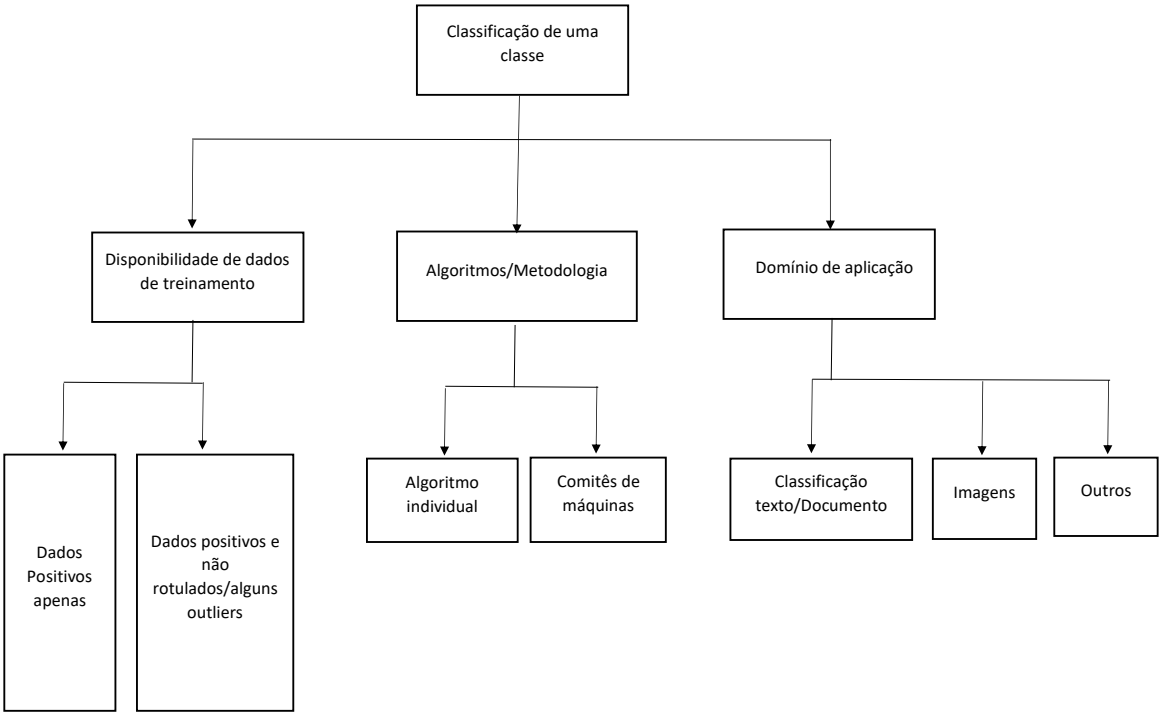
Nos últimos anos, OCC vem recebendo um crescente interesse por parte dos pesquisadores em AM. Como já mencionado, OCC é referenciada na literatura por diversos termos, mas também tem crescido o interesse relacionado a diversos domínios de aplicação. Além disso, atualmente, é encontrada na literatura uma série de métodos de aprendizagem. Sendo assim, ao longo da última década, algumas taxonomias do estudo de OCC (KHAN; MADDEN, 2014; PERERA; OZA; PATEL, 2021; VILLA-PEREZ et al., 2021) foram organizadas.

Em Khan e Madden (2014) é apresentada uma categorização para o estudo de problemas de OCC para propósito geral. As categorias propostas não são mutuamente exclusivas, logo pode ser que haja alguma sobreposição entre elas em pesquisas realizadas em cada uma das categorias. Khan e Madden (2014) dividem o estudo em OCC em três grandes categorias: (1) disponibilidade de dados de treinamento; (2) metodologia usada e (3) domínio de aplicação. Considerando os trabalhos de pesquisa realizados na comunidade de Visão Computacional, Perera, Oza e Patel (2021) apresentam uma categorização para o estudo de problemas de OCC baseados em imagens. As principais categorias identificadas em Perera, Oza e Patel (2021) são: dados, *features* e algoritmos de classificação. Já em Villa-Perez et al. (2021) é realizada uma classificação direcionada para os trabalhos relatados que usam algoritmos semi-supervisionados para detecção de anomalias. Como a categorização de Villa-Perez et al. (2021) é focada nos algoritmos usados nas estratégias de aprendizagem de OCC para detecção de anomalias, eles concentraram-se nesse tipo de abordagem, realizando um estudo empírico sobre os métodos de detecção de anomalias existentes na literatura.

Organizamos os trabalhos existentes em OCC e pesquisados nesta tese com o intuito de apresentar uma visão geral e atualizada do uso dessa abordagem. Para tanto, propomos uma taxonomia que busca detalhar um pouco mais o uso de OCC na literatura. A Figura 2 ilustra a categorização proposta nesta tese para o estudo de OCC. Nas subseções seguintes, são

apresentadas cada uma das três grandes categorias de OCC e trabalhos relacionados.

Figura 2 – Categorização para o estudo de Técnicas de *One-Class Classification*(OCC)



Fonte: Produzido pela autora (2022)

3.1.1 Disponibilidade de Dados de Treinamento

Em OCC, a questão da disponibilidade de dados de treinamento é fundamental, independentemente da metodologia utilizada. A ideia principal por trás dessa estratégia é diferenciar os dados positivos dos dados negativos construindo uma fronteira de decisão em torno dos dados positivos. Pesquisas relatam que OCC (KHAN; MADDEN, 2014; PERERA; OZA; PATEL, 2021; VILLA-PEREZ et al., 2021) tem sido estudada amplamente no que diz respeito a dados de treinamento em três aspectos:

a) Aprendizagem apenas com dados positivos. Em Yu e Han (2002) é proposto um *framework* para classificação de páginas web que automatiza a coleta de exemplos negativos para

treinamento baseado em aprendizagem de dados positivos e usando uma SVM. A OCC apenas com dados positivos busca diferenciar a classe positiva de todos os demais exemplos possíveis no espaço padrão. Isso pode ser feito construindo uma hiperesfera em torno dos dados de classe positiva que abrange quase todos os pontos no conjunto de dados com o raio mínimo, como pode ser visto em Tax e Duin (1999a).

b) Aprendizagem com exemplos positivos e um número de exemplos negativos artificiais. Exemplos negativos artificiais são gerados para viabilizar a técnica proposta em Li (2007), Hempstalk, Frank e Witten (2008) para detectar casos inesperados no conjunto de testes em classificação de documentos.

c) Aprendizagem com dados positivos e dados não rotulados. Aprendizagem com dados positivos e dados não rotulados é uma estratégia de classificação de uma classe conhecida como PU Learning (LIU et al., 2002; XIA et al., 2013; PLESSIS; NIU; SUGIYAMA, 2015; XU et al., 2017). PU Learning é muito aplicada no campo de classificação de texto (NAGY.; FARKAS.; CSIRIK., 2011). Em Liu et al. (2002) é utilizada uma técnica de classificação parcialmente supervisionada para identificar documentos que sejam de uma determinada classe. PU Learning foi adaptada em Xia et al. (2013) para seleção e ponderação de instâncias para classificação de sentimentos entre domínios. Nessa abordagem, amostras que são mais relevantes para o domínio alvo são selecionadas para dados de treinamento.

Segundo Liu et al. (2002), PU Learning recebe como entrada um conjunto $\mathcal{P}$ de exemplos da classe positiva (classe de interesse) e um conjunto $\mathcal{U}$ de exemplos não rotulados (ou seja, de classe desconhecida). O conjunto $\mathcal{U}$ pode conter tanto exemplos pertencentes a classe positiva como exemplos de outras classes. Um classificador é construído para selecionar as instâncias desconhecidas com maior probabilidade de pertencerem à classe positiva. PU Learning é uma das estratégias de OCC investigadas em nossa proposta.

A presente tese está relacionada a categoria (c), uma vez que os conjuntos de dados reais representam os exemplos da classe positiva (classe de interesse) e os dados sintéticos representam os dados não rotulados (classe desconhecida).

3.1.2 Metodologias/Algoritmos Usados

Na literatura de OCC, os algoritmos desenvolvidos podem ser divididos em duas grandes categorias: algoritmos aplicados individuais ou como comitês de máquinas.

3.1.2.1 Algoritmo Individual

Estratégias de OCC que empregam métodos únicos são muito estudadas na literatura. Alguns desses métodos são baseados em SVM. Esses métodos são chamados de *One Class Support Vector Machine* (OCSVM). Um representante clássico é o método conhecido como *Support Vector Data Description* (SVDD) (TAX; DUIN, 1999b). Diversas abordagens que aprimoram o SVDD foram publicadas (TAX; DUIN, 1999a; SCHôLKOPF et al., 2000; TAX; DUIN, 2001b; CAMPBELL; BENNETT, 2001). Em Tax e Duin (2001a) é proposto um método sofisticado que usa *outliers* gerados artificialmente para otimizar os parâmetros da OCSVM para equilibrar entre ajuste excessivo e ajuste insuficiente.

Em Ridder, Tax e Duin (1998) é realizada uma comparação experimental entre vários algoritmos de OCC. Nesses experimentos, uma rede neural simples foi treinada para filtrar documentos em situações em que apenas dados positivos estão disponíveis. Abordagens OCC com RF podem ser encontradas em Désir et al. (2012b), Désir et al. (2012a). Uma solução baseada em aprendizagem profunda para o problema da aprendizagem de características na OCC é proposta em Perera (2018). Em Désir et al. (2013) um estudo do comportamento do *One Class Random Forest* (OCRF) é apresentado, incluindo experimentos em vários conjuntos de dados públicos disponíveis no repositório UCI e, além disso é realizada uma comparação com algoritmos de classificação de uma classe de referência.

Outra abordagem é o método NN-d (TAX, 2001) para classificação de unária (TAX, 2001). Baseia-se no classificador de vizinho mais próximo, NN (DUDA; HART, 1973), um método de classificação no qual suas decisões são baseadas na distância entre a instância e a instância mais próxima no espaço de características. Em Khan e Madden (2014) é apresentada uma pesquisa com várias abordagens sobre OCC e uma análise sobre o emprego de diversos métodos de aprendizagem.

3.1.2.2 Comitês de Máquinas

Na literatura recente, trabalhos que aplicam comitês de máquinas para OCC são encontrados (KRAWCZYK; WOZNIAK; CYGANEK, 2014; KRAWCZYK et al., 2018; CAMIÑA et al., 2019). Em Krawczyk, Wozniak e Cyganek (2014) um *ensemble* de máquinas de vetores de suporte de classe única é proposto para decomposição de problemas multiclasse aplicado a conjuntos de dados de repositórios, como UCI, por exemplo. Uma nova abordagem para combinar classificadores de classe única para resolver problemas multiclasse baseados na seleção dinâmica de *ensembles* é proposta em Krawczyk et al. (2018).

A dependência de computadores pessoais tem exigido o desenvolvimento de mecanismos de segurança para proteger as informações armazenadas nesses dispositivos. Existem diferentes abordagens para traçar o perfil do comportamento do usuário para proteger as informações de um ataque disfarçado; uma dessas abordagens é baseada em padrões de acesso a arquivos do usuário. Por exemplo, considere o problema da detecção de mascarador móvel (MAZHELIS, 2007) , no qual o modelo deve determinar se a atividade corresponde ao usuário legítimo ou a um intruso. Para realizá-lo, o modelo tenta identificar qualquer tentativa de se passar por um usuário legítimo para obter acesso não autorizado a dados ou serviços sensíveis para aquele usuário, comparando cada padrão de acesso em relação aos padrões que descrevem os acessos do usuário legítimo. Um *Bagging-RandomMiner* é empregado para classificação de intrusos baseado em acesso de arquivos (CAMIÑA et al., 2019).

Em Tsai e Lin (2021) cada conjunto de dados desequilibrado de duas classes é tratado como um problema de detecção de anomalia, que contém um número maior de dados na classe majoritária, ou seja, classe positiva e um número muito pequeno de dados na classe minoritária. Os objetivos de pesquisa são compreender o desempenho dos classificadores de OCC e examinar o nível de melhoria de desempenho quando a seleção de características é considerada para o pré-processamento dos dados de treinamento. *Ensemble* de classificadores de uma classe são construídos para este fim.

3.1.3 Domínios de Aplicação

Na literatura de OCC, os domínios de aplicação podem ser divididos em: (1) classificação de texto/documentos; (2) classificação de imagem e (3) outros domínios de aplicação.

3.1.3.1 Domínios de Classificação de Texto/Documentos

Em tarefas de classificação de texto/documentos os algoritmos de OCC são aplicados, principalmente, usando dados não rotulados e classificação semi-supervisionada, pois técnicas tradicionais de classificação de texto requerem uma distribuição apropriada de exemplos positivos e negativos para construir um classificador. A técnica de PU Learning é muito aplicada no campo de classificação de texto (NAGY.; FARKAS.; CSIRIK., 2011). Em Liu et al. (2002) é utilizada classificação parcialmente supervisionada para identificar documentos de uma classe alvo. Em Liu et al. (2003) é apresentado um estudo do problema de construção de classificadores de texto usando exemplos positivos e não rotulados. Uma abordagem para resolver o problema com base em uma formulação tendenciosa de SVM é apresentada. Já em Li e Liu (2003) uma combinação do método *Rocchio* com o método SVM é realizada para construção de classificadores aplicados à classificação de texto quando apenas dados positivos são disponíveis.

Dados positivos e não rotulados em abordagens OCC para domínios de classificação texto são comuns, porém outras estratégias também são adotadas, como em Li (2007) em que é proposta uma solução para o problema da necessidade de existência de dados negativos para construção do classificador no domínio de classificação de texto. A técnica proposta gera documentos negativos artificiais que são usados, juntamente com os documentos positivos, para construir um classificador *Naive Bayes* (NB) para identificar instâncias inesperadas no conjunto de teste.

3.1.3.2 Classificação de Imagem

O uso de técnicas de OCC para classificação de imagem tem atraído cada vez mais atenção nos últimos tempos (CLIFTON et al., 2011; GUO et al., 2011; GAO et al., 2020; VILLA-PEREZ et al., 2021).

Em ambientes clínicos, por exemplo, muitos conjuntos de dados de imagens médicas sofrem com o problema de desequilíbrio que dificulta a detecção de *outliers* (eventos raros de cuidados de saúde), já que a maioria dos métodos de classificação pressupõe uma ocorrência igual de classes. Dessa forma, identificar *outliers* em conjuntos de dados desequilibrados se tornou uma questão crucial. Para ajudar a enfrentar esse desafio, a OCC, que se concentra no aprendizado de um modelo usando amostras de apenas uma determinada classe, tem recebido cada vez mais destaque. Um novo método é proposto em Gao et al. (2020) para permitir que modelos de aprendizado profundo aprendam de forma otimizada características de imagem relevantes de classe única. Realizam também uma investigação sobre os efeitos de determinadas perturbações aplicadas às imagens para capturar a complexidade da imagem e melhorar o aprendizado baseado em *features*. Uma máquina de vetor suporte de uma classe é empregada em Clifton et al. (2011) para identificar a deterioração dos pacientes nos dados de sinais vitais. Em Gardner et al. (2006a) foi usada para detectar convulsões na série de eletroencefalograma humano (EEG). Já uma máquina de vetor suporte de uma classe ponderada é projetada para detectar tumores em imagens de tomografia computadorizada do cérebro em Guo et al. (2011). Em Villa-Perez et al. (2021) é apresentada uma pesquisa envolvendo problemas OCC baseado em imagens.

3.1.3.3 Outros Domínios de Aplicação

Nas aplicações em domínios diferentes da classificação de texto/documentos, podem ser encontrados diversos trabalhos em diferentes aplicações, como o uso de OCSVM em reconhecimento de dígitos escritos à mão (TAX; DUIN, 2001b) e em classificação de páginas web (YU; HAN, 2002), reconhecimento facial e de gesto (ZENG et al., 2006), em análise médica (GARDNER et al., 2006b; WEI et al., 2018), uso de Rede Neural em detecção de anomalias (PERDISCI; GU, 2006; WEI et al., 2018; SABOKROU et al., 2018; VILLA-PEREZ et al., 2021).

Pesquisas mais aprofundadas sobre OCC, aprendizagem de dados positivos e não rotulados, dentre outros temas pertinentes podem ser encontradas em Khan e Madden (2010), Khan e Madden (2014), Krawczyk et al. (2018), Bellinger, Sharma e Japkowicz (2018), Dai et al. (2019), Bekker e Davis (2020), Perera, Oza e Patel (2021) e referências ali contidas.

3.2 CONSIDERAÇÕES FINAIS

Este capítulo discorreu sobre os principais tópicos relacionados à OCC abordados nesta tese. Foi apresentada a fundamentação da OCC e suas aplicações. Apresentamos uma breve introdução ao tema OCC fazendo um paralelo entre suas características e as características de problemas de classificação multiclasse. Logo após, apresentamos aplicações que fazem uso das metodologias de OCC e as relacionamos às diversas categorias existentes em OCC.

O próximo capítulo detalha o uso de OCC para seleção de conjuntos de dados em MA. Para validar eficiência da nossa proposta, argumentações teóricas são apresentadas. Uma descrição esquemática da solução proposta é apresentada, juntamente com os métodos adotados na seleção de conjuntos de dados para MA. Outrossim, apresenta as construções dos meta-exemplos e de cada módulo da solução proposta.

4 CLASSIFICAÇÃO DE UMA CLASSE PARA SELEÇÃO CONJUNTOS DE DADOS SINTÉTICOS

Neste capítulo, a abordagem proposta de usar OCC para selecionar conjuntos de dados em contexto MA é apresenta, bem como os métodos de OCC adotados na seleção dos conjuntos de dados. Além disso, são apresentadas as configurações dos experimentos em dois estudos de caso. Cada estudo de caso tem seu próprio conjunto de medidas de caracterização e são adotadas técnicas de geração de conjuntos de dados sintéticos diferentes em cada um deles. As tarefas de MA também são diferentes para cada estudo de caso. Sendo assim, avaliamos a qualidade dos meta-aprendizes treinados com os meta-exemplos selecionados usando técnicas de OCC em duas configurações diferentes de experimentos. Nas seções 4.4.1 e 4.4.2, descreveremos o conjunto de meta-exemplos adotados nos experimentos de cada estudo de caso (1 e 2). Discutiremos também os módulos de *Manipulação de dados*, *Caracterização de dados* e *Avaliação de algoritmos*, presentes no processo experimental, e que são específicos para cada estudo de caso. Já os módulos *Meta-Aprendizagem* é comum aos experimentos. Na seção 4.5, apresentaremos os algoritmos de MA que foram adotados em ambos os estudos de caso. Em seguida, a seção 4.6, descreve a metodologia dos experimentos para avaliar a proposta de OCC para seleção de conjuntos de dados relevantes, ou seja, aqueles exemplos que não favorecem ou até deterioram a acurácia do MA. Algumas questões motivam esta pesquisa e serão descritas na próxima Seção.

4.1 ABORDAGEM PROPOSTA

É possível usar a geração de dados para aumentar o número de conjuntos de dados disponíveis para MA, gerando um impacto potencialmente positivo no desempenho no meta-nível. Embora o processo de geração de conjuntos de dados produza meta-exemplos verdadeiramente relevantes, ele também pode resultar em muitos exemplos redundantes ou anômalos (SOARES, 2009). Além disso, conforme discutido na Seção 2.1, a rotulagem de exemplos em MA é realizada por meio de um processo experimental que pode ser caro computacionalmente, uma vez que a avaliação empírica de algoritmos candidatos é realizada. Portanto, podemos citar duas motivações importantes para nossa proposta de seleção de conjuntos de dados relevantes para a geração de meta-exemplos: (i) Melhorar a qualidade dos exemplos de treinamento

em MA; e (ii) Diminuição do custo computacional de experimentos na avaliação empírica de algoritmos candidatos no processo de rotulagem de meta-exemplos, ou seja, evitar rotulagem de meta-exemplos que não serão úteis ao processo de MA. Além disso, até onde sabemos, existem poucos trabalhos relacionados à seleção de conjuntos de dados na literatura de MA. Frequentemente, o foco está na geração ou seleção de meta-características. Sendo assim, nossa proposta tem uma motivação extra que é contribuir com uma nova técnica para selecionar conjuntos de dados nesse contexto. A OCC como técnica de seleção de conjuntos de dados em MA é uma abordagem inédita na literatura.

