Universidade Federal de São Carlos – UFSCar
Centro de Ciências Exatas e de Tecnologia – CCET
Departamento de Computação – DC
Programa de Pós-Graduação em Ciência da Computação – PPGCC

Elaine Cecília Gatto

Além do Aprendizado Local e Global: Particionando o Espaço de Classes em Problemas de Classificação Multirrótulo

São Carlos
2023

Elaine Cecília Gatto

Além do Aprendizado Local e Global: Particionando o Espaço de Classes em Problemas de Classificação Multirrótulo

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências Exatas e de Tecnologia da Universidade Federal de São Carlos , como parte dos requisitos para a obtenção do título de Doutora em Ciência da Computação.

Área de concentração: Metodologias e Técnicas de Computação

Orientador: Ricardo Cerri
Coorientador: Mauri Ferrandin

São Carlos
2023

UNIVERSIDADE FEDERAL DE SÃO CARLOS

Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Folha de Aprovação

Defesa de Tese de Doutorado da candidata Elaine Cecília Gatto, realizada em 14/11/2023.

Comissão Julgadora:

Prof. Dr. Ricardo Cerri (UFSCar)

Prof. Dr. Diego Furtado Silva (UFSCar)

Prof. Dr. Alexandre Plastino de Carvalho (UFF)

Profa. Dra. Gisele Lobo Pappa (UFMG)

Prof. Dr. Luiz Henrique de Campos Merschmann (UFLA)

O Relatório de Defesa assinado pelos membros da Comissão Julgadora encontra-se arquivado junto ao Programa de Pós-Graduação em Ciência da Computação.

Dedico este trabalho a todas as pessoas que cruzaram meu caminho durante toda a minha vida. Não evoluímos sozinhos! Eu sou muito grata por estas pessoas terem me mostrado meus pontos positivos e negativos, me ajudando a crescer, amadurecer e evoluir.

Agradecimentos

Agradeço primeiramente a Deus que tem me permitido viver bem e realizar com alegria um trabalho tão desafiador quanto uma pesquisa de Doutorado. Agradeço também ao meu orientador, o Prof. Dr. Ricardo Cerri, por ter aceitado trabalhar comigo, me ajudando na construção e realização de um grande sonho. Agradeço ao Professores Dr. Mauri Ferrandin e Dr. Alan Valejo que tem me auxiliado com inúmeras dúvidas e acompanhado meu trabalho de perto. Graças a eles consegui o Best Paper Runner-Up Award no BRACIS 2023, o que foi uma enorme conquista para mim. Agradeço aos meus amigos do Laboratório de Pesquisa e também aos meus amigos do Grupo de Pesquisa BioMal, que me acolheram e tem me apoiado constantemente. Quero agradecer também a todos os Professores e Técnicos Administrativos do Departamento de Computação que de alguma forma me encorajam em direção ao sucesso. Não posso deixar de mencionar meus amigos da Seicho-No-Ie e também tantos outros amigos espalhados pelo mundo que tem me apoiado. Agradeço a meus pais, meu irmão e irmã, minhas sobrinhas, meu afilhado, meu cunhado e minha cunhada, por entenderem a minha ausência nos encontros familiares devido a minha dedicação a este projeto de Vida. Sem a minha família me apoiando e torcendo pelo meu sucesso no dia a dia, com certeza nada disso valeria a pena. Muitos foram os desafios que enfrentei durante esses quase 5 anos: meu irmão descobriu, tratou e curou-se de um câncer no cérebro logo no meu primeiro ano de doutorado e, ao mesmo tempo, minha irmã ficou grávida e deu a luz uma linda menina. Quebrei o nariz, tive covid o que agravou minha condição de hipoglicêmica, desenvolvi tenossonovite de quiervan nos dois pulsos durante o doutorado e descobri que terei de fazer tratamento para o resto da vida. Além disso, ano passado torci o joelho esquerdo jogando vôlei, um esporte que amo, mas consegui me recuperar em seis meses. Perdi amigos, conhecidos e familiares, os quais não pude me despedir por conta do covid. Também não posso deixar de mencionar o quanto foi gratificante e enriquecedor ter atuado como representante dos discentes do PPG-CC durante dois anos e meio, justamente no período de pandemia e também tendo como coordenador do programa o meu orientador. Conheci muitos alunos

e pude ajudá-los a resolver os seus problemas, assim como também pude colaborar com o PPG-CC de forma geral. Durante o doutorado, o meu sonho do sanduíche quase escapou pelas minhas mãos. A princípio, devido a pandemia, não haviam mais chances, mas então em 2022 apareceram oportunidades. Houve inúmeras complicações e tive a certeza de que não daria certo, mas no fim, consegui uma bolsa e fui para a Bélgica. Agradeço imensamente a Profa. Celine Vens por ter me aceito em seu grupo e me dado a oportunidade de conhecer um novo mundo e pessoas tão diferentes. Apesar de todas as pedras no caminho, fui capaz de cumprir com todos os meus prazos sem prejuízos e entregar finalmente os resultados da minha pesquisa. Cresci, amadureci, conheci outras culturas e países e hoje sou uma pessoa muito melhor. Por isto, sou muito grata a todos que de alguma forma fizeram parte deste processo. Muito Obrigada, Muito Obrigada, Muito Obrigada, Muito Obrigada, infinitamente Muito Obrigada!

"Nada é real se não acreditarmos em nós mesmos" (Sylvester Stallone)

Resumo

Induzir um modelo capaz de predizer um conjunto de rótulos para uma instância é objetivo da classificação multirrótulo, uma tarefa preditiva supervisionada do aprendizado de máquina. Trabalhos na literatura têm mostrado que identificar, modelar e explorar as correlações entre rótulos, melhora o desempenho preditivo dos classificadores multirrótulo. No entanto, as abordagens tradicionais, chamadas aqui de global e local, usadas para solucionar problemas de classificação multirrótulo podem não estar tirando proveito dessas correlações, já que em ambas essas correlações não são totalmente consideradas. Na abordagem global, todos os rótulos são aprendidos de uma única vez e informações ou correlações mais específicas podem ser ignoradas, enquanto que na abordagem local os rótulos são aprendidos de forma individual, tornando o aprendizado de correlações impraticável. Também há na literatura trabalhos que mostram que os conjuntos de dados mutirrótulos disponíveis atualmente têm um nível de dependência de rótulos muito baixo, e por isso explorar as correlações é impraticável, enquanto outros afirmam que aprender os rótulos individualmente é a solução mais compatível, e ainda trabalhos que recomendam os métodos da abordagem global por gerarem um único modelo mais compacto. Neste trabalho é proposta uma abordagem híbrida, que explora as vantagens e tenta mitigar as desvantagens das tradicionais abordagens global e local, a qual é chamada aqui de Partições Híbridas para Classificação Multirrótulo - *Hybrid Partitions for Multi-Label Classification* (HPML). Essa abordagem tem como objetivo encontrar diversas partições de rótulos, que são compostas por grupos disjuntos de rótulos correlacionados, aqui chamadas de partições híbridas. Quatro experimentos foram conduzidos para testar e validar a hipótese com diferentes versões de partições híbridas, as quais foram comparadas com as partições geradas pela abordagem global, local e também diferentes versões aleatórias. De forma geral, os experimentos mostraram que é possível encontrar uma partição híbrida capaz de melhorar o desempenho preditivo dos classificadores em vários conjuntos de dados e que métodos tradicionais ainda falham em aprender os rótulos assim como também lidar corretamente com as correlações entre rótulos.

Palavras-chave: Classificação Multirrótulo. Correlações entre rótulos. Particionamento do Espaço de Rótulos. Partições Multirrótulo.

Abstract

Inducing a model capable of predicting a set of labels for an instance is the objective of multi-label classification, a supervised predictive machine learning task. Work in the literature has shown that identifying, modeling and exploring correlations between labels improves the predictive performance of multi-label classifiers. However, the traditional approaches, referred to here as global and local, used to solve multi-label classification problems may not be taking advantage of these correlations, as in both these correlations are not fully considered. In the global approach, all labels are learned at once and more specific information or correlations can be ignored, while in the local approach labels are learned individually, making correlation learning impractical. There are also works in the literature that show that the currently available multi-label datasets have a very low level of label dependence, and therefore exploring correlations is impractical, while others claim that learning the labels individually is the most compatible solution, and even works that recommend global approach methods as they generate a single, more compact model. In this work, a hybrid approach is proposed, which explores the advantages and tries to mitigate the disadvantages of traditional global and local approaches, which is called Hybrid Partitions for Multi-label Classification - HPML. This approach aims to find several label partitions, which are composed of disjoint groups of correlated labels, here called hybrid partitions. Four experiments were conducted to test and validate the hypothesis with different versions of hybrid partitions, which were compared with the partitions generated by the global, local approach and also different random versions. In general, the experiments showed that it is possible to find a hybrid partition capable of improving the predictive performance of classifiers on various data sets and that traditional methods still fail to learn the labels as well as correctly deal with the correlations between labels.

Keywords: MultiLabel Classification. Label Correlations. Label Space Partitioning. MultiLabel Partitions.

Lista de ilustrações

Lista de tabelas

Lista de siglas

A	Acurácia
AUC	*Area Under the Curve*
AM	*Aprendizado de Máquina*
ATC	*Anatomical Therapeutic Chemical*
BR	*Binary Relevance*
BP-MLL	*Back Propagation Multi-Label Learning*
C	Cobertura
CA	*Classification Accuracy*
CCA	*Canonical Correlation InformAtion*
CC	*Classifier Chains*
CLP	*Constant Label Problem*
CLR	*CaLibrated Ranking by Pairwise Comparison*
CVIR	coeficiente de variação para a taxa de desbalanceamento médio
DNN	*Deep neural networks*
Distinto	*número de conjuntos de rótulos distintos*
DTI-MLCD	*Multi-Label Learning with Community Detection Method for Identifying Drug-Target Interactions Prediction*
DTs	*Decision Trees* - Árvores de Decisão
EAGLET	*Evolutionary AlGorithm for Multi-Label Ensemble OpTimization*
ECC	*Ensemble of Classifier Chains*
EME	*Evolutionary Algorithm Multi-Label*
EMLCs	*Ensemble of Multi Label Classifiers*

EMR *Exact Match Ratio*

EPS *Ensemble of Pruned Sets*

F1Ma Macro-F1

F1Mi Micro-F1

GCC *Group Sensitive Classifier Chains*

GACC *Genetic Algorithm for ordering Classifier Chains*

GLOCAL *Global and local multi-label correlations*

HL *Hamming Loss*

HPML Partições Híbridas para Classificação Multirrótulo - *Hybrid Partitions for Multi-Label Classification*

IA Inteligencia Artificial

IE *Is Error*

IRLbl nível de desbalanceamento de um rótulo específico

***k*-NN** *k-Nearest Neighboors*

LP *Label Powerset*

LIFT *MultiLabel Learning with Label Specific Features*

LDG *Label Dependency Graph*

MLC *Multi-Label Classification*

MLC-LC *Multi-Label Classification Label Clusters*

ML-LOC *Multi-Label Learning by Exploiting Label Correlations Locally*

ML-C4.5 *Multi-Label C4.5*

ML-*k*NN *Multi-Label k-Nearest Neighboors*

MLoss *Margin Loss*

MLP *Missing Label Prediction*

MMAC *Multi-Class Multi-Label Associative Classification*

MMLG Modelo Misto Linear Generalizada

MMP *Multi-Class Multi-label Perceptron*

MuLAM *Multi-Label Ant-Miner*

MaxIR proporção do rótulo mais comum em relação ao mais raro

MeanIR razão de desbalanceamento médio de cada rótulo

MLCLE *Multi-Label Classification with Label-Feature Encoding*

NLSP *Network-based Label Space Partition*

OE	*One Error*
P	Precisão
PD	proporção de conjuntos de rótulos distintos
PCA	*Principal Component Analysis*
PCTs	*Predictive Clustering Trees*
PM	Precisão-Média
PMa	Precisão-Macro
PMi	Precisão-Micro
PMM	*Parametric Mixture Models*
PPT	*Pruned Problem Transformation*
PMin	porcentagem de instâncias rotuladas com um único rótulo
PMax	proporção de ocorrências do conjunto de rótulos com a frequência máxima
PUnic	proporção de combinações de rótulos que são únicas
R	Revocação
RA*k*EL	*Random k-labELsets*
RE	*Ranking Error*
RF-PCT	*Random Forest of Predictive Clustering Trees*
RL	*Ranking Loss*
RMa	Revocação-Macro
RMi	Revocação-Micro
ROC	*Receiver Operating Characteristic*
RPC	*Ranking by Pairwise Comparison*
RF	*Random Forests* - Florestas Aleatórias
SOM	*Self-Organizing Map*
SVMs	*Support Vector Machines*
SA	*Subset Accuracy*
STS-NLSP	*Network-Based Label Space Partition method for predicting the Specificity of Membrane Transporter Substrates*
TCS	*Theoretical Complexity Score*
ULD	*Unconditional Label Dependency*
WLP	*Wrong Label Prediction*

Sumário

Capítulo 1

Introdução

Este capítulo serve como introdução aos capítulos seguintes, onde o problema investigado e todas as motivações do trabalho desenvolvido são apresentadas.

1.1 Contextualização e Motivação

Aprendizado de Máquina (AM) é uma área da *Inteligencia Artificial (IA)* capaz de resolver problemas a partir de experiências passadas (FACELI et al., 2011; KIM, 2017). De acordo com Alpaydin (2014), um modelo descritivo emprega o aprendizado não supervisionado para explorar ou descrever um conjunto de dados, enquanto um modelo preditivo emprega aprendizado supervisionado para realizar previsões a respeito do conjunto de dados.

Os conjuntos de dados podem pertencer a diferentes domínios como documentos (prontuários médicos, artigos, etc.), áudio (sons captados da natureza, por exemplo), vídeos (filmes, documentários, videoaulas), imagens (fotos publicadas nas redes sociais e tomografias, por exemplo), transações financeiras, dados de sensores, podendo ser obtidos a partir de várias fontes (ALLAM; DHUNNY, 2019; ZHENG et al., 2020; Mahmud et al., 2020).

Um conjunto de dados representa instâncias do problema a ser resolvido (FACELI et al., 2011). Estas instâncias podem ou não ser rotuladas, sendo espaço de rótulos (ou espaço de saída) o nome dado ao conjunto de rótulos do conjunto de dados em questão. Por exemplo, uma imagem pode ser descrita pelo seu título, resolução, cores, composição, origem, localização, e o que está presente nela: uma casa, um céu azul ou nublado, um pássaro branco voando, floresta, etc. Imagens podem então ser categorizadas por meio dessas palavras, por exemplo, uma imagem pode ser categorizada como “meio-ambiente”,

"por-do-sol" e "pássaro". Essas palavras, portanto, rotulam a imagem e podem ser usadas como atributo alvo (categoria ou rótulo). Uma instância que descreve uma imagem, portanto, é composta pelos atributos de entrada (características), e seus atributos de saída (rótulos) (MITCHELL, 1997; FACELI et al., 2011; BASTANLAR; OZUYSAL, 2014)

No aprendizado não supervisionado, os rótulos dos dados não são conhecidos, isto é, os dados não são rotulados (HERRERA et al., 2016). Como exemplo, é possível aplicar um algoritmo de agrupamento em uma coleção de imagens não categorizadas para agrupar imagens semelhantes, e de acordo com as suas características. Exemplos de tarefas realizadas por este tipo de aprendizado são o agrupamento, associação e sumarização (FACELI et al., 2011).

Já no aprendizado supervisionado, os rótulos dos dados são conhecidos - os dados são rotulados - e assim é possível avaliar a predição feita pelo modelo (ALPAYDIN, 2014). No caso de um conjunto de dados de imagens rotuladas, sabe-se de antemão a quais rótulos cada imagem pertence. Esse conjunto pode ser usado como entrada em um algoritmo de classificação que aprende os rótulos a partir desse conjunto. Um modelo preditivo é gerado (treinado) e então usado para classificar uma nova imagem - que não faz parte do conjunto de dados usado para treinar o modelo - em um ou mais dos rótulos aprendidos. Tarefas como classificação e regressão são realizadas por este tipo de aprendizado (FACELI et al., 2011).

Enquanto na classificação monorrótulo uma instância do conjunto de dados pertence a um único rótulo, na classificação multirrótulo uma instância pode pertencer a vários rótulos ao mesmo (HERRERA et al., 2016). O principal objetivo na classificação multirrótulo é construir um modelo que prediz um conjunto de rótulos para uma instância. Várias aplicações do mundo real podem ser modeladas como um problema multirrótulo (ZHANG; WU, 2014), como em Bioinformática (ZHOU et al., 2020), onde proteínas podem realizar muitas funções, categorização de texto (WANG et al., 2020), onde documentos pertencem a várias categorias ao mesmo tempo, e classificação musical (SANDEN; ZHANG, 2011), onde músicas pertencem a vários gêneros simultaneamente.

Dentre os desafios envolvidos na classificação multirrótulo, destacam-se a alta dimensionalidade do espaço de rótulos - quando existe um número muito grande de rótulos no espaço de rótulos - sendo algumas vezes superior ao número de atributos de entrada (TSOUMAKAS; KATAKIS; VLAHAVAS, 2008); o desbalanceamento (TAHIR et al., 2019); e a complexidade em identificar e explorar correlações entre rótulos (ZHU; KWOK; ZHOU, 2018).

Estudos têm mostrado que o desempenho preditivo de classificadores multirrótulo pode ser melhorado explorando correlações entre rótulos, e várias abordagens têm sido propostas para este fim (BAREZI; KWOK; RABIEE, 2017; ZHU; KWOK; ZHOU, 2018). A partir da modelagem das correlações, a predição de rótulos é facilitada, isto é, um

rótulo pode ser predito corretamente devido à sua correlação com outros rótulos, ou um rótulo que dificilmente é predito, pode finalmente ser predito. Para ilustrar como o aprendizado de correlações colabora para a melhoria das predições, considere a instância de teste do domínio de imagens apresentada na Figura 1. Considere também que durante o treinamento do modelo foi encontrada uma forte correlação entre os rótulos `montanha` e `praia`.

Observa-se que na Figura 2 existe uma praia entre as montanhas, no entanto, o rótulo `praia` é difícil de ser predito pois não é predominante na imagem. O rótulo `montanha`, no entanto, é mais facilmente predito, pois é predominante. Ao se considerar a correlação entre `montanha` e `praia`, aumenta-se a chance de `praia` ser predito quando `montanha` estiver presente na imagem. Portanto, ao se aprender as correlações, estas podem ser utilizadas para predizer rótulos que provavelmente não seriam preditos utilizando métodos que não consideram tais correlações.

Figura 1 – Exemplo ilustrativo de uma imagem com rótulos correlacionados. Fonte: Pixabay free image bank.

Além das correlações, alguns trabalhos têm proposto explorar o espaço de rótulos para resolver questões de escalabilidade (TSOUMAKAS; KATAKIS; VLAHAVAS, 2008), gerar hierarquias de rótulos (NIKOLOSKI; KOCEV; DžEROSKI, 2018; FERRANDIN; CERRI, 2023) ou agrupar rótulos (ABEYRATHNA, 2018) para melhorar o desempenho preditivo. Portanto, diferentes métodos e técnicas podem ser aplicados para tratar destas questões em problemas multirrótulo.