Nesse sentido, nesta tese, concentramos nosso estudo em como melhorar a qualidade dos conjuntos de exemplos em MA, combinando a manipulação de dados para geração de conjuntos de dados sintéticos e técnicas de OCC para seleção desses conjuntos de dados de forma que os meta-exemplos gerados a partir deles seja relevantes para o processo de MA. Deve-se considerar que: a seleção de meta-exemplos sintéticos é tratada como um problema de OCC, a classe positiva é formada de meta-exemplos produzidos a partir de conjuntos de dados reais e que os meta-exemplos produzidos a partir de conjuntos de dados sintéticos constituem a classe desconhecida (ou seja, positivo/relevante ou irrelevante/redundante). Pode haver uma diferença entre as distribuições de meta-características para conjuntos de dados reais e sintéticos, já que muitos dos conjuntos de dados sintéticos não são representativos dos reais. Selecionar apenas os meta-exemplos sintéticos não rotulados mais relevantes pode ser uma boa estratégia para gerar exemplos de treinamento de MA. Nossa hipótese é que as técnicas de OCC lidam facilmente com esse tipo de problema.

Na literatura, alguns trabalhos (PRUDÊNCIO; SOARES; LUDERMIR, 2010; PRUDÊNCIO; SOARES; LUDERMIR, 2011; SOUSA; PRUDÊNCIO; LUDERMIR, 2016) foram realizados para selecionar meta-exemplos tratando este problema como uma tarefa de aprendizagem ativa (COHN; ATLAS; LADNER, 1994). Técnicas de amostragem de incerteza (SETTLES, 2009; PEREIRA-SANTOS; PRUDÊNCIO; CARVALHO, 2019) foram adotadas para selecionar iterativamente meta-exemplos para um meta-aprendiz KNN. Esta abordagem seleciona meta-exemplos para os quais o meta-aprendiz atual é mais incerto em suas previsões. Embora os trabalhos anteriores tenham alcançado resultados promissores, eles têm algumas limitações identificadas. Em primeiro lugar, técnicas de amostragem de incerteza foram adotadas em ambos os estudos, o que tende a dar grande importância a valores discrepantes (ou seja, exemplos que são incertos, mas não tão

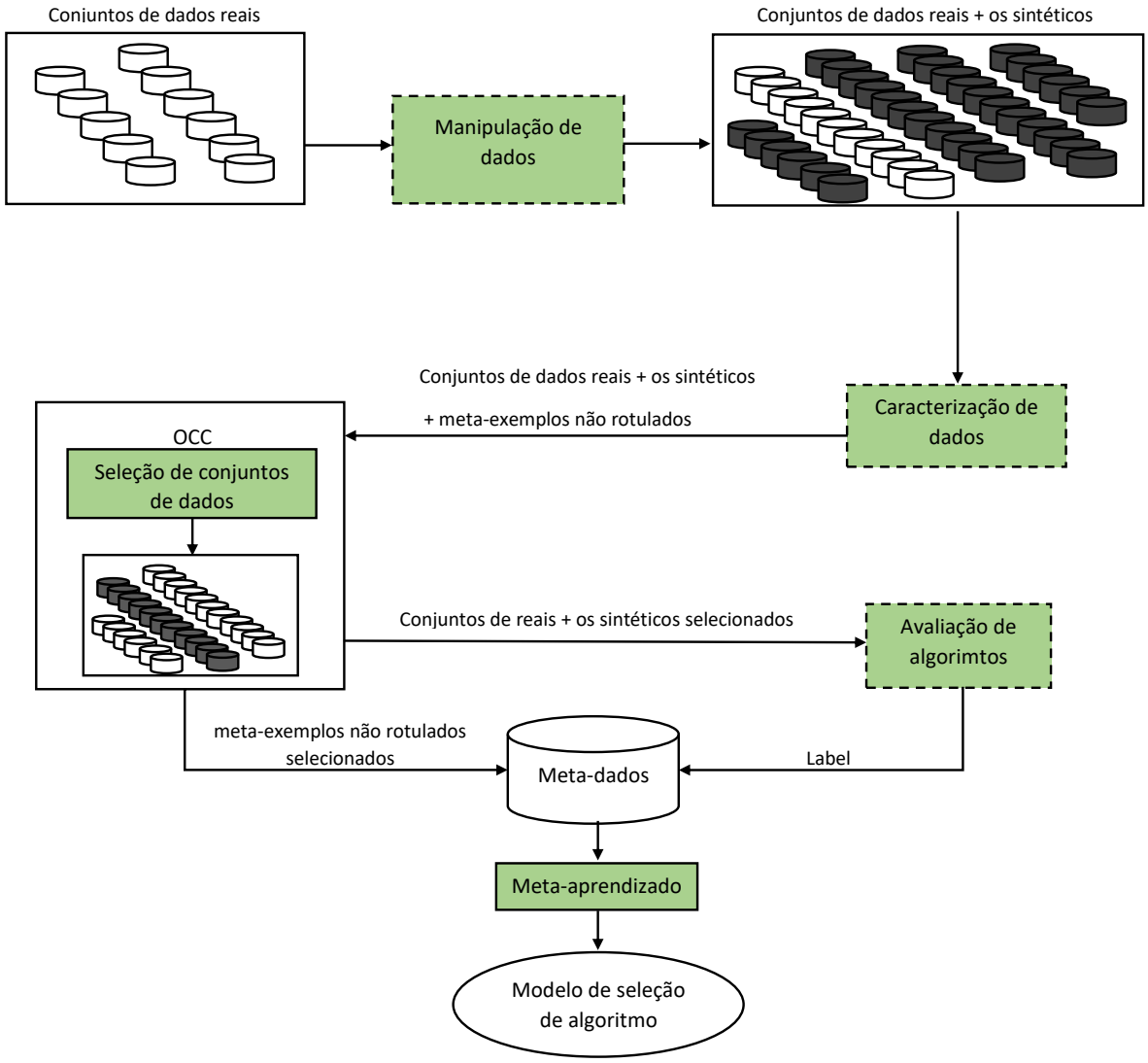
úteis para estar em um conjunto de treinamento). Além disso, a aprendizagem ativa é uma solução iterativa em que o meta-aprendiz deve ser atualizado assim que os meta-exemplos são selecionados. A diferença em nossa abordagem é o fato de que usamos técnicas de OCC na seleção de conjuntos de dados relevantes para o MA e o meta-aprendiz é construído de uma única maneira (ou seja, a abordagem é menos cara e potencialmente mais estável). Além disso, avaliamos nossa abordagem em problemas de classificação de diferentes domínios. Na literatura, até onde sabemos, não existem trabalhos semelhantes que investigam OCC na seleção de conjuntos de dados no contexto de MA.

A Figura 3 mostra o esboço da proposta desenvolvida no presente trabalho. Inicialmente, um número de conjuntos de dados reais é fornecido como entrada para um módulo de ***Manipulação de dados***. Com base nos conjuntos de dados originais, o módulo gera conjuntos de dados sintéticos. Em seguida, ambos os conjuntos de dados (reais e sintéticos) são apresentados como entrada para o módulo ***Caracterização de dados*** a fim de calcular as meta-características de cada conjunto de dados, produzindo uma ampla coleção de meta-exemplos não rotulados. Em seguida, o módulo ***OCC*** é utilizado para selecionar os meta-exemplos mais relevantes no conjunto de meta-exemplos sintéticos (desconhecido). No passo seguinte, os meta-exemplos selecionados são rotulados (ou seja, o rótulo da meta-classe é atribuído) após a avaliação dos algoritmos candidatos em um procedimento de avaliação empírica (módulo ***Avaliação de Algoritmo***) produzindo, consequentemente, a meta-base de exemplos (*Meta-Dados*). Na última etapa, o módulo ***Meta-aprendizado*** recebe como entrada os meta-dados e fornece como saída um modelo de seleção de algoritmo.

Nesse trabalho, adotamos dois diferentes métodos para geração de conjuntos de dados artificiais, os quais serão detalhados mais adiante na descrição dos estudos de caso desenvolvidos. Os meta-exemplos gerados a partir dos conjuntos de cados artificiais também serão detalhados em cada estudo de caso.

O módulo OCC é responsável por selecionar os exemplos que serão rotulados no módulo Avaliação de Algoritmos, como já mencionado. No trabalho desenvolvido, realizamos dois estudos de caso. Em ambos os estudos de caso desenvolvidos na presente pesquisa, dois métodos

Figura 3 – Uma descrição esquemática da solução proposta.



Fonte: Produzido pela autora (2022)

distintos de OCC foram considerados, *PU learning* proposto em (LIU et al., 2003) e NN-d (TAX, 2001), que são detalhados nas próximas Subseções. Em ambos os estudos de casos, dois conjuntos de instâncias estão disponíveis: (1) um conjunto $\mathcal{P}$ de exemplos na classe positiva (ou seja, rotulados com a classe positiva alvo); e (2) um conjunto $\mathcal{U}$ de exemplos com a classe desconhecida (ou seja, exemplos não rotulados), como de costume em OCC. Em nossa proposta, a classe positiva representa os conjuntos de dados reais e as instâncias desconhecidas são os conjuntos de dados sintéticos $\mathcal{U}$. O conjunto $\mathcal{U}$ pode ter exemplos pertencentes à classe positiva e à negativa (desconhecida). Um classificador é construído para selecionar instâncias na classe desconhecida com maior probabilidade de pertencerem à classe positiva. A apren-

dizagem em OCC é, portanto, realizada para classificar os conjuntos de dados sintéticos que estão mais relacionados aos reais.

Os métodos de OCC adotados nesta tese, com o objetivo de testar a viabilidade da presente proposta, são implementados em ambos os estudos de caso, como já mencionado anteriormente. Por uma questão didática, o detalhamento destes métodos de OCC escolhidos em nossos experimentos é realizado antes do detalhamento dos meta-exemplos de cada estudo de caso.

4.2 *PU LEARNING*

Em nosso trabalho, consideramos o PU Learning (LIU et al., 2002) para OCC, que é composto por duas etapas. A primeira etapa é adotada para criar um conjunto de treinamento inicial para a construção do classificador. A segunda etapa é o próprio processo de aprendizagem, no qual exemplos são incorporados ao conjunto de treinamento. A escolha do método a ser utilizado por etapa pode ser feita livremente. Na literatura, são propostas algumas combinações e variações de métodos de PU Learning(BEKKER; DAVIS, 2020).

O **Algoritmo 1** ilustra o primeiro passo do método. Inicialmente, um pequeno conjunto $\mathcal{S}$ de exemplos de classe positiva $\mathcal{P}$ (nomeado como espiões) é misturado com os dados não rotulados $\mathcal{U}$. Como sugerido em (LIU et al., 2002), usamos 10% de $\mathcal{P}$, selecionados aleatoriamente, como espiões. Em nosso trabalho, essa porcentagem de dados usados como espiões não pode ser muito grande devido a quantidade de dados reais adotada em nossa pesquisa. Sendo assim, após uma pequena análise entre 5%, 10% e 15%, decidimos por 10% dos dados reais como espiões. Os conjuntos de dados $\mathcal{P} - \mathcal{S}$ e $\mathcal{U} \cup \mathcal{S}$ são temporariamente rotulados como o conjunto positivo e negativo, respectivamente, e o classificador NB é construído. Uma instância em $\mathcal{U}$ é finalmente assumida como negativa se sua probabilidade de classe positiva não exceder um limite predefinido b. Finalmente, ele retorna o subconjunto N_r das instâncias negativas mais prováveis no conjunto desconhecido. As demais instâncias são atribuídas a U_r. Esta é uma classificação inicial que será refinada na segunda etapa do PU Learning, que tem como entrada tanto N_r quanto U_r.

Algoritmo 1: PU Learning - Inicialização do Conjunto de Treinamento

$N_r =$ NULL;
$U_r =$ NULL;
$\mathcal{S} =$ amostra($\mathcal{P}$,10%);
atribua a cada exemplo em $\mathcal{P} - \mathcal{S}$ o rótulo da classe positiva;
atribua a cada exemplo em $\mathcal{U} \cup \mathcal{S}$ o rótulo da classe negativa;
construa um modelo NB g usando $\mathcal{P} - \mathcal{S}$ e $\mathcal{U} \cup \mathcal{S}$ como conjunto de treinamento;
classifique cada instância em $\mathcal{U} \cup \mathcal{S}$ usando g
determine b;
for *cada instância* $e \in \mathcal{U}$ **do**
    use o modelo g e compute $P_r(positiva|e)$
    if *probabilidade* $P_r(positiva|e) < b$ **then**
        $N_r = N_r \cup \{e\}$;
$U_r = \mathcal{U} - N_r$;
retorne N_r e U_r para o segundo passo.

Uma questão importante na primeira etapa é o limite de decisão b adotado para atribuir as instâncias negativas. Uma ideia direta é definir b como a probabilidade de classe positiva mínima observada entre as instâncias em $\mathcal{S}$. No entanto, também é esperada a presença de ruído no conjunto positivo, de forma que a probabilidade mínima não é confiável (LIU et al., 2002). Por exemplo, a probabilidade de classe de uma instância atípica em $\mathcal{S}$ pode ser muito próxima de 0 ou muito menor do que a maioria de todas as instâncias negativas. Isso também é verdade em nosso contexto, uma vez que nem todos os conjuntos de dados em repositórios como UCI ou OpenML, por exemplo, podem ser, sem dúvida, considerados relevantes. Em nosso trabalho, definiremos um nível de ruído $Nl\%$ dos exemplos, como parâmetro do método. Em seguida, ordenamos os exemplos em $\mathcal{S}$ por suas probabilidades de classe positivas e definimos o limite b de forma que $Nl\%$ dos exemplos positivos tenham probabilidade de classe inferior a b. Em nossos experimentos, avaliamos o comportamento do método PU Learning com vários níveis de ruído.

Uma SVM é construída no segundo passo do método (**Algoritmo 2**). $\mathcal{P}$ e N_r são usados como conjuntos de treinamento para aprender um classificador SVM. Para minimizar os efeitos de dados desequilibrados no treinamento, usamos a sobreamostragem, na qual instâncias de classes minoritárias são replicadas aleatoriamente. Em nossos experimentos, realizamos uma análise do comportamento do métoodo PU Learning ao serem adotados níveis de ruídos extremos, como será visto nas configurações dos experimentos. Assim, a necessidade do balanceamento dos dados de treinamento. O SVM é aplicado para classificar as instâncias em

U_r. As instâncias classificadas como positivas (U_p), ou seja, aquelas instâncias para as quais a SVM foi capaz de predizer como positivas, são consideradas como o conjunto de exemplos relevantes e os exemplos negativos (V_n) são desprezados.

Algoritmo 2: PU Learning - Classificação de Instâncias no Conjunto Desconhecido

$U_p = \text{NULL}$;
Oversampling($\mathcal{P}$,N_r);
usa $\mathcal{P}$ and N_r para treinar um classificador SVM;
classifique U_r usando o modelo SVM;
seja $V_n \subset U_r$ o conjunto de instâncias classificadas como negativas pelo modelo atual;
seja $U_p \subset U_r$ o conjunto de instâncias classificadas como positivas pelo modelo atual;
retorne U_p

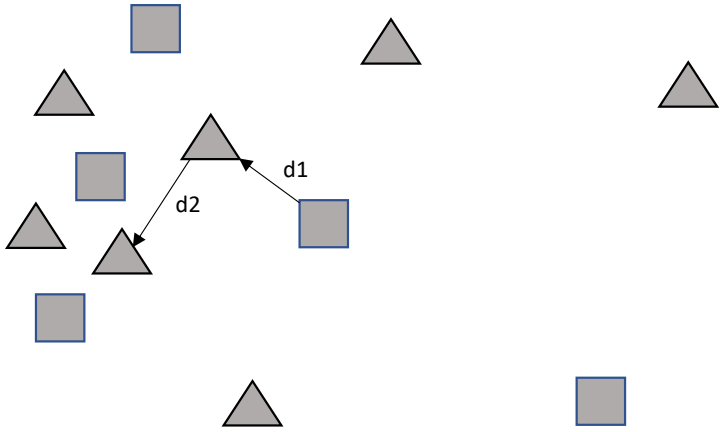
4.3 *ONE-CLASS NEAREST NEIGHBOR*

O segundo método OCC adotado é NN-d (TAX, 2001). Este método é baseado no classificador NN (DUDA; HART, 1973). Em nosso contexto constituído por meta-exemplos produzidos a partir de conjuntos de dados reais, na fase de treinamento, apenas o conjunto de instâncias de classe positiva é armazenado na memória. Na fase de teste, uma instância desconhecida (meta-exemplo sintético) é definida como relevante por NN-d considerando o seguinte procedimento. Primeiramente, dado um exemplo $e \in \mathcal{U}$, NN-d encontra seu vizinho mais próximo na classe positiva $NN_{tr}(e) \in \mathcal{P}$. Depois, o vizinho positivo mais próximo de $NN_{tr}(e)$ é retornado, ou seja, $NN_{tr}(NN_{tr}(e))$. Por fim, o exemplo desconhecido e é considerado relevante se a distância até seu vizinho positivo mais próximo $dist(e, NN_{tr}(e))$ é menor que a distância entre $NN_{tr}(e)$ e seu próprio vizinho positivo mais próximo. Isso é definido na Equação (4.1), onde $dist$ é a distância euclidiana entre dois meta-exemplos:

$$\frac{dist(e, NN_{tr}(e))}{dist(NN_{tr}(e), NN_{tr}(NN_{tr}(e)))} \leq 1 \tag{4.1}$$

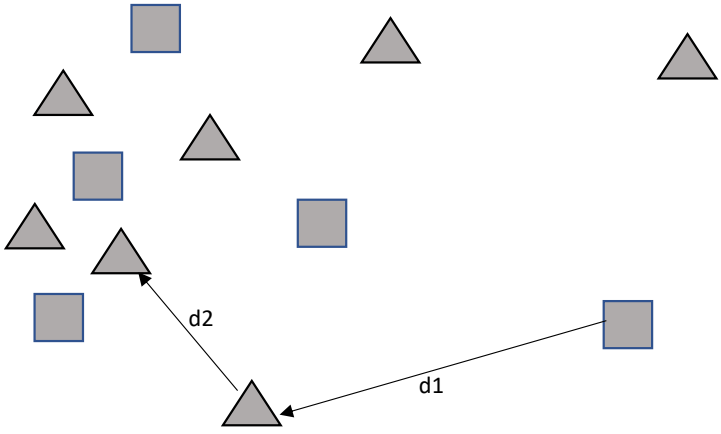
Caso contrário, e é classificado como irrelevante. Uma situação onde um meta-exemplo sintético é aceito por NN-d é vista em Figura 4, uma vez que $\frac{d1}{d2} \leq 1$. Já em Figura 5 ilustra uma situação onde um meta-exemplo sintético é rejeitado por NN-d.

Figura 4 – Ilustração de meta-exemplo sintético que é aceito por NN-d. As instâncias sintética e real são muito próximas uma da outra. Os triângulos representam os meta-exemplos reais e os quadrados os meta-exemplos desconhecidos.



Fonte: Produzido pela autora (2022)

Figura 5 – Ilustração de meta-exemplo sintético que é rejeitado por NN-d. As instâncias sintética e real são muito distantes uma da outra. Os triângulos representam os meta-exemplos reais e os quadrados os meta-exemplos desconhecidos.



Fonte: Produzido pela autora (2022)

A escolha dos métodos PU Learning e NN-d pode ser justicada em nosssos exeprimentos por serem métodos amplamente difundidos na literatura, como visto anteriormente, além de serem métodos de fácil implementação e reprodução. Os experimentos conduzidos, nesta tese, produziram dois conjuntos diferentes de meta-exemplos, como já mencionado. As próximas duas seções descrevem os processos de geração de cada um deles.

4.4 META-EXEMPLOS

4.4.1 Estudo de Caso 1

Neste primeiro estudo de caso, no nível base, adotamos a mesma tarefa de MA escolhida para avaliar uma abordagem de manipulação de dados proposta em (SOARES, 2009). Consiste em prever, a *priori*, se a poda de uma árvore de decisão melhorará (ou não) a qualidade do modelo, dado um determinado problema. O processo de rotulagem de cada meta-exemplo é baseado nos resultados dos experimentos no problema correspondente. Existem três rótulos de classe de meta-exemplos: (1) o vencedor é a árvore podada; (2) a árvore sem podada; ou (3) aquele que está empatado. Eles são, respectivamente, denotados por p, u ou t.

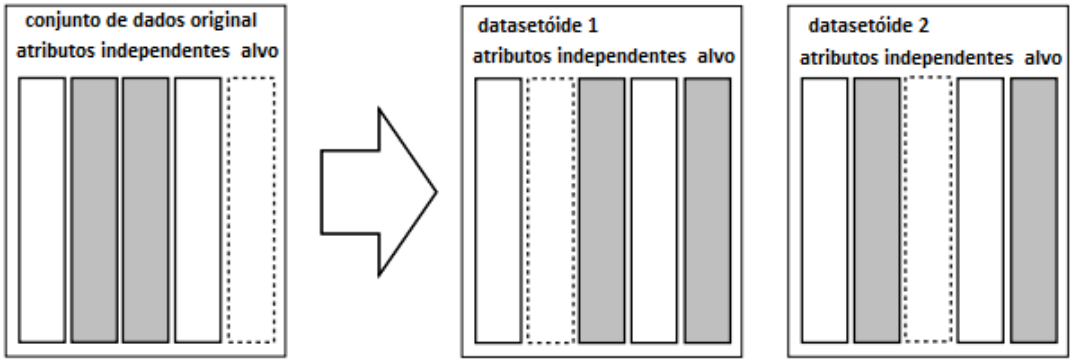
Os conjuntos de problemas originais usados neste trabalho são conjuntos de classificação coletados no repositório UCI (LICHMAN, 2013). De 64 problemas originais coletados nesses repositório, 983 novos conjuntos de dados manipulados foram produzidos. O processo de geração dos conjuntos de dados sintéticos é descrito a seguir.

4.4.1.1 Manipulação de Dados

Tendo em vista a necessidade de aumentar o número de exemplos de treinamento disponíveis para MA, adotamos o método de manipulação de dados proposto em (SOARES, 2009). Esta abordagem implica na manipulação de conjuntos de dados reais existentes. Diferentes operadores de manipulação podem ser adotados para produzir novos conjuntos de dados sintéticos. Em (SOARES, 2009), um operador simples é proposto. Um conjunto de dados sintético é gerado a partir de um determinado conjunto de dados reais trocando o atributo alvo por um atributo categórico independente. Dessa forma, o atributo alvo do conjunto de dados original

torna-se um atributo de entrada no conjunto de dados sintéticos e o atributo independente selecionado torna-se o atributo alvo. Para gerar conjuntos de dados sintéticos para classificação, o processo é repetido para cada atributo categórico do conjunto de dados reais, criando tantos conjuntos de dados novos quanto atributos categóricos existentes no conjunto de dados. Neste caso, o conjunto de dados deve conter pelo menos um atributo categórico. A Figura 6 ilustra o processo de geração de dois novos conjuntos de dados sintéticos de classificação a partir de um conjunto de dados com dois atributos simbólicos.

Figura 6 – Geração de dois conjuntos de dados sintéticos de classificação a partir de uma base de dados com dois atributos simbólicos.



Fonte: Adaptado de (SOARES, 2009)

4.4.1.2 Caracterização de Dados

Assim como em (SOARES, 2009), em nosso primeiro estudo de caso, adotamos duas medidas de caracterização para descrever os conjuntos de dados: a entropia da classe e a entropia média dos atributos (BRAZDIL; SOARES; COSTA, 2003). O algoritmo utilizado no nível base é a árvore de decisão, portanto, espera-se que as informações fornecidas por essas medidas sejam relevantes, uma vez que esse algoritmo utiliza o conceito de entropia. Certamente, o desempenho do MA poderia ser favorecido com a adoção de outras medidas existentes, ou seja, meta-características. Porém, com o objetivo de focar na combinação de técnicas de OCC e geração de conjuntos de dados sintéticos, optamos por utilizar medidas previamente adotadas. O procedimento de caracterização dos dados visa produzir um conjunto de meta-exemplos não rotulados, onde cada meta-exemplo corresponde a um conjunto de dados (original ou sintético)

considerado no estudo de caso. Os conjuntos de dados e seus respectivos meta-exemplos são as entradas do *Módulo OCC*.

4.4.1.3 Avaliação de Algoritmo

Conforme descrito na Figura 3, uma vez que um meta-exemplo é selecionado pelo módulo OCC, ele é rotulado para MA adotando o módulo de Avaliação de Algoritmo. Neste primeiro estudo de caso, cada meta-exemplo é rotulado com um dos três rótulos (p, u ou t) avaliando a árvore de decisão, conforme descrito anteriormente. A medida de desempenho usada no nível base foi a precisão da classificação, estimada por meio de *10-fold cross-validation*. A classe de cada meta-exemplo é baseada nos resultados dos experimentos no problema correspondente. Para mais informações sobre as implementações, o leitor pode consultar Soares (2009) e as referências por ele citadas.

A tabela 1 exibe as distribuições de classe para os meta-dados, tanto os dados originais quanto os sintéticos. Como pode ser visto, as distribuições de classes são diferentes, o que indica que pode haver algumas diferenças subjacentes entre conjuntos de dados reais e sintéticos. No entanto, as diferenças não são tão grandes na distribuição de classes considerando os conjuntos de dados reais e sintéticos separadamente.

Tabela 1 – Distribuição de classes (%) dos meta-dados relacionados aos conjuntos de dados originais e sintéticos

Meta-dado	Nº de meta-exemplos	Árvore podada (p)	Árvore sem poda (u)	Empate (t)
Real	64	36	23	41
Sintético	983	37	10	53

Fonte: Produzido pela autora baseado em (SOARES, 2009)

A Seção 4.4.2 mostra os detalhes do estudo de caso 2, incluindo a técnica de manipulação de dados adotada, o conjunto de características extraídas e a tarefa de meta-aprendizado.

4.4.2 Estudo de Caso 2

Em nosso segundo estudo de caso, a tarefa de meta-aprendizado adotada foi selecionar o melhor algoritmo de classificação entre quatro candidatos: K-NN, *JRip* (JR), *Naive Bayes* (NB) e *Decisão árvore* (J48). Adotamos esses métodos porque eles têm critérios de classificação diferentes. Os conjuntos de dados reais adotados neste estudo de caso são conjuntos de dados de classificação coletados no repositório *OpenML* (VANSCHOREN et al., 2013).

4.4.2.1 Manipulação de Dados

Neste estudo de caso, um algoritmo evolucionário é adotado para evoluir conjuntos de dados reais a fim de obter conjuntos de dados sintéticos com os níveis desejados de complexidade (LORENA et al., 2018). Cada novo conjunto de dados sintético é criado evoluindo subconjuntos de instâncias de um conjunto de dados real existente. Esta proposta preserva algumas características dos conjuntos de dados originais, como o número de exemplos, atributos, classes e distribuição de classes. Em primeiro lugar, o conjunto de dados original, que será a entrada para o processo de manipulação, é dividido em exemplos de instâncias. Então, um algoritmo evolucionário manipula o conjunto de dados original até que o nível desejado de uma medida de complexidade seja alcançado. O nível de complexidade dado como entrada é adotado na função objetivo do algoritmo evolucionário.

O método recebe duas entradas: um conjunto de dados D e o valor desejado c_{target} de uma medida de complexidade C_M. Então o método evolucionário proposto por (HANSEN, 2006) retorna como saída um conjunto de dados manipulado D_{new} minimizando a seguinte função objetivo:

$$f(D_{new}) = (C_M(D_{new}) - c_{target})^2 \tag{4.2}$$

Seja Q o número de classes e seja P o número de atributos no conjunto de dados D. Inicialmente, os valores médios das características são calculados para cada classe em D, ou seja, o valor médio μ_{pq} de cada característica em cada classe, onde $p \in \{1, \ldots, P\}$ e $q \in \{1, \ldots, Q\}$.

O indivíduo a ser otimizado é composto pelo conjunto de valores médios de $P \times Q$. Novos indivíduos serão gerados pela aplicação de distúrbios (passo de mutação) nos valores médios das características. Os atributos não numéricos passam por uma transformação numérica. Dado um novo indivíduo com valores médios μ'_{pq}, podemos reconstruir as instâncias sintéticas que farão parte do novo conjunto de dados amostrando os valores das características de acordo com a nova média. Mais especificamente, as instâncias são geradas para cada classe, assumindo uma distribuição normal:

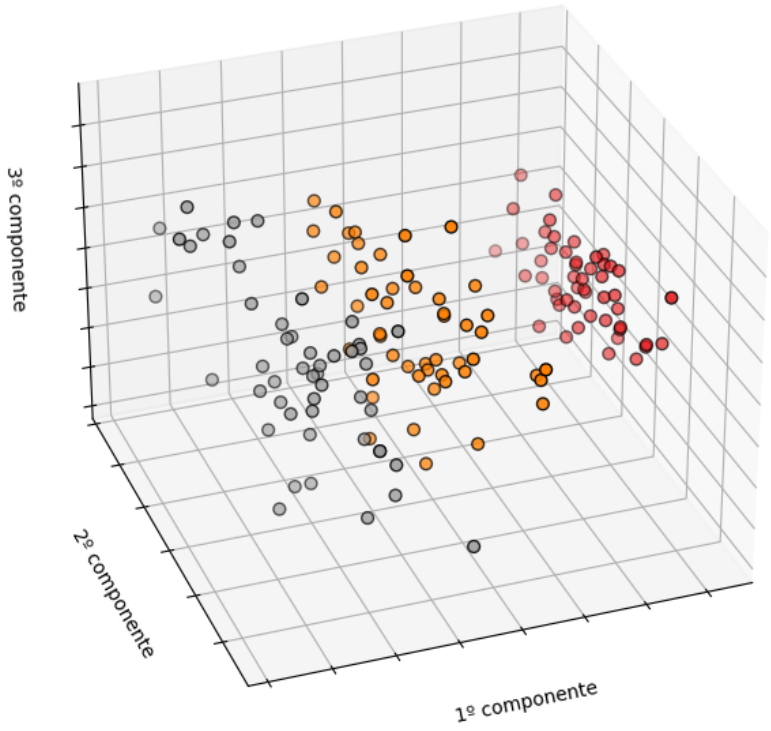
$$x_p \sim \mathcal{N}(\mu'_{pq}, \sigma^2_{pq}) \tag{4.3}$$

em que σ_{pq} é o desvio padrão original da característica p na classe q no conjunto de dados original. Assim, no conjunto de dados produzido, o centro das classes é movido, mantendo os desvios padrão. A etapa de amostragem é repetida para produzir o mesmo número de instâncias para cada classe que o conjunto de dados original. A Figura 8 ilustra um exemplo de conjunto de dados sintetizado a partir da manipulação do conjunto de dados Iris (Figura 7), disponível no repositório OpenML.

Os indivíduos que irão criar a próxima geração no processo evolucionário serão obtidos pela média entre os λ indivíduos mais adequados da geração atual. A combinação dos λ indivíduos para encontrar μ'_{pq} consiste na operação de recombinação. Assumir μ'_{pq} como o melhor indivíduo na geração atual, onde a partir do qual serão gerados novos elementos, define a seleção de sobreviventes do método evolucionário adotado. A melhor solução alcançada é aquela que minimiza a função objetivo. De modo geral, o método otimização desenvolvido neste estudo de caso pode ser ilustrado no pseudocódigo **Algoritmo 3**.

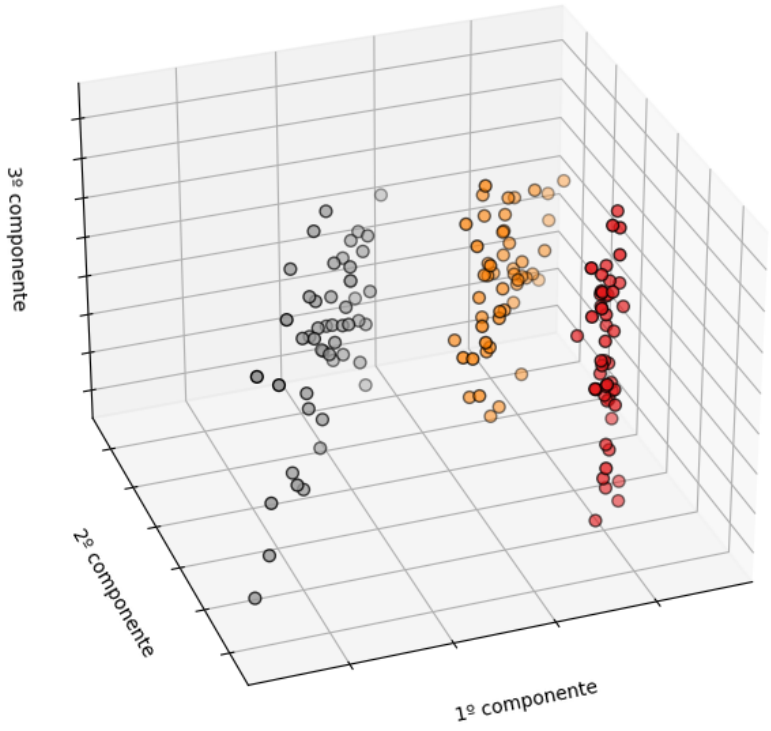
Os parâmetros e variáveis no **Algoritmo** 3 são definidos como: D é o conjunto de dados original de entrada; sub_d é o subconjunto de instâncias que serão evoluídas; x_{med} é a média entre os λ melhores indivíduos de uma determinada geração; α^2 é o vetor de desvios padrão das instâncias originais; $maxAval$ é número máximo de avaliações de função para cada cada solução gerada pelo algoritmo. γ é o tamanho da população; B é a matriz que define as coordenadas no CMA-ES; A é a matriz diagonal composta por autovalores; λ é a quantidade

Figura 7 – Conjunto de dados Iris Original



Fonte: Produzido pela autora (2022)

Figura 8 – Conjunto de dados Iris manipulado



Fonte: Produzido pela autora (2022)

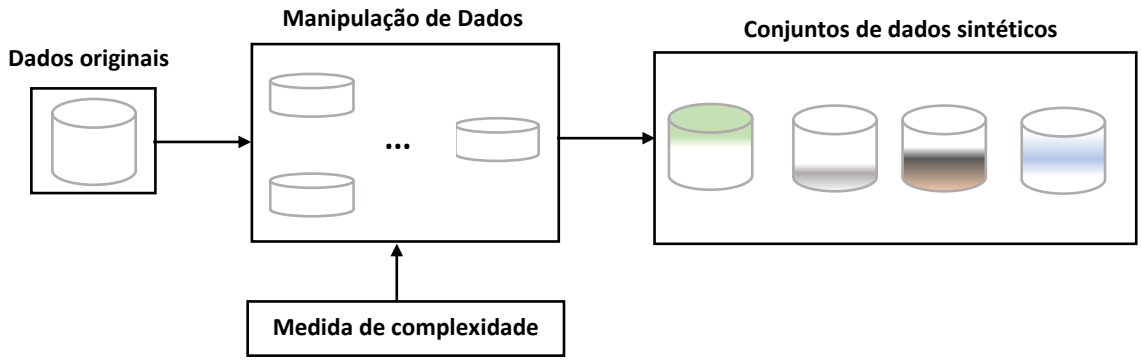
Algoritmo 3: Uma versão pseudocódigo do algoritmo de manipulação de conjuntos de dados reais

Entrada: D,δ,λ,γ,fn
// pré-processamento
$sub_d \leftarrow divideEntreClasse(D)$;
$x_{med} \leftarrow calculaMedia(sub_d)$;
$d \leftarrow$ *length*(x_{med})
$\alpha^2 \leftarrow dp(sub_d)$
$InicializarParametrosPadrão(d)$
$quantAval \leftarrow 0$
enquanto quantAval $<$ maxAval **faça**
  para $k \leftarrow$ até γ **faça**
    $x_k \leftarrow geraProle(x_{med}, \delta, B, A)$ //mutação
    $computa\ a\ aptidão\ de\ x_k$ // chamada da função objetivo (fn)
    $quantAval \leftarrow quantAval + 1$
  fim para
  $x_{med} \leftarrow calculaMedia(X_{1:\lambda})$
  $atualizaParametros()$
  se $computaErro(x_1) <= \epsilon$ **então**
    $pare$
  fim se
fim enquanto
$D_{new} \leftarrow geraD_{new}(x_1, \alpha^2, D)$
Saida: D_{new}

dos indivíduos mais aptos de uma determinada geração; d é a dimensão das variáveis; δ é o passo de mutação; x_k é o $k_ésimo$ indivíduo de uma determinada geração; $X_{1:\lambda}$ representa o conjunto composto pelos λ melhores indivíduos de uma determinada geração, considerando as suas aptidões; a função $geraD_{new}$ é responsável por receber a melhor solução retornada pelo método e o σ_{pq}, e reconstruir as instâncias do novo conjunto de dados (D_{new}), segundo a definição da Equação 4.3; fn é a função objetivo que será minimizada pelo método evolucionário. Para o funcionamento da abordagem proposta de manipulação de conjuntos de dados, a função objetivo foi definida de maneira a minimizar a função 4.2. Assim, um conjunto de dados sintético pode ser ilustrado como em Figura 9.

As partes coloridas representam as instâncias sintéticas, e as brancas são os conjuntos de dados originais, ou seja, aquelas instâncias que não passaram pelo processo de otimização. Desta forma, a partir de um conjunto de dados original, pode-se produzir uma diversidade de conjuntos de dados sintéticos com diversas medidas de complexidade.

Figura 9 – Pipeline de Geração de Conjuntos de Dados Sintéticos.



Fonte: Produzido pela autora (2022)

Usamos o *Covariance Matrix Adaptation Evolution Strategy* (CMA-ES) (CMA-ES) (HANSEN, 2006) para evoluir os conjuntos de dados os conjuntos de dados reais e gerar novos conjuntos de dados. O método CMA-ES é uma espécie de estratégia de evolução, um método de busca estocástico baseado em população, amplamente adotado e tem demonstrado eficácia em problemas de otimização de objetivo único de médio porte (HANSEN, 2006). Nossa proposta usa CMA-ES com parâmetros padrão e os mostrados na Tabela 2.

Tabela 2 – Parâmetros usados para otimizar subconjuntos de dados

Parâmetro	Valor
Máximo de avaliações de função (maxAval)	10^5
Erro de tolerância (ϵ)	10^{-3}
Passo de mutação (δ)	0.3
Tamanho da População (γ)	400
Melhores indivíduos da geração (λ)	40

Fonte: Produzido pela autora (2022)

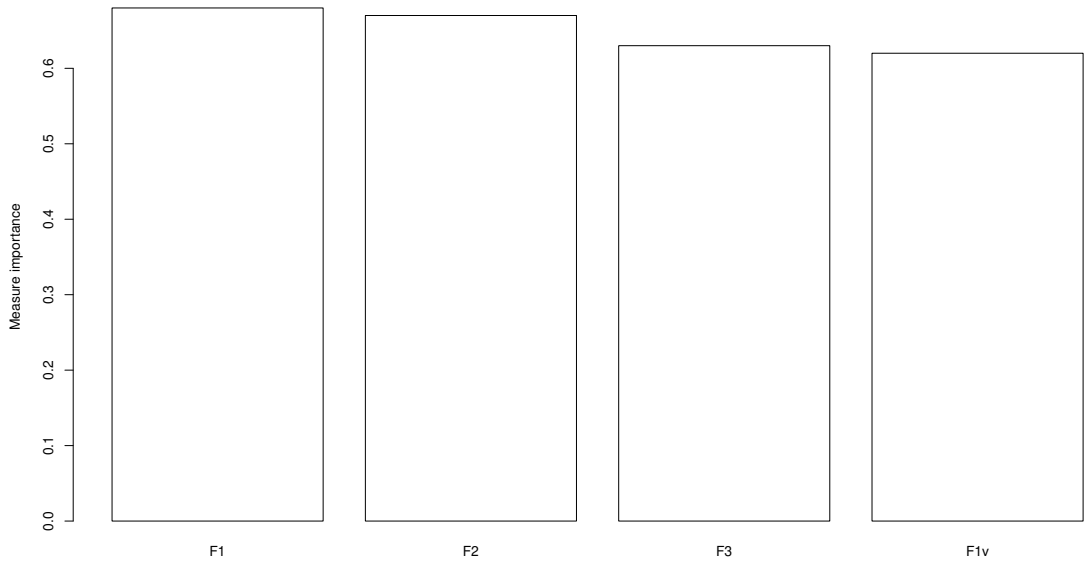
Executamos o CMA-ES 15 vezes para cada conjunto de dados original, com um limite de 10^5 avaliações de função, ou seja, o número de vezes que um conjunto de dados é gerado e testado. Os critérios de parada são pela número de avaliações de função ou quando o alvo está dentro da tolerância configurada (10^{-3}).

Dessa forma, geramos um máximo de 1170 novos conjuntos de dados. Após o processo de

avaliação, 56 conjuntos de dados foram descartados, pois eram muito semelhantes.

Conforme abordado anteriormente, neste estudo de caso, nossa abordagem de geração de conjunto de dados sintéticos é guiada por medidas de complexidade. Para escolher a medida de complexidade a ser utilizada, analisamos a relevância de cada medida de complexidade disponível em (GARCIA; LORENA; LEHMANN, 2019), com base no índice de *Gini*, na previsão do meta-classificador RF para escolher o melhor modelo de classificação dos conjuntos de dados reais disponíveis nesta pesquisa. A Figura 10 mostra as quatro medidas de complexidade consideradas as mais importantes para o meta-classificador de RF em todas as iterações. Para a análise inicial de nossa abordagem, a medida de complexidade $F1$, disponível na ferramenta ECoL(GARCIA; LORENA; LEHMANN, 2019), foi escolhida para guiar o processo de evolução. A medida $F1$ representa o valor da razão discriminante de *Fisher* mais alta entre todos as características disponíveis. É uma das medidas de sobreposição, que avalia o quão informativas são as características de entrada para separar as classes.