Tradicionalmente, os métodos de classificação multirrótulo podem ser divididos em duas categorias principais: adaptação de algoritmo e transformação de problema. Na abordagem de adaptação de algoritmo, novos algoritmos são desenvolvidos, ou algoritmos existentes são adaptados, para resolver o problema multirrótulo original. Esses algoritmos tratam todos os rótulos do problema ao mesmo tempo e treinam apenas um único classificador multirrótulo. No entanto, informações locais (individuais de cada rótulo) que podem ser úteis para explorar diferentes padrões nos dados podem ser ignoradas nesta abordagem (SILLA; FREITAS, 2011). Por exemplo, se há nos dados uma correlação entre um rótulo A e um rótulo B, e o rótulo B é predito pelo modelo tradicional mas

o A não, então isso é um indicativo de que o modelo poderia predizer A se aprendesse a correlação entre o rótulo B e A. Dessa forma, quando B for predito pelo modelo, A poderá ser predito. Árvores de decisão, algoritmos evolutivos, métodos probabilísticos, redes neurais artificiais e outros tipos de algoritmos podem ser adaptados para resolver o problema multirrótulo nesta abordagem.

Na abordagem de transformação de problema, os métodos transformam o problema multirrótulo em um conjunto de subproblemas monorrótulo, onde qualquer algoritmo de classificação convencional pode ser usado. Neste caso é necessário treinar um classificador binário para cada um dos rótulos individualmente ou um classificador multi-classe para cada subproblema multi-classe (CARVALHO; FREITAS, 2009). Apesar da flexibilidade destes métodos pode haver perda de informações e a não exploração das dependências entre rótulos durante o processo de treinamento.

Do ponto de vista do espaço de rótulos, pode-se dizer que a abordagem de adaptação de algoritmos é uma abordagem global, pois todos os rótulos são considerados ao mesmo tempo, enquanto a abordagem de transformação de problemas é uma abordagem local, pois o espaço de rótulos pode ser particionado separando os rótulos e tratando-os individualmente em pares ou em grupos. Como consequência, a abordagem global gera partições de dados globais e a abordagem local, partições locais. Neste trabalho, serão considerados como métodos que geram partições locais apenas aqueles que tratam os rótulos individualmente. Diante disto, introduz-se aqui o conceito de partições que podem ser geradas ao se realizar o particionamento do espaço de rótulos.

A Figura 2 apresenta as partições aqui introduzidas, onde o quadrado representa a partição em si, o círculo representa um grupo disjunto de rótulos e o losango representa o rótulo propriamente dito. Considere $L1$, $L2$, $L3$, $L4$, $L5$, $L6$, $L7$, $L8$ rótulos que compõem o espaço de rótulos de um conjunto de dados. Na partição global (Figura 2a) todos os rótulos estão juntos em um único círculo, isto é, um único grupo e portanto um único classificador multirrótulo é treinado. Já na partição local (Figura 2b) cada rótulo está em um círculo diferente, portanto, cada rótulo é um grupo e, neste exemplo, oito classificadores binários são treinados.

Por fim, a Figura 2c ilustra uma partição diferente chamada aqui de híbrida. Trata-se de uma partição que está entre as partições global e local. Diferentemente das partições local e global, que não procuram explorar a correlação entre sub-conjuntos de rótulos, neste trabalho as partições híbridas são obtidas ao se realizar o particionamento do espaço de rótulos explorando as correlações entre os rótulos. Cada partição híbrida gerada é composta por grupos disjuntos de rótulos correlacionados. Portanto, diferentes grupos de rótulos, compostos de diferentes rótulos correlacionados podem ser obtidos e, consequentemente diferentes partições híbridas.

Para cada círculo dentro da partição híbrida exemplificada na Figura 2c, um classificador é treinado. Se um grupo de uma partição híbrida encontrada é composto por um

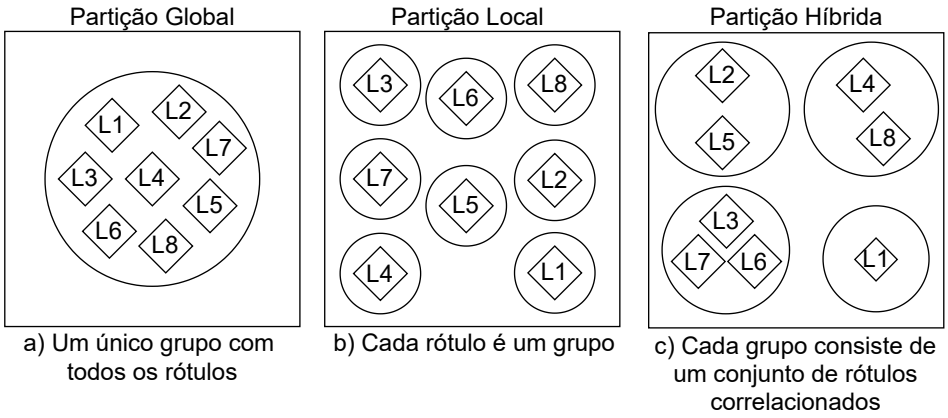


Figura 2 – Partições Global, Local e Híbrida.

único rótulo, então um classificador binário é treinado para aquele grupo, e se um grupo é composto por vários rótulos, então um classificador multirrótulo é treinado para aquele grupo. No exemplo da Figura 2c a partição híbrida é composta por quatro grupos de rótulos: $G1 = \{L2, L5\}$, $G2 = \{L4, L8\}$, $G3 = \{L3, L6, L7\}$ e $G4 = \{L1\}$. Neste caso será necessário treinar três classificadores multirrótulo ($G1$, $G2$ e $G3$) e um classificador binário ($G4$).

Uma das motivações para se propor o uso de partições híbridas é justamente estudar como particionamentos de dados considerando diretamente os rótulos correlacionados podem impactar no resultado preditivo de um classificador. Como mencionado por Read (2010), as correlações entre rótulos são naturais em dados multirrótulo, e como já mencionado, elas podem ajudar a melhorar o poder de predição dos classificadores. Portanto este trabalho propõe uma nova forma de explorar essas correlações em benefício do classificador multirrótulo tradicional.

Quando se particiona os dados multirrótulo considerando as instâncias (características + rótulos), o que se obtém como resultado é um agrupamento de instâncias similares, o que não necessariamente indica um agrupamento de rótulos similares ou reflete as correlações entre eles. A ideia aqui é aprender as correlações diretamente entre os rótulos e não as correlações entre rótulos a partir de instâncias similares. A Figura 3 ilustra um particionamento de instâncias, onde é necessário identificar todos os rótulos presentes em cada grupo (triângulos são instâncias).

É possível, ainda assim, obter partições de rótulos disjuntos a partir desse agrupamento de instâncias. Nesta situação, primeiro é necessário identificar os rótulos que estão presentes em cada grupo, depois é preciso aplicar algum critério para decidir em qual grupo cada rótulo será alocado. Com isto, pode-se obter uma partição diferente da original e, se existiam correlações entre os rótulos presentes no grupo, elas podem vir a ser desconsideradas no momento da distribuição dos rótulos nos grupos disjuntos.

Já a Figura 2c mostra o particionamento de rótulos, onde de fato pode ser identificada

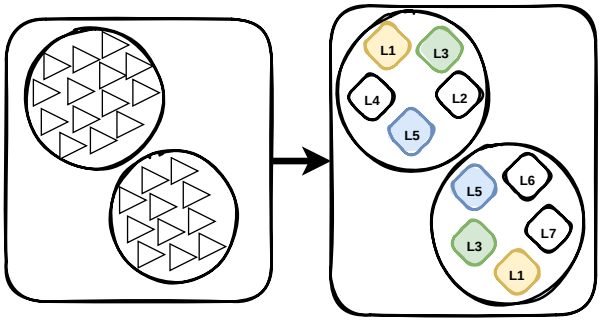


Figura 3 – Ilustração de um particionamento de instâncias e seus respectivos rótulos.

a existência de uma correlação entre rótulos diretamente. Uma correlação direta entre um rótulo l_b e um rótulo l_c pode ser medida usando alguma estratégia aplicada diretamente ao espaço de rótulos e, a partir daí, pode-se agrupar os rótulos de acordo com essa relação.

Além disso, ao se considerar a repetição dos rótulos nos grupos como mostrado na Figura 3 , um complicador é gerado no momento da predição, isto é, um classificador prediz que uma instância pertence à classe l_a, enquanto outro classificador prediz que essa mesma instância não pertence à classe l_a. Ao permitir a intersecção dos rótulos, os classificadores se tornam contraditórios e é necessário um trabalho extra de combinação dos resultados ao final, o que não é uma tarefa trivial. Esse processo de seleção transforma-se em uma desvantagem e pode acabar não explorando as correlações. Usar rótulos exclusivos facilita o trabalho do classificador, permite que as correlações entre os rótulos sejam modeladas diretamente e mantidas durante todo processo.

Considerando que nos grupos disjuntos haverá grupos com apenas um rótulo e também grupos com mais de um rótulo, é necessário pensar no uso de um classificador que consiga lidar com este contexto. Dessa forma, o uso de um classificador com versões local e global é necessário para a abordagem aqui proposta, e também considerando que uma comparação das melhorias entre partições híbridas com partições locais, globais e até aleatórias é feita. Portanto, o mesmo classificador deve ser usado em todos esses tipos de partições. Para ilustrar esta questão, considere as Figuras 4, 5 e 6, onde cada partição da Figura 2 está na forma de conjuntos de dados que serão usados para treinar e testar classificadores. O número total de atributos de entrada é dado por r e m é o número total de instâncias, $\hat{L}$ indica rótulo predito.

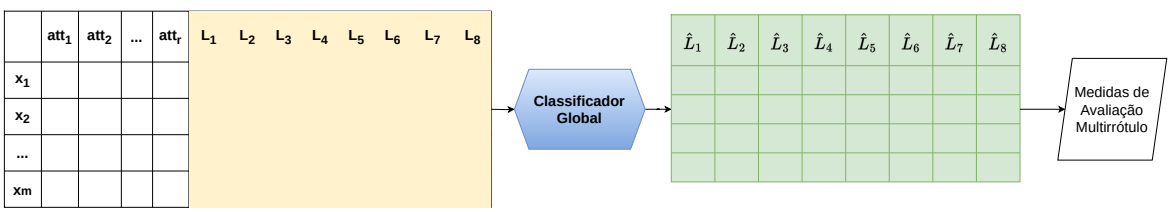


Figura 4 – Ilustração de uma partição global como um conjunto de dados estruturado.

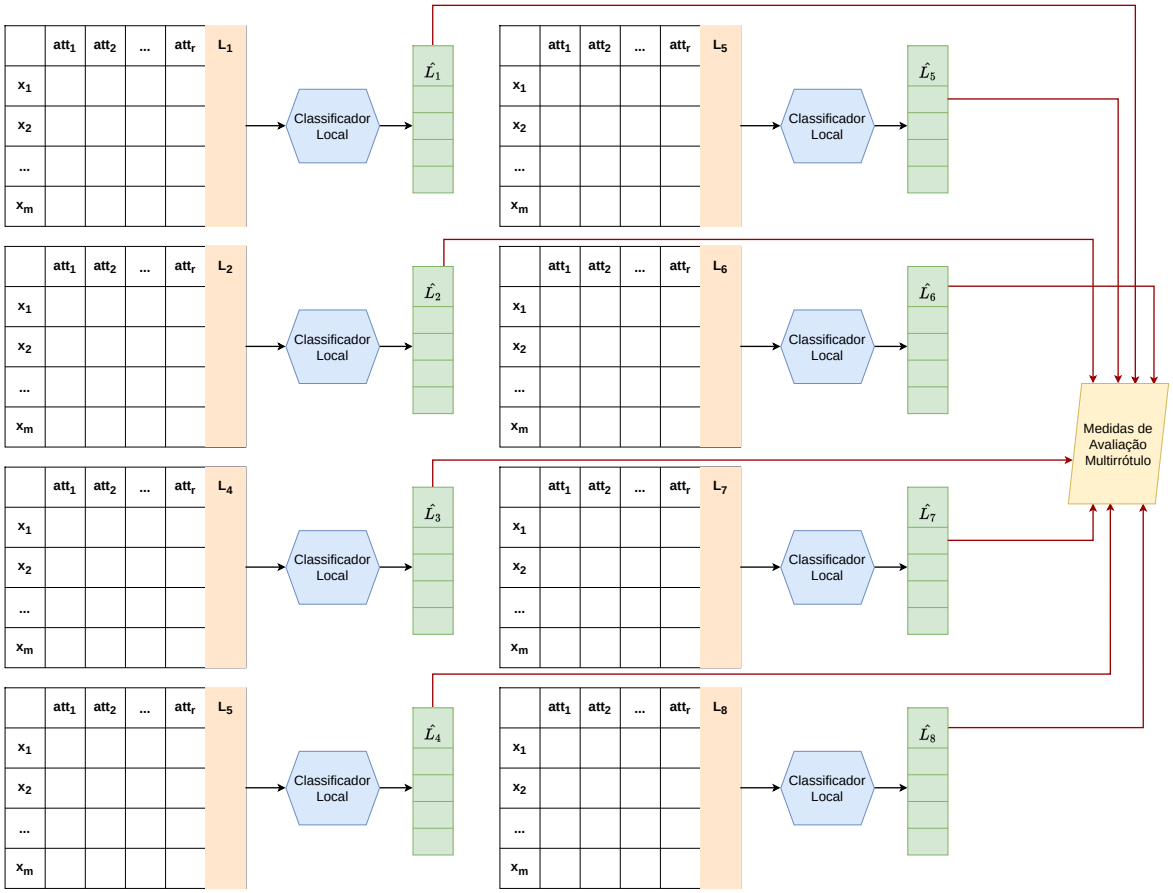


Figura 5 – Ilustração de uma partição local como um conjunto de dados estruturado.

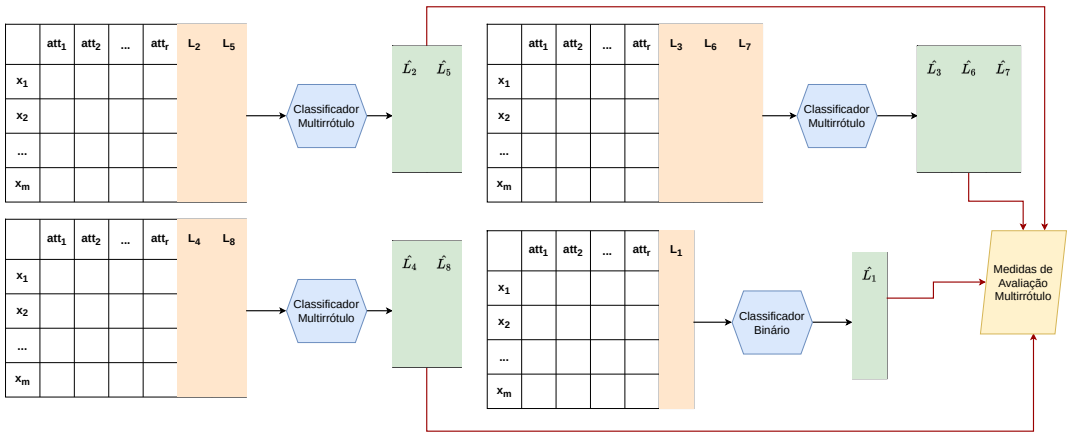


Figura 6 – Ilustração de uma partição híbrida como um conjunto de dados estruturado.

Reforçando, um classificador global para partições globais é induzido com o conjunto de treinamento e testado com o conjunto de teste, que aprende e produz simultaneamente as previsões para todos os rótulos (Figura 4). Um classificador local por grupo é induzido em partições locais, também com conjuntos de treinamento (Figura 5), e então essas previsões são reunidas para fazer a avaliação final para o conjunto de dados.

Nas partições híbridas um classificador global é induzido para cada um dos grupos que são compostos por dois ou mais rótulos, isto é, um modelo global por grupo. Em contrapartida, um classificador local (Figura 6) é induzido para cada um dos grupos da partição compostos por um único rótulo, isto é, um modelo local por grupo. As predições dos grupos com um rótulo e grupos com dois ou mais rótulos são então reunidas para fazer a avaliação final para todos os rótulos. Para obter resultados consistentes e coerentes, o mesmo classificador deve ser usado em todos os tipos de partições, - o mesmo classificador deve ser capaz de lidar com grupos compostos por um rótulo ou por vários rótulos - caso contrário, não será observado o quanto o classificador aprendeu.

Por exemplo, o algoritmo *Multi-Label C4.5* (ML-C4.5) (CLARE; KING, 2001) não pode lidar com grupos contendo um único rótulo e por isto ele não se encaixa na proposta. Também não faz sentido executar o Binary Relevance em todo o conjunto ou grupos com mais de um rótulo, pois o problema seria redividido e aprenderia rótulos separados quando é essencial aprender os rótulos agrupados. Portanto, o classificador utilizado no processo deve ser capaz de lidar com todos os tipos de partições aqui presentes.

Como um dos objetivos neste trabalho é analisar a melhoria das predições usando as partições híbridas com relação às abordagens tradicionais usando o mesmo bias, então é necessário usar exatamente o mesmo classificador em todos os tipos de particionamentos. Não faz sentido usar, em uma mesma partição, um SVM para grupos com um rótulo e uma árvore de decisão para grupos com muitos rótulos, pois eles funcionam de formas diferentes. Além disso, as predições são combinadas para fazer a avaliação final da partição, e combinar resultados de diferentes modelos não ajudaria a observar com clareza as melhorias.

Por esta razão, neste trabalho, o Framework CLUS (VENS et al., 2008) é usado, o qual induz árvores de decisão binárias e multirrótulo baseadas em árvores de agrupamento preditivas *Predictive Clustering Trees* (PCTs). Uma PCT binária é treinada para cada rótulo e suas saídas são combinadas para formar a predição final de vários rótulos para a partição local. Para a partição global, apenas uma PCT multirrótulo é necessária. Para validar as partições híbridas, um conjunto de PCTs multirrótulo, ou uma combinação de PCTs binárias e multirrótulo, pode ser aplicada, dependendo de quantos rótulos estão nos grupos da partição que está sendo validada. As saídas individuais são então combinadas para formar a predição final de vários rótulos. As Florestas Aleatórias Multirrótulo que são baseadas em árvores de decisão multirrótulo também são utilizadas pois também fornecem a mesma característica do CLUS.

Importante ressaltar que a Figura 2c ilustra apenas um exemplo de partição híbrida. No entanto, em problemas com muitos rótulos, existe um grande número de possíveis partições, portanto, um grande desafio é encontrar a partição híbrida mais adequada. Para ilustrar o desafio de se encontrar uma partição híbrida adequada entre todas as possíveis partições de rótulos, o conceito do número de Bell pode ser utilizado. O número de Bell (B_n) pode ser definido como o número total de partições de um conjunto com n elementos (onde $n \geq 0$).

Também pode ser definido como o número de partições possíveis de um conjunto com n elementos consistindo de k conjuntos separados e não vazios (conforme Equação 1), ou ainda como uma contagem das diferentes formas de se particionar um conjunto de dados. Além disso, uma partição P de um conjunto A é definida como um conjunto de subconjuntos não-vazios, disjuntos aos pares de A cuja união é A (COMTET, 1974; SPIVEY, 2008; MEZO, 2011).

$$B_n = \sum_{k=0}^{n} \binom{n}{k} \tag{1}$$

onde $\binom{n}{k}$ é o número de maneiras de se particionar um conjunto em k subconjuntos não vazios. Exemplos: se $n = 2$ então $B_2 = 2$, isto é, duas partições são geradas com apenas dois elementos ($\{1, 2\}$); e se $n = 3$, então $B_3 = 5$, isto é, cinco partições são geradas com três elementos ($\{1, 2, 3\}$). Todas as possíveis partições geradas para B_2 e B_3 são ilustradas na Tabela 1.

Tabela 1 – Exemplos Número de Bell para $n = 2$ e $n = 3$.