Figura 10 – Medidas de complexidade melhor classificadas selecionadas pelo meta-classificador RF



Fonte: Produzido pela autora (2022)

Esta nova abordagem de geração de conjuntos de dados sintéticos é semelhante à apresentada em (MELO; LORENA, 2018), onde o objetivo é criar novos conjuntos de dados com

diversidade nos valores de medida de complexidade alvo. No entanto, em nossa técnica, aplicamos uma estratégia de evolução em vez de uma estratégia de busca. Outro ponto divergente é que em (MELO; LORENA, 2018) é realizada uma troca iterativa de rótulos de pares de amostras de um conjunto de dados inicial. Em nosso estudo de caso, a técnica de manipulação permite certo controle sobre as características dos novos conjuntos de dados gerados. Por outro lado, em (SOARES, 2009), o resultado da permutação entre os atributos de entrada e destino são frequentemente conjuntos de dados anômalos, como discutido anteriormente.

De 78 problemas originais coletados no repositório OpenML, 1114 novos conjuntos de dados manipulados foram produzidos. Os detalhes das bases originais usadas neste estudo de caso podem ser vistos no Apêndice A. Todos os experimentos realizados, tanto para geração dos conjuntos de dados artificiais quanto para calcular a acurácia dos métodos nas bases(reais e sintéticas) foram executados em um computador com processador Intel Core i5 8400, 16GB de memória RAM (DDR4 2666MHz), HD de 1TB e sistema operacional Ubuntu Desktop 20.04.1 LTS 64 bits.

4.4.2.2 Caracterização de Dados

Ao contrário do primeiro estudo de caso, que se concentrou em árvores de decisão como um algoritmo de nível base, nosso segundo estudo de caso cobre algoritmos candidatos mais diversos, ou seja, que têm critérios de classificação diferentes. Consequentemente, meta-características mais diversas também são necessárias. Assim, adotamos 22 meta-características (ver Tabela 3), que foram bem investigadas na literatura (LORENA et al., 2018; MELO; LORENA, 2018; GARCIA; LORENA; LEHMANN, 2019; LORENA et al., 2018). As medidas foram implementadas na ferramenta ECoL (GARCIA; LORENA; LEHMANN, 2019; LORENA et al., 2018), a qual foi utilizada no processo de caracterização de dados do presente estudo de caso. Os meta-exemplos produzidos após a caracterização são fornecidos como entrada para o *Módulo OCC* que faz parte da nossa proposta de seleção de conjunto de dados relevantes para MA. Os meta-exemplos selecionados são então rotulados pelo *Módulo Avaliação de Algoritmos*, de acordo com o desempenho dos algoritmos candidatos em cada problema.

Ambos os conjuntos de dados (reais e sintéticos) foram mapeados no espaço de medição

Tabela 3 – Medidas de Caracterização

Medidas	**Descrição**
F1	Razão Discriminante de Fisher Máxima
F1v	Razão discriminante de Fisher máxima do vetor direcional
F2	Volume da região de sobreposição
F3	Eficiência máxima de características individuais
F4	Eficiência de características coletivas
L1	Soma das distâncias de erro por programação linear
L2	Taxa de erro de um classificador linear
L3	Não linearidade de um classificador linear
N1	Fração de pontos limítrofes
N2	Razão de distância do vizinho mais próximo intra/interclasse
N3	Taxa de erro do classificador vizinho mais próximo
N4	Não linearidade do classificador vizinho mais próximo
T1	Fração de hiperesferas cobrindo dados
LSCAvg	Cardinalidade média do conjunto local
Density	Densidade média da rede
ClsCoef	Coeficiente de agrupamento
Hubs	Pontuação do Hub
T2	Número médio de pontos por dimensão
T3	Número médio de pontos por dimensão PCA
T4	Razão da dimensão PCA para a dimensão bruta
C1	Entropia de proporções de classe
C2	Razão de desequilíbrio

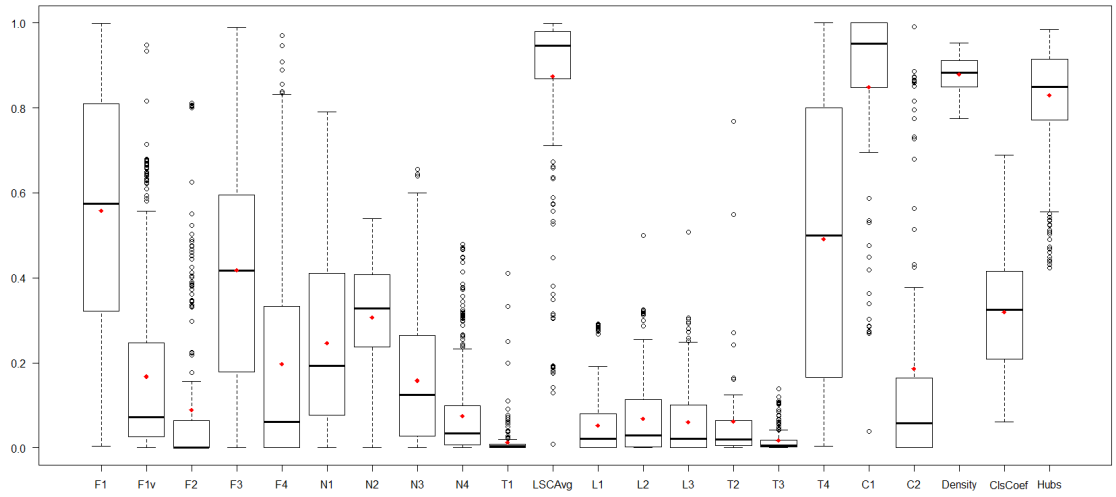
Fonte: Produzido pela autora baseado em (LORENA et al., 2018)

definido pelas medidas de complexidade, como pode ser visto em LORENA et al. (2018). Observamos que os conjuntos de dados sintéticos (veja a figura 11) fornecem uma maior cobertura do espaço de medição e melhor granularidade do que problemas do mundo real (veja a figura 12).

4.4.2.3 Avaliação de Algoritmo

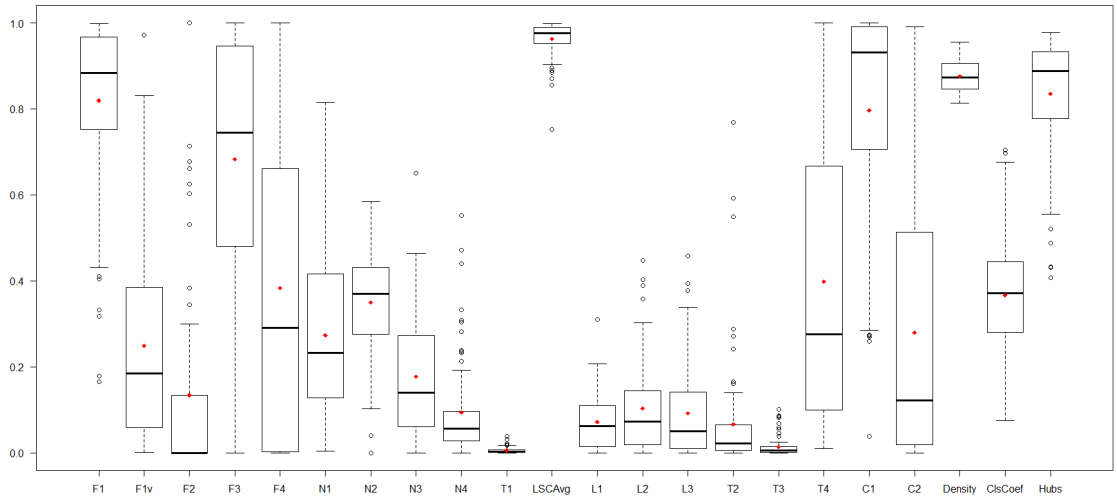
O *Módulo Avaliação de Algoritmos*, como mencionado anteriormente, é responsável por rotular os meta-exemplos selecionados pelo *Módulo OCC*. Cada meta-exemplo representa um

Figura 11 – 1114 conjuntos de dados sintéticos mapeados nas vinte e duas medidas de complexidade



Fonte: Produzido pela autora (2022)

Figura 12 – 78 conjuntos de dados originais mapeados nas vinte e duas medidas de complexidade



Fonte: Produzido pela autora (2022)

problema, ou seja, um conjunto de dados real ou sintético. Neste módulo, cada meta-exemplo selecionado é rotulado com um dos quatro rótulos candidatos (K-NN, JR, NB ou J48). A classe de cada meta-exemplo é baseada nos resultados dos experimentos no problema correspondente. A medida de desempenho adotada foi a precisão da classificação, estimada por meio de *10-fold cross-validation*. A tabela 4 mostra as distribuições de classes para os meta-dados, tanto os dados reais quanto os sintéticos.

Tabela 4 – Distribuição de classes (%) dos meta-dados correspondentes aos conjuntos de dados originais e sintéticos

Meta-dado	No. de meta-exemplos	JR	J48	NB	k-NN
Real	78	22	26	15	37
Sintético	1114	21	29	20	30

Fonte: Produzido pela autora (2022)

As distribuições de classes observadas nos meta-dados apresentam diferenças, o que indica que pode haver algumas diferenças subjacentes entre conjuntos de dados reais e sintéticos. No entanto, como observado também no primeiro estudo de caso, as diferenças não são tão grandes na distribuição de classes considerando os conjuntos de dados reais e sintéticos separadamente.

4.5 META-APRENDIZAGEM

O *Módulo Meta-aprendizado* recebe um conjunto de meta-exemplos rotulados como resultado dos passos anteriores em nosso processo experimental (veja Figura 3). Cada meta-exemplo está associado a um conjunto de dados e armazena as características descritivas e o valor da classe indicando o melhor algoritmo candidato para o conjunto de dados. Nesta investigação, adotamos dois algoritmos de classificação como meta-aprendiz: RF e KNN. Esses algoritmos são aplicados como meta-classificadores que irão indicar o melhor modelo para um conjunto de dados de teste específico. Em nosso protótipo, utilizamos as implementações desenvolvidas na linguagem de programação Java disponível no ambiente WEKA (I.H.WITTEN; FRANK, 2005). A metodologia de avaliação, adotada nos dois estudos de caso, é descrita na próxima seção.

4.6 METODOLOGIA DE AVALIAÇÃO

Como metodologia, adotamos uma avaliação de validação com 500 repetições para avaliar o desempenho de aprendizagem no meta-nível. Os conjuntos de dados reais são particionados aleatoriamente em duas partições mutuamente exclusivas (teste e treinamento), com 10 % e 90 % dos conjuntos de dados, respectivamente. Este procedimento é repetido em cada iteração. No conjunto de treinamento, além dos conjuntos de dados originais, todos os conjuntos de dados sintéticos gerados a partir deles são usados. Consequentemente, garantimos que os conjuntos de dados sintéticos gerados a partir dos conjuntos de dados originais de teste não serão incluídos no conjunto de treinamento.

Em cada iteração de validação, a técnica OCC (método PU Learning ou NN-d) é aplicada para selecionar meta-exemplos sintéticos no conjunto de treinamento. O meta-aprendiz é então construído usando os meta-exemplos reais e os sintéticos selecionados para treinamento e é avaliado para a seleção de algoritmo nos meta-exemplos de teste. A medida de desempenho de MA adotada em nosso trabalho foi a precisão média da classificação ao longo das 500 execuções de validação. Para avaliar o impacto do NL, repetimos esta metodologia variando o nível de ruído esperado nos exemplos positivos (parâmetro b). Para o método NN-d, até 300 instâncias sintéticas foram selecionadas e rotuladas no primeiro estudo de caso e até 400 instâncias no segundo estudo de caso. Em ambos os estudos de caso, a seleção das instâncias é feita de forma aleatória, a partir das instâncias aceitas pelo método NN-d.

Existem duas linhas de base para comparar nossa abordagem. Primeiro, usando como dados de treinamento apenas os meta-exemplos gerados a partir de conjuntos de dados originais (OD). Em segundo lugar, usando os conjuntos originais juntamente com todos os meta-exemplos sintéticos, gerados a partir desses conjuntos sem seleção (OD + todos os sintéticos). Essas linhas de base correspondem a situações extremas nas quais não consideramos nenhum meta-exemplo sintético para MA ou consideramos todos os meta-exemplos sintéticos sem nenhuma seleção dos relevantes. A presente metodologia é aplicada em ambos os estudos de caso (1 e 2). Por fim, nossa implementação foi desenvolvida na linguagem de programação Java usando o framework WEKA (I.H.WITTEN; FRANK, 2005) e seus parâmetros padrão.

4.7 CONSIDERAÇÕES FINAIS

Neste capítulo, a solução proposta nessa tese para seleção de conjuntos de dados sintéticos relevantes em contexto de MA é apresentada. Inicialmente, foi realizada uma introdução ao problema abordado, incluindo análise de trabalhos relacionados e suas limitações. Foram discutidas as principais motivações da nossa proposta. Uma descrição esquemática da solução proposta é mostrada. Além de discorrer sobre os detalhes do funcionamento de cada módulo da proposta, os métodos adotados foram descritos.

Ao longo deste capítulo os diversos aspectos das configurações dos experimentos puderam ser vistos. Inicialmente, são apresentados experimentos em dois estudos de caso. As Subseções 4.4.1 e 4.4.2, descrevem os conjuntos de meta-exemplos adotados nos experimentos em cada estudo de caso. Foram discutidos também os módulos propostos, que são específicos para cada estudo de caso. Em cada estudo de caso foram detalhadas as técnicas de geração de conjuntos de dados sintéticos adotadas. O processo de caracterização de cada estudo de caso também foi demonstrado juntamente com as medidas usadas como meta-características nas meta-bases utilizadas em nossos experimentos. Além disso, foi descrita a avaliação da qualidade dos meta-aprendizes treinados com os meta-exemplos selecionados usando técnicas de OCC.

Na Seção 4.5, apresentamos os algoritmos de meta-aprendizagem, que foram adotados em ambos os estudos de caso. Em seguida, a Seção 4.6, descreve a metodologia de experimentos para avaliar a proposta de OCC para seleção de conjuntos de dados relevantes.

O próximo capítulo traz os resultados experimentais observados em cada estudo de caso proposto nessa tese. Além disso, será realizada uma discussão desses resultados.

5 RESULTADOS EXPERIMENTAIS

Neste capítulo, apresentamos os resultados dos experimentos para avaliar a qualidade de meta-aprendizes quando treinados com meta-exemplos selecionados usando técnicas de OCC. Em cada estudo de caso, temos os resultados de acordo com cada uma das técnicas de OCC adotadas para seleção: PU Learning e NN-d. A estratégia proposta é comparada com duas linhas de base, conforme mencionado na Seção 4.6. Inicialmente, apresentamos os resultados observados no primeiro estudo de caso. Em seguida, são os resultados obtidos no segundo estudo de caso são relavados. Por fim, discutiremos os resultados observados em ambos os estudos de caso.

5.1 RESULTADOS - ESTUDO DE CASO 1

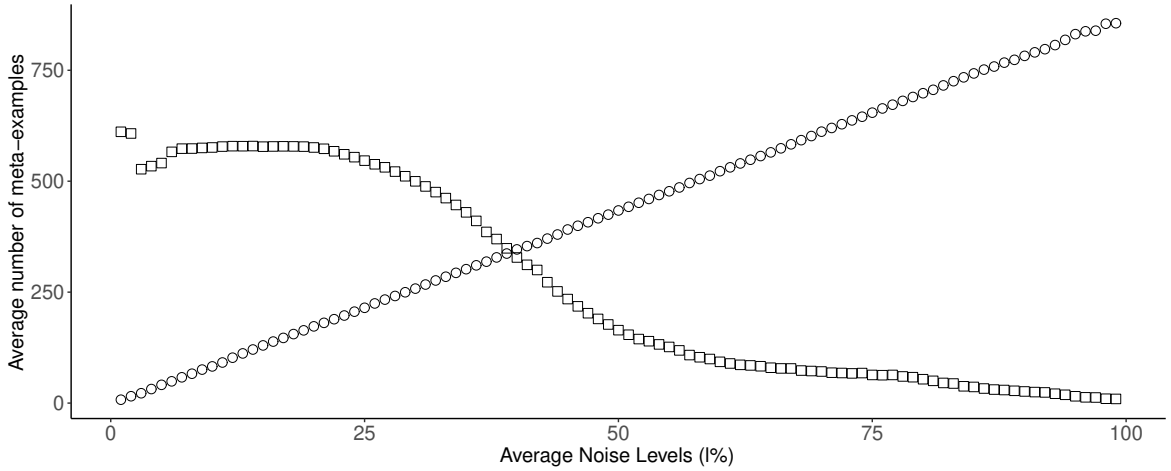
Neste primeiro estudo de caso, detalhado na Subseção 4.4.1 do capítulo 4, na linha de base adotada (OD + todos os sintéticos), descrita na seção 4.6 do mesmo capítulo, o meta-aprendiz RF, por exemplo, obteve o valor 60,14 % de acurácia média do MA. Desempenho significativamente maior do que usar apenas os meta-exemplos originais (acurácia média de 46,20 %). No entanto, um grande número de meta-exemplos é adotado e isso aumenta o custo de avaliação de algoritmos candidatos no processo de rotulagem de meta-exemplos. Assim, a aprendizagem OCC é adotada em nossa abordagem para selecionar um número reduzido de meta-exemplos sintéticos relevantes para atingir um nível de acurácia igual ou superior ao atingido quando usando todos os conjuntos de dados (60,14 %). Os resultados observados após a seleção de cada uma das técnicas utilizadas são apresentados a seguir.

5.1.1 PU learning

Como discutido na seção 4.2 do capítulo 4 da presente tese, o PU Learning realiza a seleção de meta-exemplos em duas etapas. A primeira etapa é responsável por selecionar um conjunto de instâncias negativas que serão utilizadas para treinar o classificador SVM na segunda etapa. O número de instâncias negativas selecionadas depende do nível de ruído considerado (ver

Figura 13). A Figura 14 e a Figura 15 mostram o desempenho dos meta-aprendiz RF e KNN, respectivamente, variando o nível de ruído (NL). Observe que o NL no PU Learning é um parâmetro que reflete a confiança de que todos os conjuntos de dados reais são relevantes para o MA. Um valor baixo de NL sugere que a maioria ou todos os meta-exemplos reais são relevantes. Em primeiro lugar, consideramos dois casos extremos:

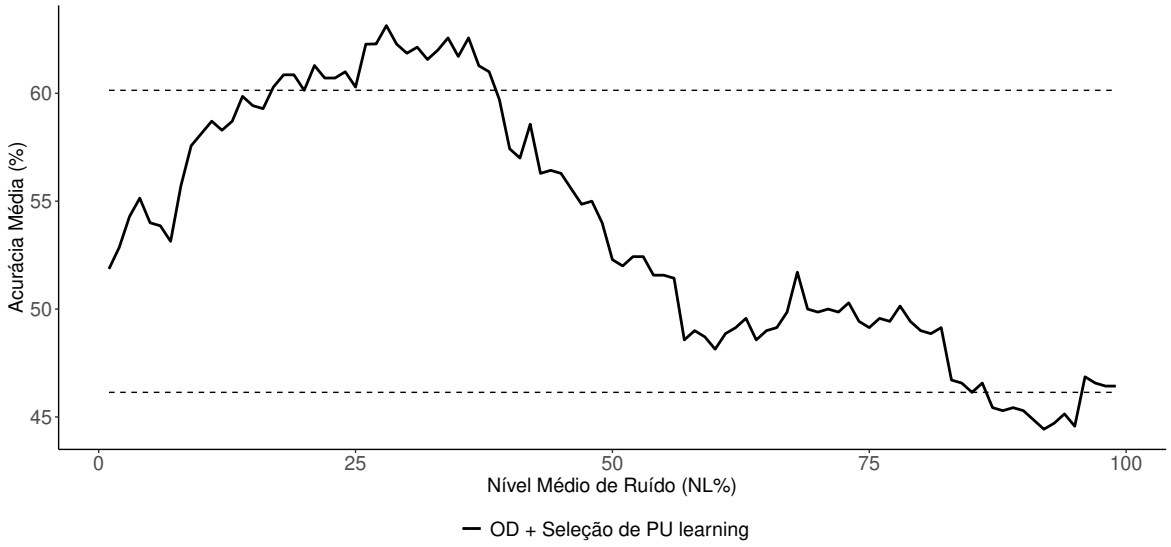
Figura 13 – A relação entre o número de instâncias negativas selecionadas (N_r) na primeira etapa consideradas verdadeiramente negativas (círculos) e o número de instâncias selecionadas (U_p) na segunda etapa de aprendizagem de PU consideradas relevantes, ou seja, pertencentes a classe positiva (quadrados).