B_2	B_3
{1, 2}	{ { 1, 2, 3 } }
{ { 1 }, { 2 } }	{ { 1 }, { 2 }, { 3 } }
	{ { 1, 2 } , { 3 } }
	{ { 1, 3 }, { 2 } }
	{ { 2, 3 }, { 1 } }

O número total de rótulos l do espaço de rótulos de um conjunto de dados multirrótulo pode então ser considerado como o parâmetro n do número de Bell. Neste caso, tomando como exemplo o conjunto apresentado na Figura 2, o número de possíveis partições do espaço de rótulos é igual a 4140, pois $n = l = 8$ e $B_8 = 4140$. Assim, para conjuntos de dados com espaços de rótulos de alta dimensão, torna-se muito mais desafiador encontrar uma partição híbrida adequada.

1.2 Hipótese

Diante do contexto e motivação apresentados para a realização deste trabalho, a seguinte hipótese foi formulada:

> *É possível encontrar uma partição composta por grupos disjuntos de rótulos correlacionados que melhore o desempenho preditivo do classificador em relação às tradicionais abordagens global e local.*

1.3 Objetivos

O objetivo geral desta tese é:

> *Desenvolver, implementar e avaliar uma estratégia capaz de particionar o espaço de rótulos, explorando as correlações entre rótulos, de forma a gerar várias partições híbridas as quais devem ser compostas por grupos disjuntos de rótulos correlacionados e que sejam capazes de otimizar o desempenho dos classificadores. Rótulos pertencentes a um determinado grupo não podem pertencer a outros grupos. O número de partições a serem geradas, assim como o número de grupos dentro de cada partição, deve ser definido de forma automática pelo método de particionamento.*

Os objetivos específicos são:

- ❏ Estudar conceitos relacionados à classificação multirrótulo e métodos atualmente utilizados para resolver problemas de classificação multirrótulo;
- ❏ Estudar conceitos relacionados à modelagem das correlações entre rótulos;
- ❏ Estudar conceitos relacionados ao particionamento do espaço de rótulos;
- ❏ Comparar as partições híbridas com as partições global, local e também aleatórias;
- ❏ Analisar a contribuição das partições híbridas no desempenho preditivo geral;
- ❏ Analisar a influência das características dos dados multirrótulo no desempenho preditivo;
- ❏ Analisar o desempenho preditivo da estratégia proposta em diferentes medidas de avaliação;
- ❏ Analisar as partições híbridas.

O método foi desenvolvido e avaliado, e a hipótese pôde ser testada a partir da condução de diferentes experimentos. Os detalhes são relatados nos Capítulos 4, 5 e 6.

1.4 Contribuições

As principais contribuições desta tese estão listadas a seguir:

- ❏ Apresentação de uma abordagem capaz de gerar partições híbridas denominada HPML;
- ❏ Três instanciações principais da abordagem principal foram propostas e testadas;
- ❏ Quatro variações da abordagem principal baseadas no conceito de cadeias de classificadores também foram propostas e testadas;
- ❏ Nove diferentes abordagens para gerar partições aleatórias foram propostas, sete baseados no algoritmo de agrupamento hierárquico aglomerativo e duas baseadas em métodos de detecção de comunidades;
- ❏ É apresentada uma abordagem para gerar todas as partições possíveis para um conjunto de dados multirrótulo. Essas partições foram testadas e verificadas qual dentre elas tem o melhor desempenho preditivo e é aqui denominada oráculo;
- ❏ Uma abordagem para gerar todas as partições possíveis, validá-las, escolher e testar uma entre elas é apresentada e aqui denominada de exaustiva;
- ❏ Uma comparação preditiva e estatística entre as diferentes abordagens é reportada;
- ❏ Uma análise de quais tipos de partições são melhores em quais tipos de cenários é realizada.

1.5 Organização do Documento

O restante deste documento está organizado conforme a seguir. A fundamentação teórica é apresentada no Capítulo 2, o qual começa explicando e formalizando a classificação multirrótulo. Em seguida, aspectos importantes da classificação multirrótulo são discutidos: abordagens para tratar problemas multirrótulo, combinação de classificadores multirrótulo, modelagem das correlações entre rótulos, medidas de avaliação de desempenho multirrótulo, desbalanceamento, escalabilidade, dimensionalidade e características dos dados multirrótulo. No Capítulo 3 são apresentados os trabalhos mais correlacionados com a tese aqui apresentada, enquanto que no Capítulo 4 é apresentado o método desenvolvido assim como suas variações. O Capítulo 5 apresenta as configurações e ferramentas utilizadas nos experimentos e, por fim, o Capítulo 6 apresenta e discute os resultados obtidos e trabalhos futuros.

Capítulo 2

Classificação Multirrótulo

Um classificador tem como objetivo atribuir uma instância, ainda não classificada, a um (ou mais) rótulos disponíveis conhecidos previamente (HAN; KAMBER; PEI, 2011). Quando uma instância é atribuída a um único rótulo, a classificação é denominada Monorrótulo. Como exemplo, pode-se determinar se uma figura (instância) pertence a uma de duas categorias (rótulos) mas nunca a ambas as categorias ao mesmo tempo. Quando uma instância pode ser atribuída a vários rótulos ao mesmo tempo, então a classificação é denominada Multirrótulo (*Multi-Label Classification* (MLC))(FACELI et al., 2011).

Considere o conjunto de dados ilustrado na Figura 7a onde cada pequeno círculo é uma instância. A Figura 7b ilustra a classificação monorrótulo, onde é posssível ver nitidamente a separação entre as instâncias que pertencem e não pertencem à classe do conjunto. A Figura 7c ilustra a classificação multirrótulo, onde as cores azul, verde e vermelha indicam uma classe diferente, enquanto que a cor cinza indica que aquela instância não foi classificada em uma das classes. Por exemplo, um pequeno círculo com as cores azul, vermelha e cinza indica que a instância foi classificada nas classes azul e vermelha mas não na classe verde. Portanto, cada instância pode pertencer a nenhuma, uma ou mais classes ao mesmo tempo.

A classificação monorrótulo e multirrótulo diferem também no retorno do resultado (HERRERA et al., 2016). Um classificador monorrótulo retorna apenas um rótulo ou um valor binário: [1] pertence ao rótulo em questão e [0] não pertence. Um classificador multirrótulo pode retornar um ou mais rótulos, ou um vetor binário de valores de saída, em que cada posição corresponde a um rótulo: $[1, 0, 0, 1]$

Diversas aplicações do mundo real podem ser modeladas como um problema multirrótulo (ZHANG; WU, 2014), como bioinformática, onde uma proteína pode desempenhar várias funções (ZHOU et al., 2020); categorização de textos, onde um documento pode

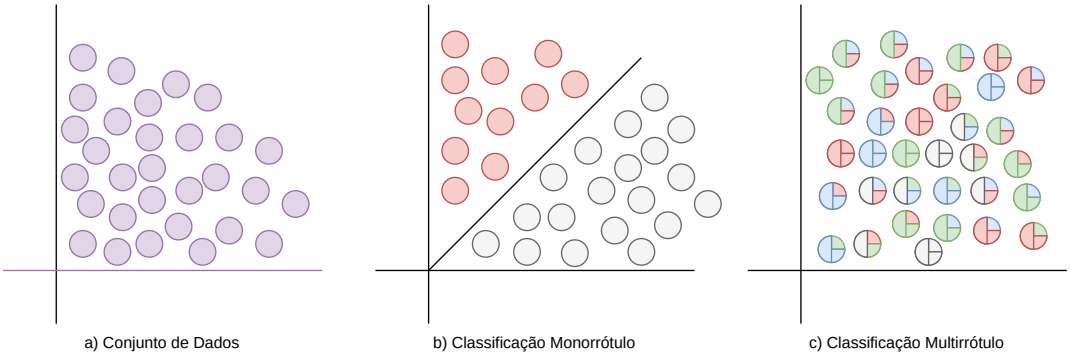


Figura 7 – Classificação Monorrótulo e Multirrótulo. Fonte: Elaborado pela autora com base em Cerri (2014).

pertencer a várias categorias ao mesmo tempo (WANG et al., 2020); classificação de músicas, onde uma música pode pertencer a vários gêneros simultaneamente (SANDEN; ZHANG, 2011); fármacos, onde medicamentos podem ter duas ou mais ações biológicas diferentes (KAWAI; TAKAHASHI, 2009); diagnóstico médico, onde os sintomas podem estar associados a mais de uma doença (SHAO et al., 2013). Devido à ampla gama de aplicações, o interesse em explorar técnicas da classificação multirrótulo aumentou na comunidade científica e na indústria. Formalmente, um conjunto de dados multirrótulo $\mathcal{D}$ pode ser definido como a seguir (MADJAROV et al., 2012; READ, 2010; MOYANO et al., 2020).

- $\forall \mathbf{x}_i \in \mathcal{X}, \mathbf{x}_i = \{x_{i1}, x_{i2}, ..., x_{id}\}$: espaço de instâncias que consiste de valores de dados primitivos, onde d é o número de atributos da instância;
- $\mathcal{L} = \{L_1, L_2, ..., L_l\}$: espaço de rótulos composto por l variáveis discretas com valores 0 ou 1;
- $\mathcal{D} = \{(\mathbf{x}_i, Y_i) \mid 1 \leq i \leq m\}$: conjunto de dados multirrótulo onde:
  - m: é o número total de instâncias do conjunto de dados;
  - $\mathbf{x}_i \in \mathcal{X}$: é uma instância do espaço de instâncias composta por d atributos;
  - $Y_i \subseteq \mathcal{L}$: é um subconjunto do espaço de rótulos associado à instância $\mathbf{x}_i$;
  - $(\mathbf{x}_i, Y_i)$: é uma instância do conjunto de dados multirrótulo composto por uma instância $\mathbf{x}_i$ e um subconjunto de rótulos Y_i associado a ela.
  - $\mathbf{Y} = \{y_1, y_2, ..., y_l\} = \{0, 1\}^l$: representação na forma de um vetor binário de q dimensões. Cada elemento do vetor tem o valor 1 se o rótulo é relevante, ou 0 caso contrário.