Fonte: Produzido pela autora (2022)

- Para valores mínimos de NL, poucos meta-exemplos negativos são selecionados para a segunda etapa em comparação com os exemplos positivos. Consequentemente, o SVM treinado tende a classificar uma grande quantidade dos meta-exemplos sintéticos restantes como positivos (vários deles provavelmente não são relevantes). Assim, neste caso, a acurácia do MA é subótima e localizada entre as duas linhas de base (o segmento inicial da Figura 14 e da Figura 15);
- Por outro lado, se tivermos valores altos de NL, uma quantidade excessiva de exemplos negativos será fornecida para o conjunto de treinamento da SVM na segunda etapa. Desta forma, apenas alguns meta-exemplos sintéticos serão finalmente classificados como positivos (relevantes). É possível observar que, neste caso, a acurácia do MA tende a se aproximar do valor de acurácia observado quando apenas instâncias reais foram utilizadas (o segmento final da Figura 14 e da Figura 15).

Figura 14 – Acurácia média obtida por RF usando PU Learning para selecionar meta-exemplos sintéticos. O conjunto de treinamento é inicializado com todos os meta-exemplos originais, deixando apenas meta-exemplos sintéticos para a seleção de PU Learning. As linhas horizontais tracejadas representam a acurácia obtida usando todos os meta-exemplos originais e meta-exemplos sintéticos (parte superior) ou apenas os meta-exemplos originais (parte inferior).

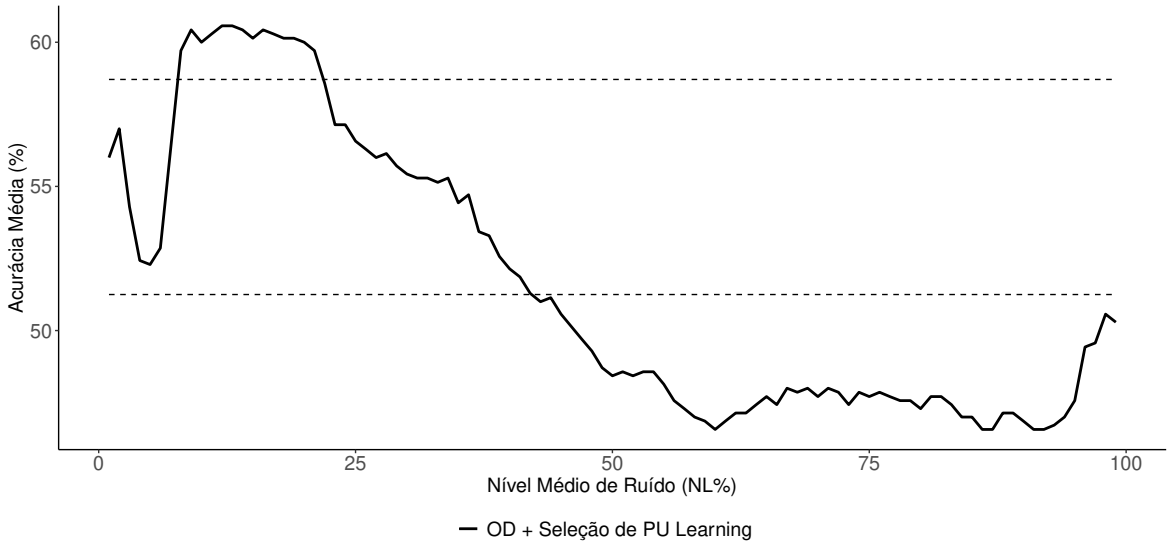


Fonte: Produzido pela autora (2022)

Para valores intermediários de NL (de 17 % a 38 %), ao considerar o meta-aprendiz RF, a acurácia do MA observada foi maior do que usando todos os conjuntos de dados sintéticos. O melhor resultado foi 63,14 % de acurácia quando o NL é 28%. Após a segunda etapa, cerca de 59,53 % dos meta-exemplos sintéticos foram finalmente selecionados como relevantes e os restantes foram descartados. Já para KNN, a acurácia do MA observada foi superior ao uso de todos os conjuntos de dados sintéticos para valores intermediários de NL (de 8% a 21%). O melhor resultado foi 60,57 % de acurácia quando NL é 13%. Assim, cerca de 34% dos meta-exemplos sintéticos foram desprezados e o restante foi selecionado como conjuntos de dados relevantes para MA. O meta-aprendiz RF certamente foi menos sensível ao conjunto de meta-exemplos de treinamento. A Tabela 5 sumariza a acurácia média obtida pelos meta-aprendizes KNN e RF quando a técnica PU Learning foi usada para selecionar meta-exemplos sintéticos em nossos experimentos.

Na prática, é difícil saber de antemão qual o NL a ser escolhido no método proposto. Dependerá da suposta relevância dos meta-exemplos disponíveis, ou seja, até que ponto se pode confiar na relevância e representatividade dos conjuntos de dados disponíveis em repositórios

Figura 15 – Acurácia média obtida por KNN usando PU Learning para selecionar meta-exemplos sintéticos. O conjunto de treinamento é inicializado com todos os meta-exemplos originais, deixando apenas meta-exemplos sintéticos para a seleção de PU Learning. As linhas horizontais tracejadas representam a acurácia média observada usando todos os meta-exemplos originais e meta-exemplos sintéticos (parte superior) ou apenas os meta-exemplos originais (parte inferior).



Fonte: Produzido pela autora (2022)

Tabela 5 – Acurácia média (%) obtida pelos algoritmos RF e KNN e o número médio de meta-exemplos em conjuntos de treinamento formados pela seleção de PU Learning.

NL(%) /Dados Treinamento	kNN			Random Forest		
	OD	OD + todos	OD + Seleção PU	OD	OD + todos	OD + Seleção PU
13%	51.25	58.71	**60.57 (+1.86)**	46.20	60.14	58.71 (-1.43)
28%			56.14 (-2.57)			**63.14 (+3)**
Número médio de exemplos	57	933.35	**635.87**	57	933.35	635.87
			578.68			**578.68**

Fonte: Produzido pela autora (2022)

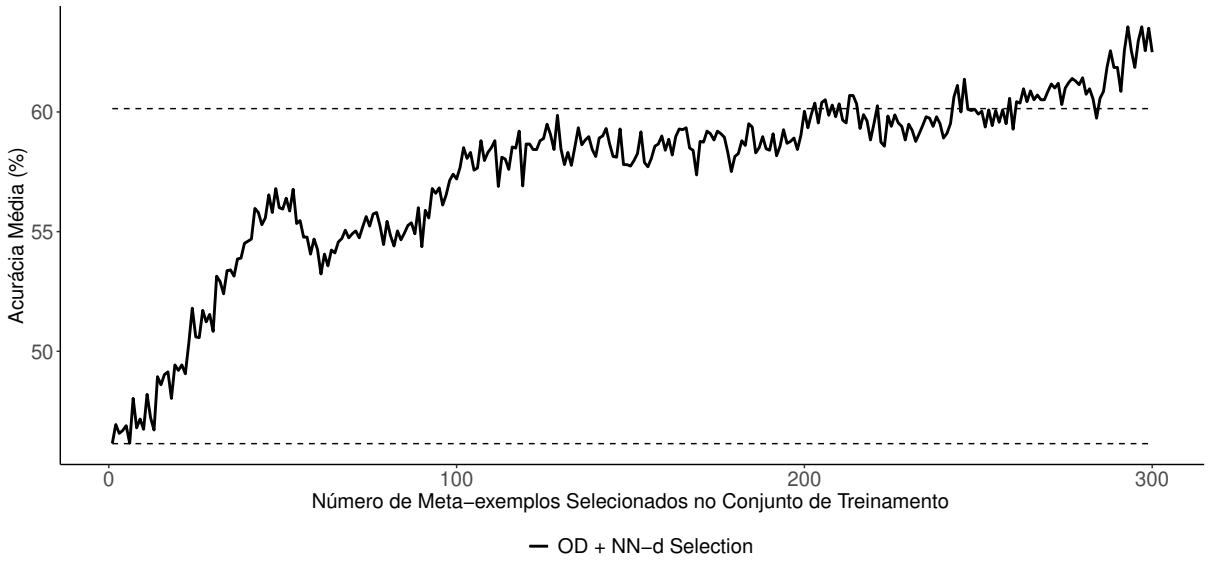
como o UCI, por exemplo. Ao considerar a Figura 14, os resultados sugerem, por exemplo, que aproximadamente 15 % dos meta-exemplos reais adotados em nossos experimentos não são completamente relevantes. Alguns valores de NL(por exemplo, 50%) pareciam ser muito extremos e foram considerados apenas nos experimentos a fim de avaliar a robustez da abordagem proposta. Deixamos para uma investigação futura a questão em aberto que é como estimar o nível de ruído.

5.1.2 NN-d

As Figuras 16 e 17 mostram a acurácia obtida pelos meta-aprendizes KNN e RF quando a técnica NN-d foi usada para selecionar meta-exemplos sintéticos em nossos experimentos. Os resultados mostraram, de maneira geral, uma melhora na acurácia quando o método NN-d foi comparado com as duas linhas de base adotadas. O desempenho do MA é impactado positivamente. Assim como na seleção de PU Learning, RF obteve os melhores resultados quando comparados aos resultados obtidos pelo meta-aprendiz KNN. O meta-aprendiz RF parece ser mais robusto a meta-exemplos ruidosos e redundantes no conjunto de treinamento, o que pode ser explicado pelo fato de RF utilizar *bagging* para produzir os componentes do conjunto, apresentando assim maior robustez em relação à qualidade dos conjuntos de treinamento. Após a inclusão de 293 meta-exemplos sintéticos selecionados por NN-d, o RF obteve uma taxa de acurácia de 63,56 %, enquanto o meta-aprendiz KNN obteve uma taxa de acurácia de 61,23 %. Dessa forma, , os melhores resultados foram observados usando o método RF como meta-aprendiz em comparação com o método KNN. A Tabela 6 resume a acurácia obtida pelos meta-aprendizes KNN e RF quando o método NN-d foi usado para selecionar meta-exemplos sintéticos.

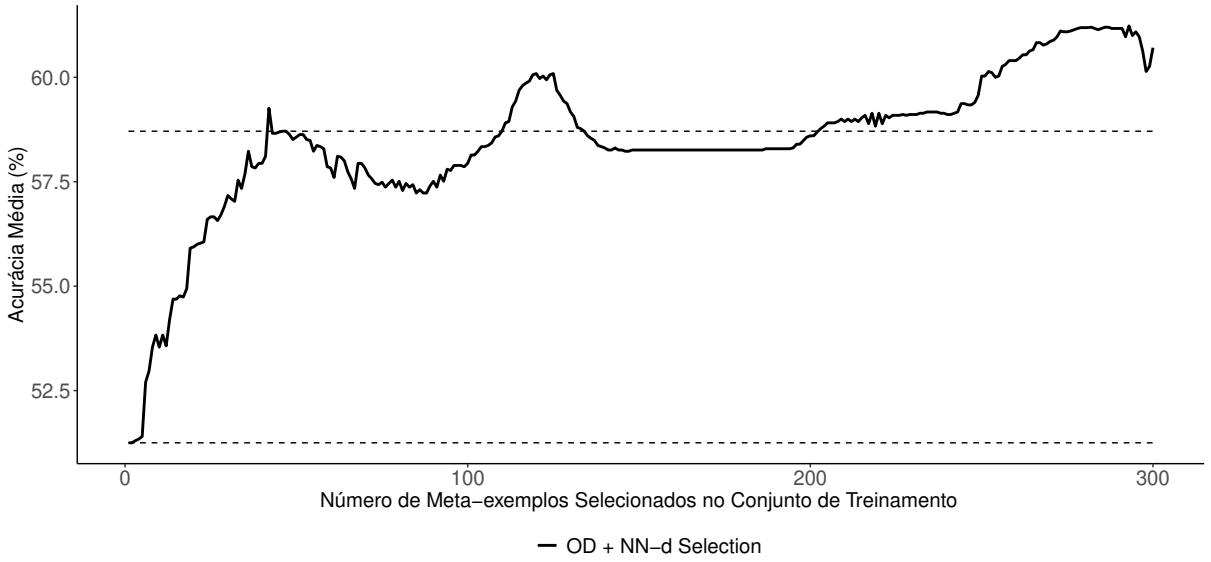
Ao comparar os métodos de aprendizagem NN-d e PU Learning, observamos que os resultados obtidos pelo NN-d são semelhantes aos melhores resultados obtidos pelo PU Learning.

Figura 16 – Acurácia média obtida por RF usando NN-d para selecionar meta-exemplos sintéticos. O conjunto de treinamento é inicializado com todos os meta-exemplos originais, deixando apenas meta-exemplos sintéticos para seleção. As linhas horizontais tracejadas representam a acurácia obtida usando todos os meta-exemplos originais e meta-exemplos sintéticos (parte superior) ou apenas os meta-exemplos originais (parte inferior).



Fonte: Produzido pela autora (2022)

Figura 17 – Acurácia média obtida por KNN usando NN-d para selecionar meta-exemplos sintéticos. O conjunto de treinamento é inicializado com todos os meta-exemplos originais, deixando apenas meta-exemplos sintéticos para seleção. As linhas horizontais tracejadas representam a acurácia obtida usando todos os meta-exemplos originais e meta-exemplos sintéticos (parte superior) ou apenas os meta-exemplos originais (parte inferior).



Fonte: Produzido pela autora (2022)

Tabela 6 – Acurácia média (%) obtida pelos meta-aprendizes RF e KNN e o número médio de exemplos no conjunto de treinamento que consiste na seleção NN-d e instâncias originais.

Algoritmos/Dados de Treinamento	**OD**	**OD + todos**	**OD + Seleção NN-d**
RF	46.20	60.14	**63.56 (+3.42)**
k-NN	51.25	58.71	**61.23 (+2.52)**
Número médio de exemplos	57	933.35	**350**

Fonte: Produzido pela autora (2022)

5.2 RESULTADOS - ESTUDO DE CASO 2

Nesta seção, são apresentados os resultados do nosso segundo estudo de caso. Conforme já discutido, neste estudo de caso adotamos uma tarefa de MA, que consiste em escolher entre quatro diferentes métodos de classificação, um método específico, ou saber qual ferramenta utilizar para um novo problema de classificação, representado por um determinado conjunto de dados, de forma que as soluções oferecidas sejam adequadas e eficientes. Os detalhes do segundo estudo de caso podem ser vistos na Subseção 4.4.2 do capítulo 4 do presente trabalho. Assim como no primeiro estudo de caso, os resultados aqui encontrados também são apresentados em relação às técnicas de OCC adotadas, (PU Learning e NN-d). A abordagem proposta é comparada com duas linhas de base, conforme descrito na Seção 4.6 também no capítulo 4.

A acurácia média do MA obtida pela linha de base (OD + todos os sintéticos), sempre que o método RF é adotado, foi de 63,92%. Esse valor foi foi expressivamente maior do que o desempenho observado quando usando apenas conjuntos de dados originais (tendo 47,81% de acurácia média). Contudo, foi necessário um grande número de conjuntos de dados, tornando alto o custo de avaliar algoritmos candidatos no processo de rotulagem de meta-exemplos. Portanto, o objetivo de aprendizagem OCC é selecionar um número reduzido de conjuntos de dados sintéticos relevantes, a fim de atingir um nível de acurácia próximo ao de todos os conjuntos de dados (63,92 %). Os resultados observados após a seleção de cada técnica adotada são apresentados a próxima seção..

5.2.1 PU learning

Conforme realizado no estudo de caso 1 e descrito na Seção 4.2 do Capítulo 4, o método PU Learning seleciona os conjuntos de dados em duas etapas. Inicialmente, um conjunto negativo de instâncias é escolhido. Na segunda etapa, o classificador SVM é treinado com o conjunto de instâncias negativas selecionado anteriormente.

As Figuras 18 e 19 ilustram, respectivamente, o desempenho dos meta-aprendizes RF e KNN, variando o nível de ruído. Para analisar o comportamento do método PU Learning com os novos conjuntos de dados, neste segundo estudo de caso, consideramos também, inicialmente, situações extremas. No contexto do método PU Learning, o número de instâncias negativas selecionadas na primeira etapa não pode ser extremamente pequeno ou extremamente grande. Cada conjunto de dados tem seu próprio nível de ruído. O objetivo de analisar o nível de ruído extremo é encontrar o nível mais adequado para o treinamento da SVM na segunda etapa. Para níveis de ruído extremamente baixos, a SVM tende a classificar os dados como positivos (mas possivelmente, muitos deles não são). Por outro lado, para níveis de ruído extremamente altos, um grande número de instâncias negativas é introduzido no treinamento da SVM e, portanto, poucos conjuntos de dados são considerados como positivos (relevantes). Observando os níveis médios de ruído (NL) nas Figuras 18 e 19, temos três situações:

- Extremamente baixo: neste caso, a acurácia do MA é subótima e está entre as duas linhas de base (NL abaixo de 10% nas Figuras. 18 e 19);
- Mais adequado aos dados: a acurácia do MA tende a ser mais próxima do valor de acurácia buscado (NL entre 10 e 30 % nas Figuras. 18 e 19), e neste caso, temos maior acurácia em comparação àquela com todos os meta-exemplos sintéticos;
- Extremamente alto: a acurácia do MA tende a se aproximar do valor de acurácia observado usando como dados de treinamento apenas os meta-exemplos originais (NL maior que 30 % nas Figuras. 18 e 19).

Os resultados alcançados no processo de MA, neste segundo estudo de caso, corroboram com nossa hipótese de que as técnicas de OCC são capazes de selecionar exemplos relevantes de treinamento. Observando o desempenho do meta-aprendiz RF podemos perceber que o melhor resultado foi quando adotamos 17% de NL apresentando 68,54% de acurácia. O

método PU Learning foi capaz de reduzir em 54% os conjuntos de dados sintéticos, os quais foram considerados irrelevantes. Portanto, de uma média de 996 conjuntos de dados sintéticos, 458,15 foram considerados relevantes para o treinamento do meta-aprendiz RF. Levando em consideração o meta-classificador KNN, o melhor resultado foi quando 18% de NL foi adotado, produzindo um valor de acurácia de 59,69%. Assim, 45% dos conjuntos de dados sintéticos foram considerados relevantes (448,19). O resumo da acurácia alcançada pelos meta-aprendizes RF e KNN quando adotado o método o PU Learning para selecionar conjuntos de dados sintéticos pode ser visto na Tabela 7.

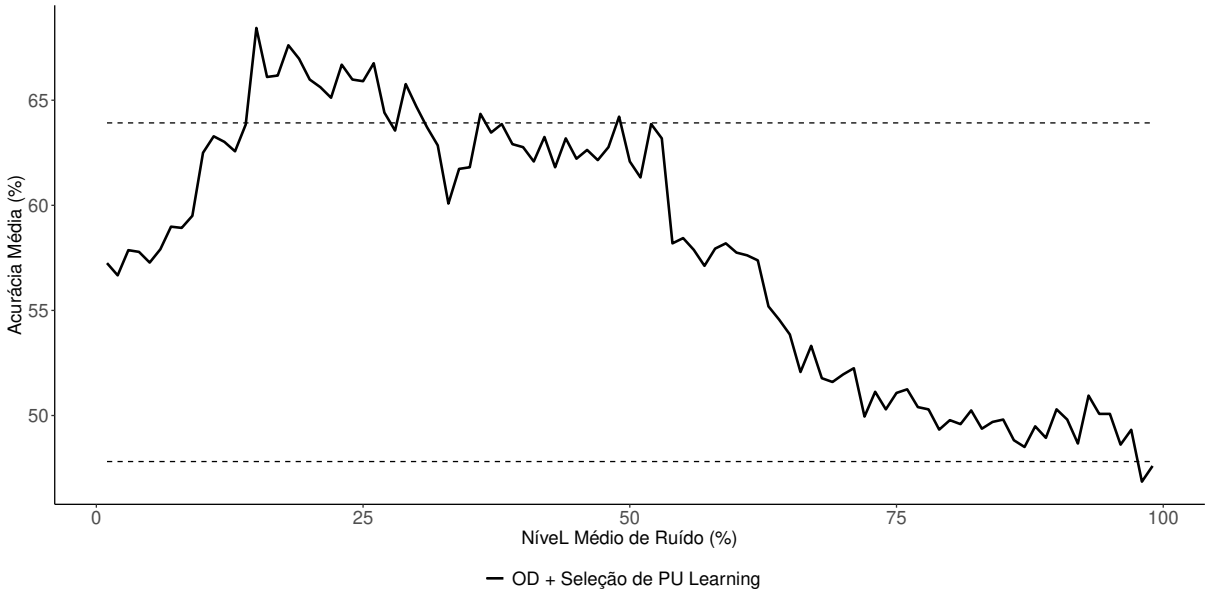
Tabela 7 – Acurácia média (%) obtida pelos algoritmos Random Forest e KNN e o número médio de exemplos em conjuntos de treinamento formados pela seleção de PU Learning .