Dado um conjunto de dados multirrótulo, um modelo preditivo deve ser induzido para a obtenção de um conjunto de rótulos para novas instâncias. Esse modelo pode fornecer

o conjunto de rótulos diretamente (bipartição), ou um ranqueamento (lista ordenada de rótulos relevantes) de todos os rótulos conhecidos. No caso do fornecimento direto do conjunto de rótulos, deve ser encontrado um modelo preditivo $h : \mathcal{X} \rightarrow 2^{\mathcal{L}}$ que forneça um conjunto de rótulos $\hat{Y} = h(\mathbf{x}_i)$ para uma instância de teste $\mathbf{x}_i$, onde:

- $2^{\mathcal{L}}$: é o conjunto de todos os subconjuntos de $\mathcal{L}$;
- $\hat{Y}_i = \{\hat{y}_1, \hat{y}_2, ..., \hat{y}_p\}$: é o conjunto de rótulos preditos para $\mathbf{x}_i$ onde p é o número total de rótulos preditos;
- $Y_i = \{y_1, y_2, ..., y_n\}$: é o conjunto de rótulos verdadeiros para $\mathbf{x}_i$ onde n é o número de rótulos verdadeiros de Y_i;

No caso do fornecimento de um ranqueamento de todos os rótulos conhecidos, deve ser encontrado um modelo $f : \mathcal{X} \times \mathcal{L} \rightarrow \mathcal{R}$. A função f induz um modelo que gera uma lista ordenada de todos os possíveis rótulos que expressa a relevância dos rótulos dada uma instância $\mathbf{x}_i$. Essa lista pode ser obtida usando métodos como votação ou ponderação. Por exemplo, se o problema multirrótulo é dividido em problemas binários, então a saída de cada classificador binário pode ser usada como um voto, os quais são contabilizados para cada rótulo e a lista ordenada é gerada de acordo com o total de votos para cada rótulo (GIBAJA; VENTURA, 2014). Outro método de votação consiste em cada classificador gerar como saída a probabilidade da relevância de cada rótulo. Se a probabilidade estiver abaixo de um limiar, então os outros classificadores são consultados para decidir a relevância do rótulo no ranking (MADJAROV; GJORGJEVIKJ; DŽEROSKI, 2011). Outras abordagens podem ser encontradas em um estudo detalhado feito por Vembu e Gärtner (2011).

A partir de um ranking é possível obter uma bipartição por meio da aplicação de um limiar[1], o que permite a utilização de ambos os modelos (de ranqueamento ou não) para a resolução de um problema multirrótulo. Várias abordagens para a utilização de limiares foram propostas na literatura, e uma revisão detalhada pode ser encontrada em (FAN; LIN, 2007; IOANNOU et al., 2010). Ao longo deste texto, serão também utilizadas a seguintes notações:

- $\mathcal{D}_{treino} = \{(\mathbf{x}_i, Y_i) \mid 1 \leq i \leq m_{tr}\}$: é o conjunto de dados de treinamento contido em $\mathcal{D}$, onde m_{tr} é o número de instâncias em $\mathcal{D}_{treino}$;
- $\mathcal{D}_{teste} = \{(\mathbf{x}_i, Y_i) \mid 1 \leq i \leq m_{ts}\}$: conjunto de teste contido em $\mathcal{D}$, onde m_{ts} é o número de instâncias em $\mathcal{D}_{teste}$;
- $\mathcal{D}_{val} = \{(\mathbf{x}_i, Y_i) \mid 1 \leq i \leq m_{val}\}$: é o conjunto de dados de validação contido em $\mathcal{D}$, onde m_{val} é o número de instâncias em $\mathcal{D}_{val}$;

[1] um limite de corte aplicado aos valores reais resultantes para definir se o rótulo predito tem valor 1 ou 0.

Problemas multirrótulo podem ser tratados de duas principais formas: Global (ou abordagem global ou ainda adaptação de algoritmo) e Local (ou abordagem local ou ainda transformação de problemas). Cada uma delas é detalhada na seção 2.1. Neste capítulo também serão discutidos os seguintes aspectos e desafios da classificação multirrótulo: combinação de classificadores multirrótulo (seção 2.2), correlações entre rótulos (seção 2.3), medidas de avaliação (seção 2.4), outros aspectos como desbalanceamento, dimensionalidade e escalabilidade (seção 2.5) e características dos conjuntos de dados multirrótulo (seção 2.6).

2.1 Abordagens para Problemas Multirrótulo

Nesta seção serão apresentados os principais métodos para resolver problemas de classificação multirrótulo. A subseção 2.1.1 apresenta métodos locais e a subseção 2.1.2 os métodos globais.

2.1.1 Abordagem Local

A abordagem local transforma o problema multirrótulo em um conjunto de subproblemas monorrótulo, onde qualquer algoritmo de classificação convencional pode ser usado. A Figura 8 apresenta os principais métodos de transformação de problema e os mesmos serão explicados ao longo desta seção. A transformação pode ocorrer com base nos rótulos ou nas instâncias. Nos métodos baseados em rótulos, N classificadores são treinados, cada qual para uma classe ou um subconjunto de classes do problema, sendo N igual ao número de rótulos (l) do conjunto. Nos métodos baseados em instâncias, o conjunto de rótulos associados a cada instância é redefinido para converter o problema multirrótulo original em um ou mais problemas monorrótulo do tipo binário ou multi-classe (ZHANG; WU, 2014). Em termos de partição, conforme definido nesta pesquisa, os métodos da abordagem local trabalham com partições locais.

Para fins ilustrativos, considere o conjunto de dados de exemplo ($\mathcal{D}_e$) apresentado na Tabela 2, o qual contém quatro atributos ($Atr1$, $Atr2$, $Atr3$ e $Atr4$), cinco rótulos ($L1$, $L2$, $L3$, $L4$ e $L5$) e cinco instâncias ($\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, $\mathbf{x}_4$ e $\mathbf{x}_5$). A Tabela 4 apresenta cada instância com seu respectivo conjunto de rótulos, isto é, os rótulos para os quais a instância é positiva. A frequência de cada rótulo é apresentada na Tabela 3. No caso do método baseado em rótulos, para $\mathcal{D}_e$ serão necessários cinco classificadores binários, um para cada rótulo do conjunto.

Os métodos baseados em instâncias podem ser divididos em três formas diferentes, conforme apresenta a Figura 8. A eliminação de instâncias consiste em remover do conjunto de dados as instâncias multirrótulo. Para $\mathcal{D}_e$ não há instâncias monorrótulo, apenas multirrótulo, portanto, esta técnica não é viável neste caso (CARVALHO; FREITAS, 2009).

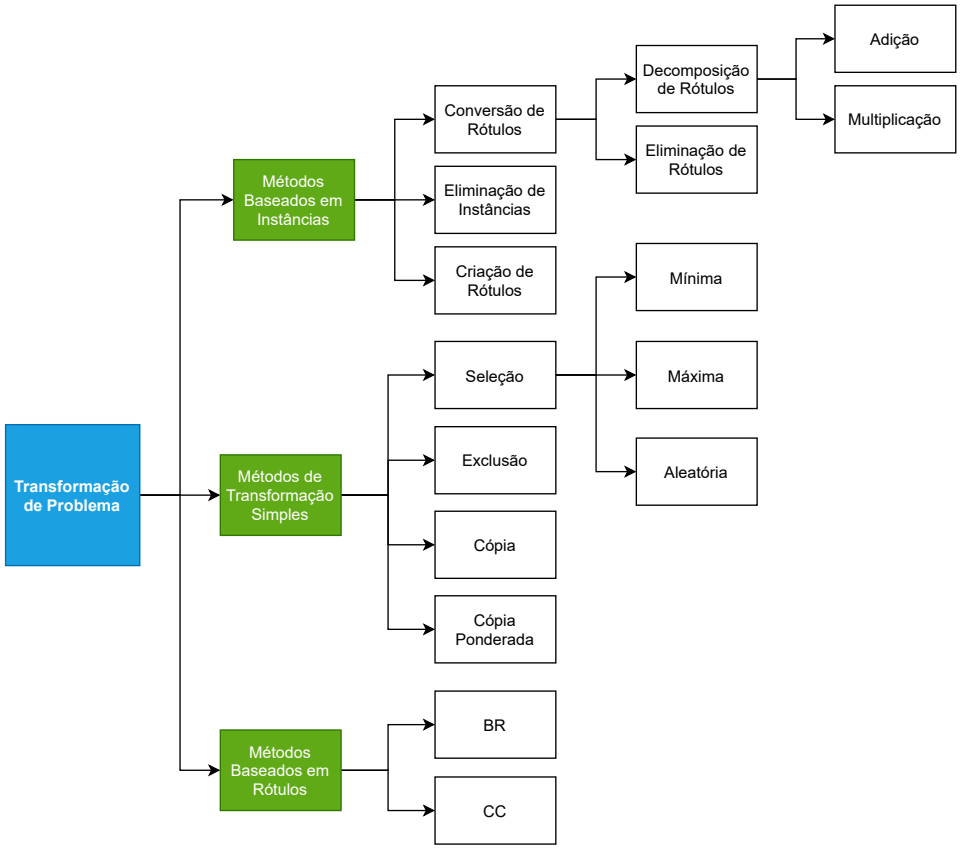


Figura 8 – Taxonomia de estratégias que utilizam abordagem local.Fonte: Elaborado pela autora com base em Carvalho e Freitas (2009) e Sorower (2010).

Tabela 2 – Conjunto de dados de exemplo $\mathcal{D}_e$.

Instância	**Atr1**	**Atr2**	**Atr3**	**Atr4**	**L1**	**L2**	**L3**	**L4**	**L5**
$\mathbf{x}_1$	25	58	24	57	1	0	1	1	0
$\mathbf{x}_2$	43	38	38	781	1	1	1	0	0
$\mathbf{x}_3$	8	73	24	70	0	1	0	1	0
$\mathbf{x}_4$	79	9	65	63	1	0	0	0	1
$\mathbf{x}_5$	100	61	5	48	1	0	1	1	0

Tabela 3 – Frequência dos Rótulos.