NL(%) /Dados treinamento	k-NN			Random Forest		
	OD	OD + todos	OD + Seleção PU	OD	OD + todos	OD + Seleção PU
17%	45.60	54.77	59.50 (+4.73)	47.81	63.92	**68.54** (+4.62)
18%			**59.69** (+4.92)			66.15 (+2.23)
Número médio de exemplos	70	1066	528.15	70	1066	**528.15**
			518.19			518.19

Fonte: Produzido pela autora (2022)

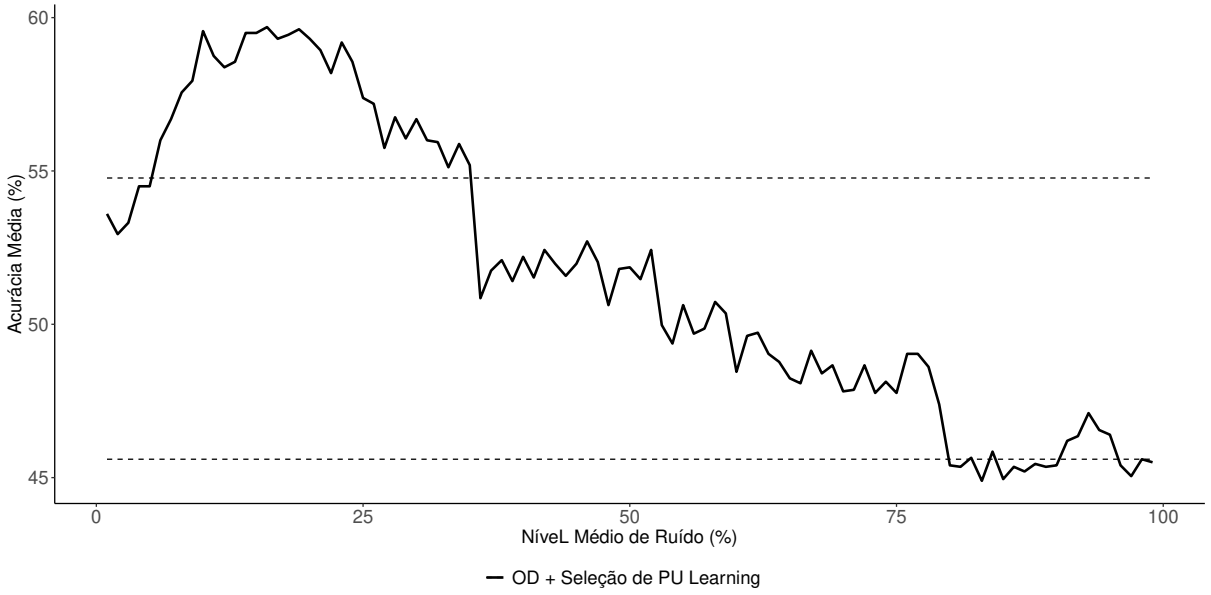
Os resultados revelaram ganho sempre que os meta-exemplos selecionados pelo método PU Learning foram comparados com as duas linhas de base adotadas. Neste estudo de caso, também decidimos ultrapassar os valores razoáveis adotando níveis de ruído excessivos (baixo e alto), como mostrado nas Figuras. 18 e 19, a fim de avaliar o comportamento e a força da nossa abordagem, uma vez que a princípio o nível de ruído a ser adotado é desconhecido. Neste caso, os dados parecem ter relevância semelhante para ambos os meta-aprendizes (RF e KNN). Os níveis de ruído mais adequados para ambos os métodos também foram valores muito semelhantes (ver Tabela 7).

Figura 18 – Acurácia média obtida por RF usando PU Learning para selecionar conjuntos de dados sintéticos. O conjunto de treinamento é inicializado com todos os conjuntos de dados originais, deixando apenas conjuntos de dados sintéticos para seleção de PU. As linhas horizontais tracejadas representam a acurácia obtida usando todos os conjuntos de dados originais e sintéticos (parte superior) ou apenas os conjuntos de dados originais (parte inferior).



Fonte: Produzido pela autora (2022)

Figura 19 – Acurácia média obtida por KNN usando PU Learning para seleção de conjuntos de dados sintéticos. O conjunto de treinamento é inicializado com todos os conjuntos de dados originais, deixando apenas conjuntos de dados sintéticos para seleção de PU. As linhas horizontais tracejadas representam a acurácia obtida usando todos os conjuntos de dados originais e sintéticos (parte superior) ou apenas os conjuntos de dados originais (parte inferior).

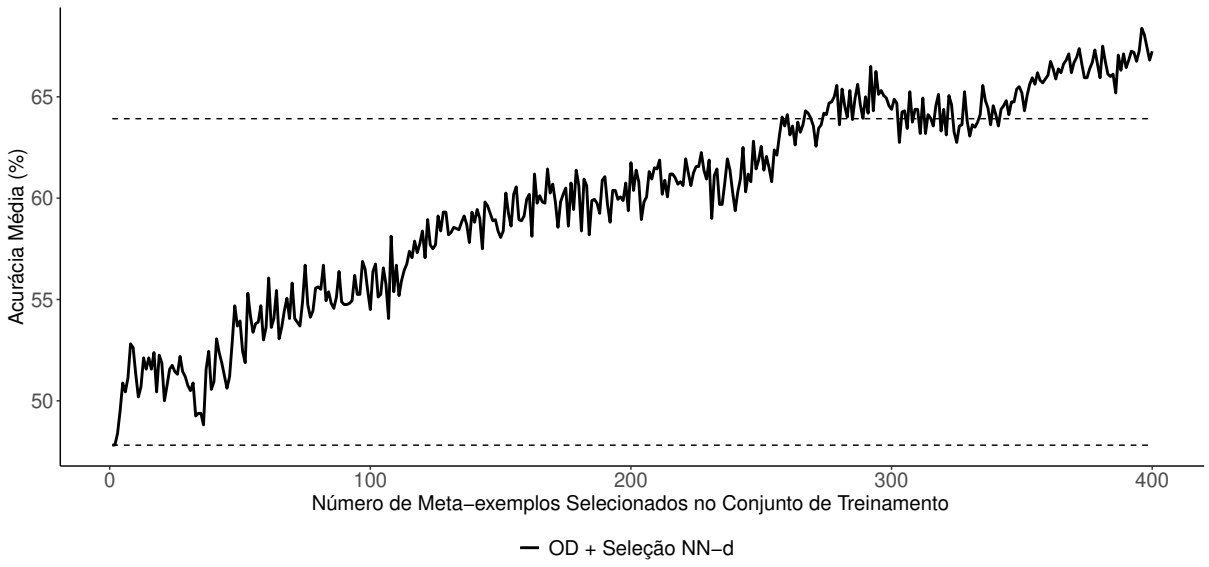


Fonte: Produzido pela autora (2022)

5.2.2 NN-d

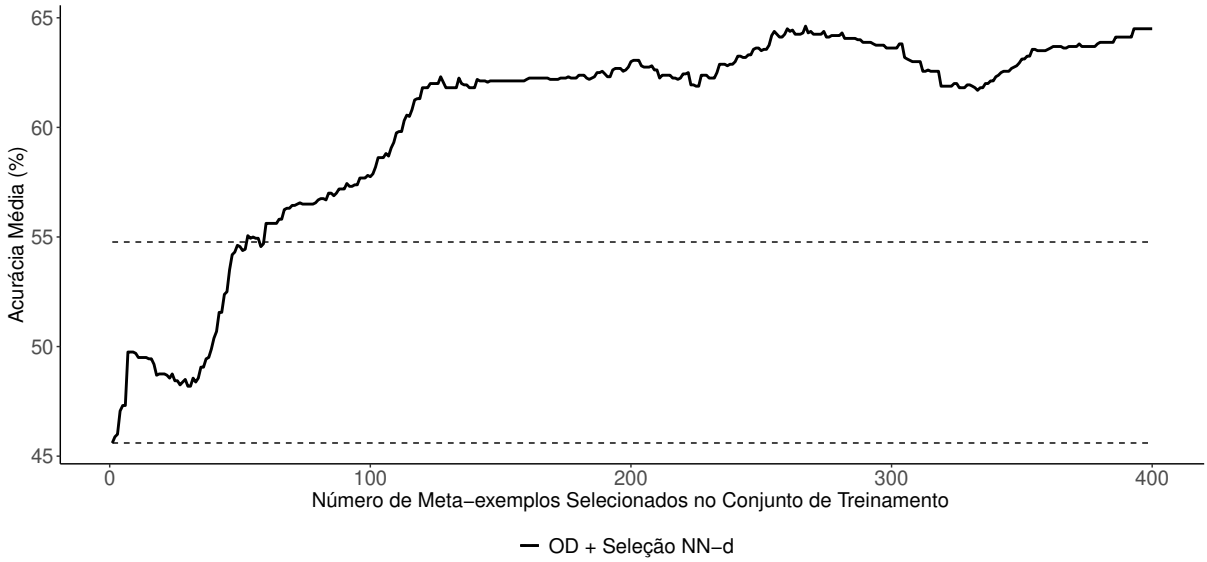
Observamos resultados expressivos nos experimentos com a técnica NN-d. Nas Figuras 20 e 21, são mostradas as acurácias obtidas pelos métodos meta-aprendizes KNN e RF quando treinados com os meta-exemplos sintéticos selecionados por NN-d. Os resultados mostraram ganho significativo em acurácia sempre que o método de aprendizagem NN-d foi comparado com as linhas de base adotadas. O desempenho do MA foi superior para ambos os meta-aprendizes, mas KNN obteve um ganho de acurácia maior em comparação com o RF. Porém, assim como na seleção de PU Learning, o método RF obteve os melhores resultados quando comparado aos obtidos pelo meta-aprendiz KNN. Sendo assim, como no primeiro estudo de caso, o meta-aprendiz RF parece ser mais robusto para lidar com exemplos ruidosos e redundantes no conjunto de treinamento. Conforme mencionado anteriormente, isso pode ocorrer devido às características do método RF, com seus componentes produzidos por *bagging*. Após a inclusão de 396 meta-exemplos sintéticos selecionados por NN-d, as taxas de acurácia dos meta-aprendizes RF e KNN foram 68,38% e 64,50%, respectivamente. A Tabela 8 apresenta um resumo da acurácia obtida pelos meta-aprendizes quando o método NN-d foi usado para selecionar os meta-exemplos sintéticos mais relevantes.

Figura 20 – Acurácia média obtida pela Random Forest usando NN-d para selecionar conjuntos de dados sintéticos. O conjunto de treinamento é inicializado com todos os conjuntos de dados originais, deixando apenas conjuntos de dados sintéticos para seleção. As linhas horizontais tracejadas representam a acurácia obtida usando todos os conjuntos de dados originais e sintéticos (parte superior) ou apenas os conjuntos de dados originais (parte inferior).



Fonte: Produzido pela autora (2022)

Figura 21 – Acurácia média obtida por KNN usando NN-d para selecionar conjuntos de dados sintéticos. O conjunto de treinamento é inicializado com todos os conjuntos de dados originais, deixando apenas conjuntos de dados sintéticos para seleção. As linhas horizontais tracejadas representam a acurácia obtida usando todos os conjuntos de dados originais e conjuntos de dados sintéticos (parte superior) ou apenas os conjuntos de dados originais (parte inferior).



Fonte: Produzido pela autora (2022)

Tabela 8 – Acurácia média (%) obtida por meta-alunos Random Forest e KNN e o número médio de exemplos no conjunto de treinamento que consiste em seleção NN-d e meta-exemplos originais.

Algoritmos/Dados treinamento	OD	OD + todos	OD + Seleção NN-d
RF	47.81	63.92	**68.38 (+4.46)**
KNN	45.60	54.77	**64.50 (+9.73)**
Número médio de exemplos	70	1066	**466**

Fonte: Produzido pela autora (2022)

5.3 DISCUSSÃO DOS RESULTADOS

Os resultados alcançados em nossos experimentos revelam algumas peculiaridades. Como já discutido, o sucesso do aprendizado da técnica PU Learning depende da utilização de um

determinado nível de ruído, na primeira etapa, que seja adequado aos dados fornecidos ao método. Em nosso primeiro estudo de caso, observamos que os níveis de ruído que mais favoreceram a seleção dos dados foram muito diferentes, considerando a acurácia dos meta-aprendizes, 13% e 28% para KNN e RF, respectivamente. Já no segundo estudo de caso, os melhores resultados dos meta-aprendizes foram observados para valores de níveis de ruído próximos, sendo 17% para o método RF e 18% para o método KNN. Essas diferenças são esperadas, uma vez que os conjuntos de dados sintéticos adotados em cada estudo de caso são distintos entre si, o que faz com que também apresentem características distintas. Uma análise mais profunda dessas diferenças será deixada para uma investigação futura. Sempre que se considera os resultados resumidos em ambos os estudos de caso (ver Tabelas 5 e 7) para PU Learning, é possível notar uma frequência no ganho de acurácia para ambos os meta-aprendizes no segundo estudo de caso. Este fato pode ser devido às características específicas dos conjuntos de dados deste estudo de caso 2.

Em relação à técnica NN-d, no estudo de caso 1, o ganho de acurácia entre os dois meta-aprendizes não apresentou diferença significativa nos valores. Por outro lado, no segundo estudo de caso, o método KNN obteve um ganho de acurácia maior do que o meta-aprendiz RF. Essa diferença provavelmente está relacionada ao fato de que o método KNN é um método mais sensível aos dados. Em contraste, o RF é um método mais robusto no que diz respeito à qualidade dos meta-exemplos de treinamento.

Nesta tese, desenvolvemos dois estudos de caso distintos para avaliar nossa proposta de seleção de dados de treinamento usando técnicas OCC. Adotamos repositórios diferentes para coleta de dados reais e medidas de complexidade também diferentes para cada estudo de caso. Além disso, os dados sintéticos gerados em cada estudo de caso foram produzidos a partir de técnicas de manipulação de dados diferentes. Como cada classificador se comporta melhor ou pior, dependendo das propriedades dos conjuntos de dados, como sensibilidade à sobreposição e desequilíbrio de classes, nesta investigação avaliamos a proposta de seleção de conjuntos de dados utilizando técnicas de OCC em dois cenários diferentes. Em ambos os estudos de caso, foi possível observar a capacidade da estratégia OCC para selecionar conjuntos de dados que são de fato relevantes para o conjunto de treinamento.

5.4 CONSIDERAÇÕES FINAIS

Este capítulo apresentou os resultados experimentais da avaliação da proposta de seleção de conjuntos de dados relevantes para MA adotando técnicas de OCC. Os resultados foram observados por meio de dois estudos de caso. Em ambos os estudo de caso foram realizadas análises dos resultados individualmente e análise comparativa na seção 5.3. Os resultados apresentados demonstram o sucesso da solução proposta nesta tese.

O próximo capítulo traz as conclusões do trabalho realizado. Além das contribuições, limitações e propostas de trabalhos futuros.

6 CONCLUSÕES

Nesta tese, propomos a combinação de técnicas de OCC e manipulação de dados para melhorar a qualidade de exemplos de treinamento para MA. O principal objetivo dessa abordagem é reduzir o custo computacional do processo de rotulagem dos meta-exemplos e melhorar o desempenho do MA, aumentando o número de meta-exemplos e removendo dados irrelevantes.

A proposta de seleção de conjuntos de dados sintéticos para MA combina técnicas de OCC e manipulação de dados, objetivando a redução do custo experimental no processo de rotulagem dos meta-exemplos e melhoria do desempenho do MA por meio do aumento da disponibilidade do número de conjuntos de dados relevantes para a criação de meta-exemplos de treinamento.

Nossos experimentos foram conduzidos em dois estudos de caso. Em ambos os estudos de caso, usamos os métodos PU Learning e NN-d para selecionar os conjuntos de dados mais relevantes. Esses métodos são representantes da técnica OCC. Os métodos meta-aprendizes adotados foram o KNN e RF.

Podemos concluir, pelos resultados observados, que é razoável aproveitar a grande quantidade de meta-exemplos fornecidos por conjuntos de dados sintéticos, sem aumento significativo de custos computacionais em experimentos. Além disso, foi possível observar que a eliminação de meta-exemplos irrelevantes/redundantes gerou ganho de desempenho. A melhoria no desempenho do MA foi percebida com as técnicas de OCC em comparação com as linhas de base adotadas (consulte a Seção 4.6).Um ganho de acurácia dos métodos KNN e RF foi observado em comparação com resultados anteriores obtidos em Prudêncio, Soares e Ludermir (2011) usando o KNN como meta-aprendiz. O mesmo pode ser observado quando comparamos os resultados observados no presente trabalho com os resultados observados em Sousa, Prudêncio e Ludermir (2016) usando Amostragem de Incerteza para selecionar meta-exemplos.

6.1 CONTRIBUIÇÕES

Os itens a seguir destacam as principais contribuições produzidas por esta tese.

- Propõe uma nova técnica para seleção de exemplos relevantes em contexto de MA;

- Demonstra que OCC é eficiente para selecionar meta-exemplos realizando dois estudos de caso com configurações de dados diferentes;

- Propõe uma nova técnica para manipulação de conjuntos de dados existentes;

Por fim, os resultados alcançados por esta tese foram disponibilizados em publicações em conferência e eventos com boa classificação no qualis da CAPES, além de ter resultados submetidos recentemente em periódicos, conforme a seguir:

- **Transfer learning for synthetic examples selection in meta-learning** (Regina Rosa Parente e Ricardo Bastos Cavalcante Prudêncio) - Publicado em Outubro de 2018 em **Anais do XV Encontro Nacional de Inteligência Artificial e Computacional. SBC**. (PARENTE; PRUDENCIO, 2018). Qualis B4.
- **One-Class Classification for Selecting Synthetic Datasets in Meta-Learning** (Regina Rosa Parente e Ricardo Bastos Cavalcante Prudêncio) - Apresentado na **International Joint Conference on Neural Networks (IJCNN)** em Junho de 2020. (PARENTE; PRUDÊNCIO, 2020). Qualis A1.
- **Selecting Synthetic Datasets for Meta-Learning by One-Class Clas-sification** (Regina Rosa Parente e Ricardo Bastos Cavalcante Prudêncio) - Submetido ao periódico **Knowledge-Based Systems** em Novembro de 2021. Qualis A1.

6.2 TRABALHOS FUTUROS

Por fim, abordamos os aspectos do nosso estudo que deixamos para uma investigação mais aprofundada em trabalhos futuros, a saber: (1) adoção de outros classificadores, como SVMs, que poderiam ser potencialmente úteis como meta-aprendizes; (2) investigar outras técnicas de

OCC, sempre que diferentes classificadores são usados como meta-aprendizes; (3) investigar métodos para estimar *priori* o nível de ruídos dos dados de treinamento. (4) investigar o impacto individual dos dados sintéticos no processo de aprendizagem.

REFERÊNCIAS

ABATI, D.; PORRELLO, A.; CALDERARA, S.; CUCCHIARA, R. Latent space autoregression for novelty detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2019.

ACOSTA, M. A. M.; VILLANOVA, L.; BAATAR, D.; SMITH-MILES, K. Instance spaces for machine learning classification. *Machine Learning*, v. 107, p. 109–147, 01 2018.

AHA, D. W. Generalizing from case studies: A case study. In: SLEEMAN, D.; EDWARDS, P. (Ed.). *Machine Learning Proceedings 1992*. San Francisco (CA): Morgan Kaufmann, 1992. p. 1–10. ISBN 978-1-55860-247-2. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9781558602472500061>.

BEKKER, J.; DAVIS, J. Learning from positive and unlabeled data: a survey. *Machine Learning*, p. 719–760, 2020.

BELLINGER, C.; SHARMA, S.; JAPKOWICZ, N. One-class classification – from theory to practice: A case-study in radioactive threat detection. *Expert Systems with Applications*, v. 108, p. 223 – 232, 2018. ISSN 0957-4174. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417418302926>.

BENSUSAN, H.; GIRAUD-CARRIER, C. Casa batlo is in passeig de gracia or landmarking the expertise space. *Proceedings of the ECML 2000 workshop on Meta-Learning: Building Automatic Advice Strategies for Model Selection and Method Combination, p. 2947. ECML 2000*, 2000.

BENSUSAN, H.; GIRAUD-CARRIER, C.; KENNEDY, C. A higher-order approach to meta-learning. *Proceedings of the ECML 2000 workshop on Meta-Learning: Building Automatic Advice Strategies for Model Selection and Method Combination, p. 109-117. ECML 2000.*, 2000.

BENSUSAN, H.; KALOUSIS, A. Estimating the predictive accuracy of a classifier. *EMCL 01: Proceedings of the 12th European Conference on Machine Learning, p. 25-36, London, UK. Springer-Verlag.*, 2001.

BLOCKEEL, H.; VANSCHOREN, J. Experiment databases: Towards an improved experimental methodology in machine learning. *Kok J.N., Koronacki J., Lopez de Mantaras R., Matwin S., Mladenič D., Skowron A. (eds) Knowledge Discovery in Databases: PKDD 2007. PKDD 2007. Lecture Notes in Computer Science, vol 4702. Springer, Berlin, Heidelberg*, 2007.

BRAZDIL, P.; GIRAUD-CARRIER, C. Metalearning and algorithm selection: progress, state of the art and introduction to the 2018 special issue. *Machine Learning*, v. 107, n. 1, p. 1–14, Jan 2018. ISSN 1573-0565. Disponível em: <https://doi.org/10.1007/s10994-017-5692-y>.

BRAZDIL, P.; GIRAUD-CARRIER, C.; SOARES, C.; VILALTA, R. *(Eds.), Metalearning - Applications to Data Mining*. [S.l.]: Springer, Berlin, Heidelberg, 2009.

BRAZDIL, P.; SOARES, C.; COSTA, J. P. da. Ranking learning algorithms: Using ibl and meta-learning on accuracy and time results. *Machine Learning*, v. 50(3), p. 251–277, 2003.

BRODLEY, C. E. Recursive automatic bias selection for classifier construction. *Mach. Learn., v.20, n.1-2, p.63-94.*, 1995.

CAMIÑA, J. B.; MEDINA-PEREZ, M. A.; MONROY, R.; LOYOLA-GONZALEZ, O.; VILLANUEVA, L. A. P.; GURROLA, L. C. G. Bagging-randomminer: a one-class classifier for file access-based masquerade detection. *Machine Vision and Applications*, v. 30, p. 959–974, July 2019. Disponível em: <https://doi.org/10.1007/s00138-018-0957-4>.

CAMPBELL, C.; BENNETT, K. P. *A Linear Programming Approach to Novelty Detection*. 2001.

CARUANA, R.; NICULESCU-MIZIL, A. An empirical comparison of supervised learning algorithms. *ICML 06: Proceedings of the 23rd international conference on Machine learning, p. 161-168, New York, NY, USA. ACM Press. Citado na página 52.*, 2006.

CAVALCANTI, G. D.; REN, T. I.; VALE, B. A. Data complexity measures and nearest neighbor classifiers: A practical analysis for meta-learning. In: *2012 IEEE 24th International Conference on Tools with Artificial Intelligence*. [S.l.: s.n.], 2012. v. 1, p. 1065–1069.

CHEN, J.; SATHE, S.; AGGARWAL, C.; TURAGA, D. Outlier detection with autoencoder ensembles. In: *Proceedings of the 2017 SIAM International Conference on Data Mining (SDM)*. [S.l.: s.n.], 2017. p. 9.

CLIFTON, L.; CLIFTON, D. A.; WATKINSON, P. J.; TARASSENKO, L. Identification of patient deterioration in vital-sign data using one-class support vector machines. In: *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)*. [S.l.: s.n.], 2011. p. 125–131.

COHN, D.; ATLAS, L.; LADNER, R. Improving generalization with active learning. *Machine Learning*, n. 15, p. 201–221, 1994.

CUNHA, T.; SOARES, C.; de Carvalho, A. C. Metalearning and recommender systems: A literature review and empirical study on the algorithm selection problem for collaborative filtering. *Information Sciences*, v. 423, p. 128–144, 2018. ISSN 0020-0255. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0020025517309702>.

DAI, H.; CAO, J.; WANG, T.; DENG, M.; YANG, Z. Multilayer one-class extreme learning machine. *Neural Networks*, 2019. ISSN 0893-6080. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0893608019300772>.

DUDA, R. O.; HART, P. E. Pattern classification and scene analysis. *Published by John Wiley - Sons, New York*, 1973.

DéSIR, C.; BERNARD, S.; PETITJEAN, C.; HEUTTE, L. A new random forest method for oneclass classification. *IAPR International Workshop on Statistical Techniques in Pattern Recognition (SPR), Hiroshima, Japan, Vol. 7626 of LNCS, pp. 282–290*, 2012.

DéSIR, C.; BERNARD, S.; PETITJEAN, C.; HEUTTE, L. A random forest based approach for one-class classification in medical imaging. *3rd MICCAI International Workshop on Machine Learning in Medical Imaging (MLMI), Nice, France 7588. pp. 250–257*, 2012.

DéSIR, C.; BERNARD, S.; PETITJEAN, C.; HEUTTE, L. *One Class Random Forests*. 2013.

FILCHENKOV, A.; PENDRYAK, A. Datasets meta-feature description for recommending feature selection algorithm. In: *2015 Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference (AINL-ISMW FRUCT)*. [S.l.: s.n.], 2015. p. 11–18.

GAO, L.; ZHANG, L.; LIU, C.; WU, S. Handling imbalanced medical image data: A deep-learning-based one-class classification approach. *Artificial Intelligence in Medicine*, v. 108, p. 101935, 2020. ISSN 0933-3657. Disponível em: <https://www.sciencedirect.com/science/article/pii/S093336572030261X>.

GARCIA, L. P. F.; CARVALHO, A. C. P. de Leon Ferreira de; LORENA, A. C. Noise detection in the meta-learning level. *Neurocomputing*, v. 176, p. 14–25, 2016. ISSN 0925-2312. Recent Advancements in Hybrid Artificial Intelligence Systems and its Application to Real-World Problems. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231215005482>.

GARCIA, L. P. F.; LORENA, A. C.; LEHMANN, J. Ecol: Complexity measures for classification problems. *https://CRAN.Rproject.org/package=ECoL*, 2019.

GARDNER, A. B.; KRIEGER, A. M.; VACHTSEVANOS, G.; LITT, B. One-class novelty detection for seizure analysis from intracranial eeg. In: . [S.l.: s.n.], 2006. p. 1025–1044.

GARDNER, A. B.; KRIEGER, A. M.; VACHTSEVANOS, G.; LITT, B. One-class novelty detection for seizure analysis from intracranial eeg. *Journal of Machine Learning Research*, p. 1025–1044, 2006.

GUERBAI, Y.; CHIBANI, Y.; HADJADJI, B. The effective use of the one-class svm classifier for handwritten signature verification based on writer-independent parameters. *Pattern Recognition*, v. 48, n. 1, p. 103–113, 2015. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320314002751>.

GUO, L.; ZHAO, L.; WU, Y.; LI, Y.; XU, G.; YAN, Q. Tumor detection in mr images using one-class immune feature weighted svms. *IEEE Transactions on Magnetics*, v. 47, n. 10, p. 3849–3852, 2011.

HANSEN, N. The cma evolution strategy: A comparing review. In: *Lozano J.A., Larrañaga P., Inza I., Bengoetxea E. (eds) Towards a New Evolutionary Computation. Studies in Fuzziness and Soft Computing, vol 192. Springer, Berlin, Heidelberg*. [s.n.], 2006. Disponível em: <https://doi.org/10.1007/3-540-32494-1_4>.

HEMPSTALK, K.; FRANK, E.; WITTEN, I. H. One-class classification by combining density and class probability estimation. In: DAELEMANS, W.; GOETHALS, B.; MORIK, K. (Ed.). *Machine Learning and Knowledge Discovery in Databases*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. p. 505–519.

HILARIO, M.; KALOUSIS, A. Quantifying the resilience of inductive classification algorithms. In: ZIGHED, D. A.; KOMOROWSKI, J.; ŻYTKOW, J. (Ed.). *Principles of Data Mining and Knowledge Discovery*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000. p. 106–115. ISBN 978-3-540-45372-7.

HO, T. K.; BASU, M. Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 24, n. 3, p. 289–300, 2002.

HO, T. K.; BASU, M.; LAW, M. H. C. Measures of geometrical complexity in classification problems. *Basu M., Ho T.K. (eds) Data Complexity in Pattern Recognition. Advanced Information and Knowledge Processing. Springer, London*, 2006.

I.H.WITTEN; FRANK, E. *Data Mining: Practical machine learning tools and techniques*. [S.l.]: Morgan Kaufmann, 2005.

KALOUSIS, A. *Algorithm Selection via Meta-Learning*. Tese (Doutorado) — Centre Universiteire d'Informatique, Université de Genève, Geneva, Suíça. Citado nas páginas 39, 43, 53, 66, 68, e 79., ago. 2002.

KALOUSIS, A.; HILARIO, M. Feature selection for meta-learning. In: SPRINGER. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. [S.l.], 2001. p. 222–233.

KANDA, J.; CARVALHO, A. C. P. de Leon Ferreira de; HRUSCHKA, E. R.; SOARES, C. Using meta-learning to recommend meta-heuristics for the traveling salesman problem. *2011 10th International Conference on Machine Learning and Applications and Workshops*, v. 1, p. 346–351, 2011.

KHAN, S.; MADDEN, M. A survey of recent trends in one class classification. In: COYLE, L.; FREYNE, J. (Ed.). *(eds) Artificial Intelligence and Cognitive Science.AICS 2009. Lecture Notes in Computer Science, vol 6206. Springer, Berlin, Heidelberg*. [S.l.]: Publishing Press, 2010. p. 188–197.

KHAN, S. S.; MADDEN, M. G. One-class classification: taxonomy of study and review of techniques. *The Knowledge Engineering Review*, Cambridge University Press, v. 29, n. 3, p. p. 345–374, 2014.

KOEPF, C.; TAYLOR, C. C.; KELLER, J. Meta-analysis: From data characterisation for meta-learning to meta-regression. *Brazdil, P.; Jorge, A., editores, Proceedings of the PKDD-00 Workshop on Data Mining, Decision Support,Meta-Learning and ILP: Forum for Practical Problem Presentation and Prospective Solutions, Lyon, France.*, 2000.

KRAWCZYK, B.; GALAR, M.; WOZNIAK, M.; BUSTINCE, H.; HERRERA, F. Dynamic ensemble selection for multi-class classification with one-class classifiers. *Pattern Recognition*, v. 83, p. 34 – 51, 2018. ISSN 0031-3203. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0031320318301857>.

KRAWCZYK, B.; WOZNIAK, M.; CYGANEK, B. Clustering-based ensembles for one-class classification. *Information Sciences*, v. 264, p. 182–195, 2014. ISSN 0020-0255. Serious Games. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0020025513008694>.

LAI, C.; ZOU, D.; LERMAN, G. Robust subspace recovery layer for unsupervised anomaly detection. *CoRR*, abs/1904.00152, 2019. Disponível em: <http://arxiv.org/abs/1904.00152>.

LI, X.; LIU, B. Learning to classify text using positive and unlabeled data. In: *In: Proceedings of the 19th international joint conference on artificial intelligence*. [S.l.: s.n.], 2003. p. 587–594.

LI, X. li. *Learning to Identify Unexpected Instances in the Test Set*. 2007.

LIANG, L.; ZHONG, J.; LIU, J.; LI, P.; ZHAN, C.; MENG, Z. An implementation of synthetic generation of wind data series. *2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT)*, p. 1–6, 2013.

LICHMAN, M. Uci machine learning repository. url http://archive.ics.uci. edu/ml. 2013.

LIU, B.; DAI, Y.; LI, X.; LEE, W. S.; YU, P. S. Building text classifiers using positive and unlabeled examples. *Third IEEE International Conference on Data Mining*, p. 179–186, 2003.

LIU, B.; LEE, W. S.; YU, P. S.; LI, X. Partially supervised classification of text documents. *ICML-02*, 2002.

LORENA, A. C.; GARCIA, L. P. F.; LEHMANN, J.; SOUTO, M. C. P.; HO, T. K. How complex is your classification problem? a survey on measuring classification complexity. *arXiv Preprint arXiv:18080359*, arXiv Preprint arXiv:180803591, 2018.

LORENA, A. C.; MACIEL, A. I.; MIRANDA, P. B. C. de; COSTA, I. G.; PRUDÊNCIO, R. B. C. Data complexity meta-features for regression problems. *Mach Learn*, v. 107, p. 209–246, 2018.

MACIÀ, N.; BERNADÓ-MANSILLA, E. Towards uci+: A mindful repository design. *Information Sciences.*, v. 261, p. 237–262, 2014.

MACIÀ, N.; ORRIOLS-PUIG, A.; BERNADÓ-MANSILLA, E. Genetic-based synthetic data sets for the analysis of classifiers behavior. *Proceedings of 15th International Conference on Hybrid Intelligent Systems, pp. 507-512*, 2008.

MAUCERI, S.; SMITH, L.; SWEENEY, J.; MCDERMOTT, J. Subject recognition using wrist-worn triaxial accelerometer data. *Nicosia G., Pardalos P., Giuffrida G., Umeton R. (eds) Machine Learning, Optimization, and Big Data. MOD 2017. Lecture Notes in Computer Science, vol 10710. Springer, Cham*, 2017. Disponível em: <https://doi.org/10.1007/978-3-319-72926-8_48>.

MAZHELIS, O. One-Class Classifiers: A Review and Analysis of Suitability in the Context of Mobile-Masquerader Detection. *Revue Africaine de la Recherche en Informatique et Mathématiques Appliquées*, Volume 6, april 2007, joint Special Issue ARIMA/SACJ on Advances in end-user data mining techniques, set. 2007. Disponível em: <https://arima.episciences.org/1877>.

MELO, V. V. de; LORENA, A. C. Using complexity measures to evolve synthetic classification datasets. In: *2018 International Joint Conference on Neural Networks (IJCNN)*. [S.l.: s.n.], 2018. p. 1–8.

MICHIE, D.; SPIEGELHALTER, D. J.; TAYLOR, C. C.; CAMPBELL J., e. Machine learning, neural and statistical classification. *Ellis Horwood, Upper Saddle River, NJ, USA. Citado nas páginas 4 e 42.*, 1994.

MIRANDA, P.; PRUDÊNCIO, R.; CARVALHO, A. de; SOARES, C. A hybrid meta-learning architecture for multi-objective optimization of svm parameters. *Neurocomputing*, v. 143, p. 27–43, 11 2014.

MIRANDA, P. B.; PRUDÊNCIO, R. B.; PAPPA, G. L. H3ad: A hybrid hyper-heuristic for algorithm design. *Information Sciences*, v. 414, p. 340 – 354, 2017. ISSN 0020-0255. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0020025516314323>.

MITCHELL, T. *Machine Learning*. [S.l.]: McGraw-Hill Science/Engineering/Math. Citado nas páginas 4, 23, 25, 26, 29, 32, e 39., 1997.

MOLLINEDA, R. A.; SÁNCHEZ, J. S.; SOTOCA, J. Data characterization for effective prototype selection. *Marques J.S., Pérez de la Blanca N., Pina P. (eds) Pattern Recognition and Image Analysis. IbPRIA 2005. Lecture Notes in Computer Science, vol 3523. Springer, Berlin, Heidelberg*, 2005.

MOLLINEDA, R. A.; SÁNCHEZ, J. S.; SOTOCA, J. M. A meta-learning framework for pattern classification by means of data complexity measures. *Inteligencia artificial*, v. 10, p. 31–38, 2006.

MORENO-TORRES, J. G.; RAEDER, T.; ALAIZ-RODRíGUEZ, R.; CHAWLA, N. V.; HERRERA, F. A unifying view on dataset shift in classification. *Pattern Recognition*, v. 45, n. 1, p. 521 – 530, 2012. ISSN 0031-3203. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0031320311002901>.

NAGY., T.; FARKAS., R.; CSIRIK., J. On positive and unlabeled learning for text classification. *Habernal I., Matoušek V. (eds) Text, Speech and Dialogue. TSD 2011. Lecture Notes in Computer Science, vol 6836. Springer, Berlin, Heidelberg*, 2011.

NASCIMENTO, A.; PRUDÊNCIO, R.; SOUTO, M.; COSTA, I. Mining rules for the automatic selection process of clustering methods applied to cancer gene expression data. *Lecture Notes in Computer Science, vol. 5769, pp. 20–29*, 2009.

ORRIOLS-PUIG, A.; MACIÀ, N.; HO., T. K. Documentation for the data complexity library in c++. *Technical report, La Salle - Universitat Ramon Llull*, 2010.

PARENTE, R. R.; CANUTO, A. M. P.; XAVIER-JUNIOR, J. C. Characterization measures of ensemble systems using a meta-learning approach. *Proceedings of International Joint Conference on Neural Networks, Dallas, Texas, USA, August 4-9*, 2013.

PARENTE, R. R.; PRUDENCIO, R. B. Transfer learning for synthetic examples selection in meta-learning. In: SBC. *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional*. [S.l.], 2018. p. 811–822.

PARENTE, R. R.; PRUDÊNCIO, R. B. One-class classification for selecting synthetic datasets in meta-learning. In: IEEE. *2020 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2020. p. 1–8.

PARMEZAN, A. R. S.; LEE, H. D.; WU, F. C. Metalearning for choosing feature selection algorithms in data mining: Proposal of a new framework. *Expert Systems with Applications*, v. 75, p. 1–24, 2017. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417417300222>.

PERDISCI, R.; GU, G. Using an ensemble of one-class svm classifiers to harden payload-based anomaly detection systems. In: *In Proceedings of the IEEE International Conference on Data Mining (ICDM'06*. [S.l.]: IEEE Computer Society, 2006. p. 488–498.

PEREIRA, G. T.; SANTOS, M. R. d.; CARVALHO, A. C. P. d. L. F. de. Evaluating meta-feature selection for the algorithm recommendation problem. *arXiv preprint arXiv:2106.03954*, 2021.

PEREIRA-SANTOS, D.; PRUDÊNCIO, R. B. C.; CARVALHO, A. C. P. de Leon Ferreira de. Empirical investigation of active learning strategies. *Neurocomputing*, v. 326-327, p. 15–27, 2019. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231217315357>.

PERERA, P.; NALLAPATI, R.; XIANG, B. Ocgan: One-class novelty detection using gans with constrained latent representations. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2019.

PERERA, P.; OZA, P.; PATEL, V. M. One-class classification: A survey. *CoRR*, abs/2101.03064, 2021. Disponível em: <https://arxiv.org/abs/2101.03064>.

PERERA, V. M. P. P. Learning deep features for one-class classification. *arXiv:1801.05365v1 [cs.CV]*, 2018.

PFAHRINGER, B.; BENSUSAN, H.; GIRAUD-CARRIER, C. Metalearning by landmarking various learning algorithms. *In Proceedingsof the Seventeenth International Conference on Machine Learning, 743-750.*, 2000.

PHUA, C.; LEE, V.; SMITH, K.; GAYLER, R. A comprehensive survey of data mining-based fraud detection research. 2010.

PIDHORSKYI, S.; ALMOHSEN, R.; ADJEROH, D. A.; DORETTO, G. Generative probabilistic novelty detection with adversarial autoencoders. *CoRR*, abs/1807.02588, 2018. Disponível em: <http://arxiv.org/abs/1807.02588>.

PIMENTEL, B. A.; de Carvalho, A. C. A new data characterization for selecting clustering algorithms using meta-learning. *Information Sciences*, v. 477, p. 203–219, 2019. ISSN 0020-0255. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0020025518308624>.

PLESSIS, M. C. du; NIU, G.; SUGIYAMA, M. Convex formulation for learning from positive and unlabeled data. In: . [S.l.: s.n.], 2015. v. 37, p. 1386 – 1394.

PRUDÊNCIO, R.; LUDERMIR, T. Meta-learning approaches to selecting time series models. *Neurocomputing 61, 121–137*, 2004.

PRUDÊNCIO, R.; SOARES, C.; LUDERMIR, T. Combining meta-learning and active selection of datasetoids for algorithm selection. *Corchado E., Kurzyński M., Woźniak M. (eds) Hybrid Artificial Intelligent Systems. HAIS 2011. Lecture Notes in Computer Science, vol 6678. Springer, Berlin, Heidelberg*, 2011.

PRUDÊNCIO, R.; SOARES, C.; LUDERMIR, T. Uncertainty sampling methods for selecting datasets in active meta-learning. *in: Proceedings of the 2011 International Joint Conference on Neural Networks*, p. 1082–1089, 2011.

PRUDÊNCIO, R. B. C.; SOARES, C.; LUDERMIR, T. B. Active selection of datasetoids for meta-learning. *in Proceedings of the 3rd Planning to Learn - ECAI Workshop*, 2010.

REIF, M.; SHAFAIT, F.; DENGEL, A. Dataset generation for meta-learning. *Poster and Demo Track of the 35th German Conference on Artificial Intelligence, pp. 69–73*, 2012.

RICE, J. R. The algorithm selection problem. *Advances in Computers, v.15, p.65-118.*., 1976.

RIDDER, D. de; TAX, D. M. J.; DUIN, R. P. W. An experimental comparison of one-class classification methods. *Proceedings of the 4th Annual Conference of the Advacned School for Computing and Imaging, Delft*, 1998.