Rótulo	**Frequência**
$L1$	4
$L2$	2
$L3$	3
$L4$	3
$L5$	1

Tabela 4 – Conjuntos de Rótulos.

Instância	**Rótulo**
$\mathbf{x}_1$	$L1, L3, L4$
$\mathbf{x}_2$	$L1, L2, L3$
$\mathbf{x}_3$	$L2, L4$
$\mathbf{x}_4$	$L1, L5$
$\mathbf{x}_5$	$L1, L3, L4$

Os rótulos existentes no conjunto de dados podem ser eliminados ou decompostos com o método de conversão de rótulos. Uma instância que possui mais de um rótulo passa a pertencer a apenas um rótulo na eliminação de rótulos. O rótulo a que a instância passará a pertencer pode ser escolhido de maneira aleatória - um dos rótulos é selecionado, ou determinística - o rótulo que mais parece ser verdadeiro é selecionado. A Tabela 5 apresenta um possível resultado do método de conversão de rótulos para $\mathcal{D}_e$ (FACELI et al., 2011).

Tabela 5 – Método de Eliminação para $\mathcal{D}_e$.

Instância	**Rótulo**
$\mathbf{x}_1$	$L4$
$\mathbf{x}_2$	$L3$
$\mathbf{x}_3$	$L2$
$\mathbf{x}_4$	$L5$
$\mathbf{x}_5$	$L1$

O problema multirrótulo original com L rótulos e m instâncias é dividido em k conjuntos de problemas monorrótulo no método de decomposição de rótulos, o qual pode ser feito de maneira aditiva ou multiplicativa. Quando da aplicação do método aditivo, um número de classificadores igual ao número de rótulos que rotulam pelo menos uma instância é utilizado ($C = l$). No método multiplicativo, classificadores são treinados combinando todos os possíveis sub-problemas monorrótulo. Aplicando o método aditivo em $\mathcal{D}_e$, cinco classificadores serão necessários, um para cada rótulo. No caso de $\mathcal{D}_e$, ao aplicar o método multiplicativo, 25 combinações diferentes serão obtidas[2]. Como são necessários vários classificadores, o método de decomposição de rótulos pode ter problemas de escalabilidade[3] (CARVALHO; FREITAS, 2009).

A criação de rótulos consiste em combinar em um novo rótulo todos os rótulos atribuídos a uma instância. Essa combinação pode aumentar de maneira significativa o número de rótulos no conjunto de dados, o que pode gerar problemas de escalabilidade e desbalanceamento[4] (FACELI et al., 2011). No caso de $\mathcal{D}_e$ cada um dos conjuntos de rótulos identificados (Tabela 4) podem ser transformados nos seguintes rótulos: $L6$, $L7$, $L8$, $L9$ e $L10$. Apesar de simples, o Label Powerset pode produzir problemas de perda de informações em termos de rótulos ou correlações entre rótulos que não estão representadas no conjunto de treino.

Existem ainda outros métodos de transformação simples que podem ser utilizados para dividir o problema multirrótulo. O método cópia, simplesmente cria uma nova instância para cada conjunto de rótulos identificado, enquanto que o método cópia ponderada (Tabela 6) faz o mesmo mas utilizando um peso para cada instância. Ambos os métodos

[2] (5 rótulos × 5 rótulos = 25 classificadores monorrótulo).
[3] problemas de processamento devido ao alto número de rótulos.
[4] poucas instâncias positivas para cada rótulo.

aumentam significativamente o número de instâncias do conjunto de dados (TSOUMAKAS, 2010).

Tabela 6 – Métodos de Cópia e Cópia Ponderada para $\mathcal{D}_e$.

Instância	**Cópia**	**Ponderada**
$\mathbf{x}_{1a}$	$L1$	0.33
$\mathbf{x}_{1b}$	$L3$	0.33
$\mathbf{x}_{1c}$	$L4$	0.33
$\mathbf{x}_{2a}$	$L1$	0.33
$\mathbf{x}_{2b}$	$L2$	0.33
$\mathbf{x}_{2c}$	$L3$	0.33
$\mathbf{x}_{3a}$	$L2$	0.5
$\mathbf{x}_{3b}$	$L4$	0.5
$\mathbf{x}_{4a}$	$L1$	0.5
$\mathbf{x}_{4b}$	$L5$	0.5
$\mathbf{x}_{5a}$	$L1$	0.33
$\mathbf{x}_{5b}$	$L3$	0.33
$\mathbf{x}_{5c}$	$L4$	0.33

Outra maneira de dividir o problema multirrótulo consiste em selecionar (seleção) um dos rótulos do conjunto de rótulos da instância e pode ser feita com base na frequência dos rótulos da instância com relação ao conjunto de dados (Tabela 7), ou de maneira aleatória (seleção aleatória). As frequências máxima ou mínima podem ser usadas como critério de seleção (seleção máxima e seleção mínima). O método exclusão funciona da mesma forma que o método de eliminação de instâncias multirrótulo (SOROWER, 2010). Apesar da simplicidade na transformação, esses métodos levam à perda de informações, não modelam correlações entre os rótulos, entre outros problemas que podem ser resolvidos com métodos mais sofisticados (GIBAJA; VENTURA, 2014).

Tabela 7 – Métodos de Seleção para $\mathcal{D}_e$.

Instância	**Máxima**	**Mínima**	**Aleatória**
$\mathbf{x}_1$	$L1$	$L3$	$L4$
$\mathbf{x}_2$	$L1$	$L2$	$L3$
$\mathbf{x}_3$	$L4$	$L4$	$L2$
$\mathbf{x}_4$	$L1$	$L5$	$L5$
$\mathbf{x}_5$	$L1$	$L4$	$L1$

Dentro da taxonomia apresentada na Figura 8, diversos métodos podem ser encontrados na literatura. Nesta pesquisa, apenas os métodos clássicos mais populares são abordados. Esses métodos derivados podem ainda ser divididos em três principais categorias: baseados em Relevância Binária (*Binary Relevance* (BR)), baseados em *Label Powerset* (LP) e baseados em pares - os quais também foram estendidos por outros métodos.

2.1.1.1 Métodos baseados no BR

O Binary Relevance (BOUTELL et al., 2004) é um método baseado em rótulos que divide o problema original em l problemas binários, sendo necessários então l classificadores, um para cada rótulo do conjunto. Para cada classificador binário, as instâncias que não contêm o rótulo específico daquele classificador, são rotuladas como instâncias negativas. Isto pode levar a um desbalanceamento de rótulos, pois provavelmente o número de instâncias negativas para o rótulo específico será maior que o de positivas (READ, 2010; ZHANG et al., 2018).

Como todos os rótulos são tratados individualmente pelo BR, não é possível modelar as correlações entre os rótulos. A simplicidade e escalabilidade são as grandes vantagens deste método (GIBAJA, 2015). A Tabela 8 apresenta o resultado da transformação de $\mathcal{D}_e$ usando BR. Instâncias negativas são representadas pelo símbolo $-$ e as positivas por $+$ antes de cada rótulo. Cada um dos conjuntos transformados correspondem a um subconjunto de rótulos de uma partição local do conjunto original, conforme mostra a Figura 9[5].

Tabela 8 – Conjunto de dados BR para $\mathcal{D}_e$.

$\mathbf{x}_i$	Y	$\mathbf{x}_i$	Y	$\mathbf{x}_i$	Y	$\mathbf{x}_i$	Y	$\mathbf{x}_i$	Y
$\mathbf{x}_1$	$+L1$	$\mathbf{x}_1$	$-L2$	$\mathbf{x}_1$	$+L3$	$\mathbf{x}_1$	$+L4$	$\mathbf{x}_1$	$-L5$
$\mathbf{x}_2$	$+L1$	$\mathbf{x}_2$	$+L2$	$\mathbf{x}_2$	$+L3$	$\mathbf{x}_2$	$-L4$	$\mathbf{x}_2$	$-L5$
$\mathbf{x}_3$	$-L1$	$\mathbf{x}_3$	$+L2$	$\mathbf{x}_3$	$-L3$	$\mathbf{x}_3$	$+L4$	$\mathbf{x}_3$	$-L5$
$\mathbf{x}_4$	$+L1$	$\mathbf{x}_4$	$-L2$	$\mathbf{x}_4$	$-L3$	$\mathbf{x}_4$	$-L4$	$\mathbf{x}_4$	$+L5$
$\mathbf{x}_5$	$+L1$	$\mathbf{x}_5$	$-L2$	$\mathbf{x}_5$	$+L3$	$\mathbf{x}_5$	$+L4$	$\mathbf{x}_5$	$-L5$

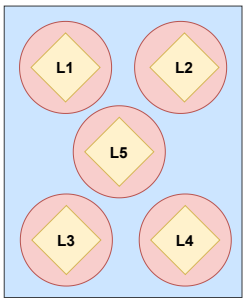


Figura 9 – Partição Local Binary Relevance para $\mathcal{D}_e$.

Um método de transformação de problema baseado no Binary Relevance é o *Classifier Chains* (CC)) (READ et al., 2009), que constrói uma cadeia de classificadores individuais baseada nas predições dos classificadores binários anteriores na cadeia. Classifier Chains resolve o problema de modelagem das correlações do Binary Relevance, pois as predições de rótulos individuais se tornam entrada para os outros classificadores na cadeia. Ainda

[5] quadrado ou retângulo (azul) = partição; círculo (vermelho) = subconjunto de rótulos; losango (amarelo) = rótulo

que Classifier Chains possa ser paralelizado, pode haver problemas de escalabilidade (GIBAJA; VENTURA, 2014).