ROSA-RIVERA, F. de la; NUNEZ-VARELA, J. I.; ORTIZ-BAYLISS, J. C.; TERASHIMA-MARÌN, H. Algorithm selection for solving educational timetabling problems. *Expert Systems with Applications*, v. 174, p. 114694, 2021. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417421001354>.

ROSSI, A. L. D.; CARVALHO, A. C. P. de Leon Ferreira de; SOARES, C.; SOUZA, B. F. de. Metastream: A meta-learning based method for periodic algorithm selection in time-changing data. *Neurocomputing*, v. 127, p. 52–64, 2014. ISSN 0925-2312. Advances in Intelligent Systems. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231213007819>.

SABOKROU, M.; KHALOOEI, M.; FATHY, M.; ADELI, E. Adversarially learned one-class classifier for novelty detection. *arXiv:1802.09088v2 [cs.CV]*, 2018.

SCHLEGL, T.; SEEBöCK, P.; WALDSTEIN, S. M.; SCHMIDT-ERFURTH, U.; LANGS, G. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. *Niethammer M. et al. (eds) Information Processing in Medical Imaging. IPMI 2017. Lecture Notes in Computer Science, vol 10265. Springer, Cham*, 2017. Disponível em: <https://doi.org/10.1007/978-3-319-59050-9_12>.

SCHMIDHUBER, J. Z.; WIERING, M. Simple principles of meta-learning. *Technical Report IDSIA-69-96*, 1996.

SCHôLKOPF, B.; WILLIAMSON, R.; SMOLA, A.; SHAWE-TAYLOR, J.; PLATT, J. *Support Vector Method for Novelty Detection*. 2000.

SERBAN, F.; VANSCHOREN, J.; KIETZ, J.-U.; BERNSTEIN, A. A survey of intelligent assistants for data analysis. *ACM Computing Surveys 45(3)*, n. Art.31, 2013.

SETTLES, B. Active learning literature survey. *Computer Sciences Technical Report, vol. 1648, University of Wisconsin-Madison*, 2009.

SMITH-MILES, K. Cross-disciplinary perspectives on meta-learning for algorithm selection. *ACM Comput. Surv.*, v. 41, p. 6:1–6:25, 2008.

SMITH-MILES, K. A. Cross-disciplinary perspectives on meta-learning for algorithm selection. *ACM Computing Survey*, v. 41, p. 1–25, 2009.

SOARES, C. *Learning Rankings of Learning Algorithms: Recommendation of Algorithms with Meta-Learning*. Tese (Doutorado) — Departamento de Ciência da Computação, Faculdade de Ciências da Universidade do Porto, Porto, Portugal, ago. 2004.

SOARES, C. Uci++, improved support for algorithm selection using datasetoids. In: COYLE, L.; FREYNE, J. (Ed.). [S.l.]: Lecture Notes in Computer Science 5476, 2009. p. 499–506.

SOARES, C.; BRAZDIL, P. Zoomed ranking: Selection of classification algorithms based on relevant performance information. *PKDD 00: Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery, p. 126-135, London, UK. Springer-Verlag. Citado na página 52.*, 2000.

SOARES, C.; BRAZDIL, P. B.; KUBA, P. A meta-learning method to select the kernel width in support vector regression. 2004.

SOUSA, A.; PRUDÊNCIO, R.; LUDERMIR, C. S. T. Active learning and data manipulation techniques for generating training examples in meta-learning. *Neurocomputing, 194 (2016), pp. 45-55*, 2016.

SOUSA, A. F. M.; PRUDÊNCIO, R. B. C.; SOARES, C.; Ludermir, T. B. Active selection of training instances for a random forest meta-learner. In: *The 2013 International Joint Conference on Neural Networks (IJCNN)*. [S.l.: s.n.], 2013. p. 1–7. ISSN 2161-4407.

SOUZA, B.; SOARES, C. Meta-learning approach to gene expression data classification. *J. Intell. Comput. Cybern. 2 (2), 285–303*, 2009.

SOUZA, B. F. de. *Meta-aprendizagem aplicada à classificação de dados de expressão gênica*. Tese (Doutorado) — USP-São Carlos, ago. 2010.

SWERSKY, L.; MARQUES, H. O.; SANDER, J.; CAMPELLO, R. J.; ZIMEK, A. On the evaluation of outlier detection and one-class classification methods. In: *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. [S.l.: s.n.], 2016. p. 1–10.

TAX, D. One-class classification: Concept learning in the absence of counter-examples. *Ph.D. thesis. Delft University of Technology*, 2001.

TAX, D. M.; DUIN, R. P. Combining one-class classifiers. In: *in Proc. Multiple Classifier Systems, 2001*. [S.l.]: Springer Verlag, 2001. p. 299–308.

TAX, D. M. J.; DUIN, R. Uniform object generation for optimizing one-class classifiers. *Machine Learning Research (2001) 155-173*, 2001.

TAX, D. M. J.; DUIN, R. P. W. Data domain description using support vectors. In: *Proceedings of the European Symposium on Artificial Neural Networks*. [S.l.: s.n.], 1999. p. 251–256.

TAX, D. M. J.; DUIN, R. P. W. Support vector domain description. *Pattern Recognition Letters*, v. 20, p. 1191–1199, 1999.

TSAI, C.-F.; LIN, W.-C. Feature selection and ensemble learning techniques in one-class classifiers: An empirical study of two-class imbalanced datasets. *IEEE Access*, v. 9, p. 13717–13726, 2021.

VANSCHOREN, J. Meta-learning: A survey. *CoRR*, abs/1810.03548, 2018. Disponível em: <http://arxiv.org/abs/1810.03548>.

VANSCHOREN, J.; RIJN, J. N. van; BISCHL, B.; TORGO, L. Openml: Networked science in machine learning. *SIGKDD Explorations*, ACM, New York, NY, USA, v. 15, n. 2, p. 49–60, 2013. Disponível em: <http://doi.acm.org/10.1145/2641190.2641198>.

VILALTA, R.; DRISSI, Y. A perspective view and survey of metalearning. *Journal of Artificial Intelligence Review. p. 77-95*, 2002.

VILALTA, R.; GIRAUD-CARRIER, C.; BRAZDIL, P. *Data Mining and Knowledge*. [S.l.]: Discovery Handbook: A Complete Guide for Practitioners and Researchers, chapter Meta-Learning: Concepts and Techniques, p. 1-17. Kluwer Academic Publishers., 2005.

VILLA-PEREZ, M. E.; ALVAREZ-CARMONA, M. A.; LOYOLA-GONZALEZ, O.; MEDINA-PEREZ, M. A.; VELAZCO-ROSSELL, J. C.; CHOO, K.-K. R. Semi-supervised anomaly detection algorithms: A comparative summary and future research directions. *Knowledge-Based Systems*, v. 218, p. 106878, 2021. ISSN 0950-7051. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705121001416>.

WANG, H.; ZHANG, Q.; WU, J.; PAN, S.; CHEN, Y. Time series feature learning with labeled and unlabeled data. *Pattern Recognition*, v. 89, p. 55–66, 2019. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320318304473>.

WEI, Q.; REN, Y.; HOU, R.; SHI, B.; LO, J. Y.; CARIN, L. *Anomaly detection for medical images based on a one-class classification*. 2018. 10575 - 10575 - 6 p. Disponível em: <https://doi.org/10.1117/12.2293408>.

WEN, J.; GREINER, R.; SCHUURMANS, D. Correcting covariate shift with the frank-wolfe algorithm. In: *IJCAI*. [S.l.: s.n.], 2015.

XIA, R.; HU, X.; LU, J.; YANG, J.; ZONG, C. Instance selection and instance weighting for cross-domain sentiment classification via pu learning. *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

XU, Y.; XU, C.; XU, C.; TAO, D. Multi-positive and unlabeled learning. In: . [S.l.: s.n.], 2017. p. 3182–3188.

YOU, C.; ROBINSON, D. P.; VIDAL, R. Provable self-representation based outlier detection in a union of subspaces. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2017.

YU, H.; HAN, J. *PEBL: Positive Example Based Learning for Web Page Classification Using SVM*. 2002.

ZENG, Z.; FU, Y.; ROISMAN, G. I.; WEN, Z.; HU, Y.; HUANG, T. S. *One-class classification on spontaneous facial expressions, Automatic Face and Gesture Recognition*. 2006.

APÊNDICE A – PROBLEMAS DE CLASSIFICAÇÃO

A.1 DESCRIÇÃO DAS BASES DE DADOS

Tabela 9 – Descrição das Bases de Dados

Id	Nome	Descrição
1	ada-agnostic	A tarefa da ADA é descobrir pessoas de alta renda a partir de dados do censo
2	analcatdata-authorship	Uma coleção de conjuntos de dados usados no livro "Analyzing Categorical Data", de Jeffrey S. Simonoff, Springer-Verlag, Nova York, 2003
3	appendicitis	Uma cópia do conjunto de dados proposto em: SM Weiss e CA Kulikowski, Computer Systems That Learn (1991)
4	ar4	Busca identificar a presença de defeitos em códigos de *software*
5	Australian	Este é o conjunto de dados *Australian Credit Approval*, originário do projeto StatLog. Trata-se de pedidos de cartão de crédito.
6	autoUniv-au1-1000	busca avaliar tarefas de classificação
7	balance-scale	Escala de equilíbrio de peso e distância
8	banana	buscar identicar dados de *clusters* com formato de banana
9	banknote-authentication	Conjunto de dados sobre a distinção de notas genuínas e falsificadas

Continua na próxima página

Tabela 9 – continuação da página anterior

Id	Nome	Descrição
10	blood-transfusion-service-center	Conjunto de dados do centro de serviço de transfusão de sangue. Dados retirados do Centro de Serviços de Transfusão de Sangue na cidade de Hsin-Chu, em Taiwan
11	breast-tissue	Busca identificar a presença de tumores em mamas, que pode ser benignos ou malignos
12	cardiotocography	Uma versão de 3 classes do conjunto de dados Cardiotocography
13	churn	busca relacionar as características dos recursos e uso da conta de telefonia e se o cliente mudou ou não
14	climate-model-simulation-crashes	Este conjunto de dados contém registros de falhas de simulação encontradas durante conjuntos de quantificação de incerteza do modelo climático (UQ)
15	datatrieve	Este é um conjunto de dados do Repositório de Engenharia de Software PROMISE disponibilizado publicamente para incentivar modelos preditivos de engenharia de software repetíveis, verificáveis, refutáveis e/ou melhoráveis
16	diabetes	A partir de medidas fisiológicas e e testes clínicos, diagnostica a presença ou não de diabetes
17	diggle-table-a2	Conjuntos de Diggle, P.J. (1990). SÉRIE TEMPORAL: Uma introdução a Bio Estatística. Imprensa da Universidade de Oxford

Continua na próxima página

Tabela 9 – continuação da página anterior

Id	Nome	Descrição
18	ecoli	Avalia oito tipos diferentes de proteínas
19	fertility	Busca prever a quantidade e a qualidade dos espermatozoides
20	first-order-theorem-proving	Os atributos são uma mistura de características estáticas e dinâmicas derivadas de teoremas a serem provados
21	forest	Busca prever a ecorrência de incêndios em florestas
22	glass	Identifica o tipo de vidro a partir de características físicas da amostra
23	hayes-roth	Busca estudar algumas características em seres humanos, como *hobby*, idade, escolaridade e estado civil
24	heart-h	Diagnostica a presença ou ausência de risco de doença coronária a partir dos resultados de vários testes clíínicos.
25	heart-long-beach	Diagnostica a presença ou ausência de risco de doença coronária a partir dos resultados de vários testes clíínicos.
26	heart-statlog	Diagnostica a presença ou ausência de risco de doença coronária a partir dos resultados de vários testes clíínicos.
27	hill-valley	avalia funcoes se sao colinas ou vales, com formato suave. Usualmente usado para treinamento

Continua na próxima página

Tabela 9 – continuação da página anterior

Id	Nome	Descrição
28	ionosphere	Classifica sinais de radar provenientes da ionosfera
29	iris	Classifica a planta íris a partir do formato da sua folha
30	jEdit-4.0-4.2	Avalia cotagem de bugs em código fonte
31	jEdit-4.2-4.3	Avalia cotagem de bugs em código fonte
32	kc1-binary	Avalia a qualidade de modelos preditivos em Engenharia de Software
33	kc3	Avalia a qualidade de modelos preditivos em Engenharia de Software
34	LED-display-domain-7digit	Avalia as possibilidades LED representando os possíveis dígitos exibidos no display
35	mc2	Conjunto de dados de previsão de defeitos de software
36	mfeat-fourier	Este conjunto de dados consiste de características de numerais manuscritos $(0-9)$. 76 coeficientes de Fourier das formas de caracteres
37	mfeat-karhunen	Este conjunto de dados consiste de características de numerais manuscritos $(0-9)$. 64 *Karhunen-Love coeficient*s
38	mfeat-morphological	Este conjunto de dados consiste de características de numerais manuscritos $(0-9)$. seis características morfológicas

Continua na próxima página

Tabela 9 – continuação da página anterior

Id	Nome	Descrição
39	mfeat-zernike	Este conjunto de dados consiste de características de numerais manuscritos $(0-9)$. 47 momentos *Zernike*
40	MiceProtein	O conjunto de dados consiste nos níveis de expressão de 77 proteínas/modificações de proteínas que produziram sinais detectáveis na fração nuclear do córtex
41	oil-spill	Avalia a possibilidade de derramamento de óleo
42	optdigits-RemoveFolds	reconhecimento óptico de dígitos manuscritos
43	ozone-level-8hr	Avalida dados de nível de ozônio no solo
44	page-blocks	O problema consiste em classificar todos os blocos do layout da página de um documento que foi detectado por um processo de segmentação
45	parkinsons	Conjunto de dados de detecção da doença de Parkinson de Oxford
46	pc1	Conjunto de dados de previsão de defeitos de software
47	pc2	Conjunto de dados de previsão de defeitos de software
48	pc3	Conjunto de dados de previsão de defeitos de software
49	pc4	Conjunto de dados de previsão de defeitos de software

Continua na próxima página

Tabela 9 – continuação da página anterior

Id	Nome	Descrição
50	phpcSeK3V	Avalia anomalias em padrões de codificação
51	planning-relax	O conjunto de dados diz respeito à classificação de estágios mentais a partir de sinais de EEG
52	prnn-synth	Conjunto de dados sobre vírus que afetam diversas culturas
53	qsar-biodeg	Avalia a classificação de produtos químicos
54	ringnorm	Este conjunto de dados é uma implementação do exemplo de ringnorm de Leo Breiman
55	rmftsa-sleepdata	Avalia as medições do estado de sono de recém-nascidos
56	robot-failures-lp4	Contém medições de força e torque em um robô após a detecção de falha
57	robot-failures-lp5	Contém medições de força e torque em um robô após a detecção de falha
58	satellite-image	Consiste nos valores multiespectrais de pixels em bairros 3x3 em uma imagem de satélite e na classificação associada ao pixel central em cada bairro
59	seeds	Consiste em dados de análise de diferentes tipos de grãos de trigo
60	segment	Analisa pixels de imagens tiradas ao ar livre classificando os tipos de segmentos
61	semeion	Consiste em dados de dígitos manuscritos

Continua na próxima página

Tabela 9 – continuação da página anterior

Id	Nome	Descrição
62	sonar	Distingui entre materiais metálicos ou rochosos de acordo com os níveis de intensidade de sinais de sonar enviados sob diferentes condições
63	spambase	Coleção de e-mails de spam e-mails não-spam
64	spectf	O conjunto de dados descreve o diagnóstico de imagens cardíacas de tomografia computadorizada por emissão de próton único (SPECT)
65	sick	Classifica a presença ou ausência de tireoidiana
66	twonorm	O conjunto de dados é uma implementação do exemplo de duas normas de Leo Breiman
67	user-knowledge	Avalia o conhecimentos de usuários
68	vehicle	Classifica uma dada silhueta 2D dentre quatro tipos de veículos usando um conjunto de atributos físicos
69		Conjunto de dados contendo valores de características biomecânicas usadas para classificar pacientes ortopédicos
70	volcanoes-a1	Dados usados no mapeamento global da superfície de Vênus
71	volcanoes-a2	Dados usados no mapeamento global da superfície de Vênus
72	volcanoes-a3	Dados usados no mapeamento global da superfície de Vênus

Continua na próxima página

Tabela 9 – continuação da página anterior

Id	Nome	Descrição
73	wall-robot-navigation	Os dados foram coletados à medida que o robô SCITOS G5 navega pela sala seguindo a parede no sentido horário, por 4 rodadas, usando 24 sensores de ultrassom dispostos circularmente ao redor de sua 'cintura'
74	wall-robot-navigation-5	Os dados foram coletados à medida que o robô SCITOS G5 navega pela sala seguindo a parede no sentido horário, por 4 rodadas, usando 24 sensores de ultrassom dispostos circularmente ao redor de sua 'cintura'
75	wdbc	descrevem características dos núcleos celulares presentes na imagem de nódulos de mamários
76	wilt	Uma abordagem de nitidez híbrida e análise de imagem baseada em objetos multiescala para mapear pinheiros e carvalhos doentes
77	wine	Conjunto de dados de reconhecimento de vinho
78	yeast	Conjunto de dados de leveduras

Fonte: Produzido pela autora (2022)

A.2 CARACTERÍSTICAS DAS BASES DE DADOS

Tabela 10 – Características das Bases de Dados

Id	**Nome**	N^o**. Atributos**	N^o**. Padrões**	N^o**. Classes**
1	ada-agnostic	48	4562	2
2	analcatdata-authorship	71	841	5
3	appendicitis	8	106	2
4	ar4	30	107	2
5	Australian	15	690	2
6	autoUniv-au1-1000	21	1000	2
7	balance-scale	5	625	3
8	banana	3	5300	2
9	banknote-authentication	5	1372	2
10	blood-transfusion-service-center	5	748	2
11	breast-tissue	10	106	4
12	cardiotocography	36	2126	10
13	churn	17	5000	2
14	climate-model-simulation-crashes	21	540	2
15	datatrieve	9	130	2
16	diabetes	9	768	2
17	diggle-table-a2	9	310	9
18	ecoli	8	336	8
19	fertility	10	100	2
20	first-order-theorem-proving	52	6118	6
21	forest	28	523	4
22	glass	10	214	6
23	hayes-roth	5	160	3

Continua na próxima página

Tabela 10 – continuação da página anterior

Id	Nome	N. Atributos	N. Padrões	N. Classes
24	heart-h	14	294	5
25	heart-long-beach	14	200	5
26	heart-statlog	14	270	2
27	hill-valley	101	1212	2
28	ionosphere	34	351	2
29	iris	5	150	3
30	jEdit-4.0-4.2	9	274	2
31	jEdit-4.2-4.3	9	369	2
32	kc1-binary	87	145	2
33	kc3	40	458	2
34	LED-display-domain-7digit	8	500	10
35	mc2	40	161	2
36	mfeat-fourier	77	2000	10
37	mfeat-karhunen	65	2000	10
38	mfeat-morphological	7	2000	10
39	mfeat-zernike	48	2000	10
40	MiceProtein	78	1080	8
41	oil-spill	49	937	2
42	optdigits	63	5620	10
43	ozone-level-8hr	73	2534	2
44	page-blocks	11	5473	5
45	parkinsons	23	195	2
46	pc1	22	1109	2

Continua na próxima página

Tabela 10 – continuação da página anterior

Id	Nome	N. Atributos	N. Padrões	N. Classes
47	pc2	37	5589	2
48	pc3	38	1563	2
49	pc4	38	1458	2
50	phpcSeK3V	6	4839	2
51	planning-relax	13	182	2
52	prnn-synth	3	250	2
53	qsar-biodeg	42	1055	2
54	ringnorm	21	7400	2
55	rmftsa-sleepdata	3	1024	4
56	robot-failures-lp4	91	117	3
57	robot-failures-lp5	91	164	5
58	satellite-image	37	6435 6	
59	seeds	8	210	3
60	segment	19	2310	7
61	semeion	257	1593	10
62	sonar	61	208	2
63	spambase	58	4601	2
64	spectf	45	267	2
65	steel-plates-fault	34	1941	2
66	twonorm	21	7400	2
67	user-knowledge	6	403	5
68	vehicle	19	846	4
69	vertebra-column	7	310	3

Continua na próxima página

Tabela 10 – continuação da página anterior

Id	Nome	N. Atributos	N. Padrões	N. Classes
70	volcanoes-a1	4	3252	5
71	volcanoes-a2	4	1623	5
72	volcanoes-a3	4	1521	5
73	wall-robot-navigation	25	5456	4
74	wall-robot-navigation-5	5	5456	4
75	wdbc	31	569	2
76	wilt	6	4839	2
77	wine	14	178	3
78	yeast	9	1269	4

Fonte: Produzido pela autora (2022)