2.1.1.2 Métodos baseados em LP

O Label Powerset (BOUTELL et al., 2004) é um método baseado na criação de rótulos onde cada combinação de rótulos no conjunto de dados é considerado como um novo e único rótulo no conjunto de dados transformado, permitindo assim explorar as correlações entre os rótulos (Tabela 4). Apesar disso, alguns conjuntos de rótulos presentes no conjunto de dados podem não ser identificados no processo de treinamento. Label Powerset também pode ter problemas de desbalanceamento - uma vez que o conjunto de dados pode ter várias combinações de rótulos - e também escalabilidade - muitos rótulos novos podem ser criados exigindo mais classificadores (TSOUMAKAS, 2010). A Figura 10 apresenta a representação de partições de Label Powerset para $\mathcal{D}_e$. Cada novo rótulo criado pela transformação pode ser considerado um subconjunto de rótulos de uma partição.

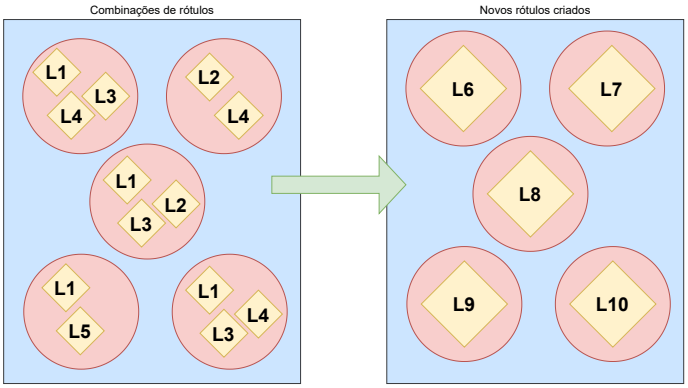


Figura 10 – Partição Label Powerset para $\mathcal{D}_e$.

Um método de transformação de problema que diminui a complexidade e minimiza a perda de informação do Label Powerset foi proposto por Read (2008), denominado *Pruned Problem Transformation* (PPT). Nesse método, não somente as combinações de rótulos se tornam novos e únicos rótulos, mas também as combinações de rótulos distintos. Por se tratar de um método de transformação baseado em poda, elimina as combinações de rótulos infrequentes de acordo com um limite[6]. Além disso, uma combinação de rótulos frequente pode ser inserida para evitar perda de informações durante o processo (TSOUMAKAS, 2010).

[6] O parâmetro de poda (p) é o limite da poda. Por exemplo, p = 1 poda todas as instâncias onde o conjunto de rótulos é único, p = 2 poda todas as instâncias que ocorrem no máximo duas vezes, e assim por diante.

2.1.1.3 Métodos baseados em pares e rankings

Métodos baseados em pares mapeiam instâncias para gerar um ranking de rótulos. *Ranking by Pairwise Comparison* (RPC) (HÜLLERMEIER et al., 2008) é um método baseado em ranking que divide o problema original em $l(l-1)/2$ sub-problemas binários, onde cada sub-problema retem instâncias que pertencem a pelo menos um dos dois rótulos, mas não a ambos. Classificadores binários são utilizados para aprender cada sub-problema e uma nova instância é classificada submetendo-a a todos os modelos. Cada predição de um modelo é interpretada como um voto, o que gera um ranking de rótulos. O rótulo com o maior número de votos é selecionado (HÜLLERMEIER et al., 2008). Como é necessário consultar todos os modelos binários gerados em tempo de execução, isto pode levar a problemas de escalabilidade, e para casos em que l é muito alto o método pode se tornar impraticável (READ, 2010; GIBAJA; VENTURA, 2014).

Para o conjunto de dados de exemplo, dez pares de rótulos são encontrados[7] e podem ser visualizados na Tabela 9. Cada um desses dez pares corresponde a um subconjunto de rótulos de uma partição do conjunto original (Figura 11).

Tabela 9 – Pares de Rótulos para $\mathcal{D}_e$.

Par	Rótulos	Par	Rótulos
1	$L1, L2$	6	$L2, L4$
2	$L1, L3$	7	$L2, L5$
3	$L1, L4$	8	$L3, L4$
4	$L1, L5$	9	$L3, L5$
5	$L2, L3$	10	$L4, L5$

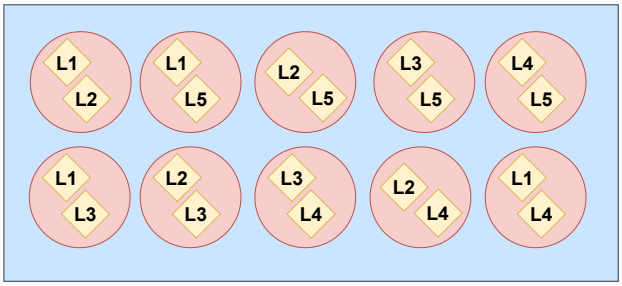


Figura 11 – Partição RPC para $\mathcal{D}_e$.

O método *CaLibrated Ranking by Pairwise Comparison* (CLR)(FÜRNKRANZ et al., 2008) estende o método *Ranking by Pairwise Comparison* adicionando um rótulo artificial que separa os rótulos relevantes dos irrelevantes para cada instância, permitindo criar ao mesmo tempo um ranking de rótulos e uma bipartição. Para cada rótulo, são consideradas instâncias positivas aquelas que possuem o rótulo em questão, e as mesmas são consideradas negativas para o rótulo artificial. De maneira semelhante, as instâncias consideradas negativas para o rótulo em questão se tornam positivas para o rótulo artificial.

[7] $5(5-1)/2 = 10$

Classificadores binários são então induzidos nos rótulos virtuais e o rótulo majoritário é selecionado (GANDA; BUCH, 2018; SOROWER, 2010).

A grande vantagem do uso da abordagem local é a flexibilidade, pois diversos tipos de métodos e algoritmos podem ser usados para resolver o problema. No entanto, essas diferentes formas de dividir o problema podem levar a perda de informação, problemas de escalabilidade, desbalanceamento, entre outros problemas, os quais podem levar a resultados ineficientes e irrelevantes. Porém, como mencionado nesta seção, existem métodos que transformam o problema multirrótulo de maneira a evitar tais deficiências e limitações. Quanto às partições, os métodos desta abordagem lidam com as locais e também com partições compostas por pares de rótulos. A Tabela 10 sumariza os métodos de transformação clássicos apresentados nesta seção.

Tabela 10 – Resumo dos métodos baseados na Abordagem Local

Método	Vantagens	Desvantagens	Transformação
BR	Flexibilidade , simplicidade, escalabilidade e completude	Modelagem de correlações e desbalanceamento	Baseado em Rótulo
CC	Modelagem de Correlações, paralelizável	Escalabilidade	Baseado em BR
CLR	Modelagem de Correlações	Escalabilidade	Baseado em Pares
LP	Modelagem de Correlações	Desbalanceamento, incompletude, escalabilidade	Criação de Rótulos
PPT	Modelagem de Correlações e escalabilidade	Incompletude	Baseado em LP
RPC	Modelagem de Correlações	Escalabilidade	Baseado em Pares

2.1.2 Abordagem Global

Na abordagem global novos algoritmos são desenvolvidos, ou algoritmos existentes são adaptados, para o problema de classificação multirrótulo original, lidando com todas as classes do problema ao mesmo tempo. Nesse caso, algoritmos convencionais podem ser utilizados, como mostra a Figura 12, ou ainda outros pouco explorados (CARVALHO; FREITAS, 2009). Em termos de partição, conforme definido neste trabalho, os métodos da abordagem global, em geral, trabalham com a partição global. No entanto há casos em que, dependendo da modificação feita, as partições podem ser locais. Nas árvores de decisão, por exemplo, cada nó folha, que geralmente corresponde a um único rótulo, pode ser um subconjunto de uma partição local. Enquanto na abordagem local os dados se ajustam ao algoritmo, na abordagem global o algoritmo se ajusta aos dados.

Diversos algoritmos podem ser encontrados na literatura para cada tipo de modelo de classificação ilustrado na Figura 12. Entre os *Métodos Baseados em Árvores de Decisão* (VILLE, 2013) podem ser citados ML-C4.5 (CLARE; KING, 2001) e PCTs (BLOCKEEL; RAEDT; RAMON, 1998). Clare e King (2001) adaptaram o algoritmo C4.5 para lidar com múltiplos rótulos. A pesquisa tinha como objetivo classificar genes de acordo com as suas funções. No ML-C4.5 a medida de entropia foi modificada para considerar

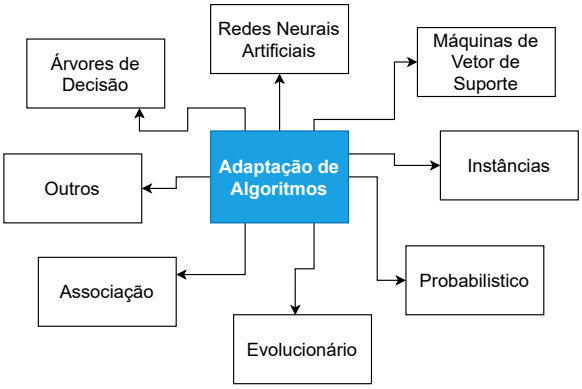


Figura 12 – Estratégias baseadas na abordagem Global. Fonte: Elaborado pela autora com base em Madjarov et al. (2012) e Zhang e Wu (2014).

a não associação de instâncias a um rótulo e as folhas das árvores alteradas para conter múltiplos rótulos (GIBAJA; VENTURA, 2014).

Predictive Clustering Trees (BLOCKEEL; RAEDT; RAMON, 1998) consideram árvores de decisão como uma hierarquia de clusters (grupos). Usando uma estratégia top-down, os dados são particionados e a variação intra-cluster é minimizada. Neste algoritmo, a métrica de distância e a função protótipo[8] são considerados parâmetros, o que permite que as PCTs sejam instanciadas para uma tarefa específica. Além disso, elas também conseguem trabalhar com séries temporais, classificação hierárquica multirrótulo, entre outros tipos de estruturas (BLOCKEEL; RAEDT; RAMON, 1998; MADJAROV et al., 2012). Devido a fácil interpretabilidade, métodos baseados em árvores de decisão têm sido muito utilizados.

Back Propagation Multi-Label Learning (BP-MLL)(ZHANG; ZHOU, 2006), *Multi-Class Multi-label Perceptron* (MMP)(CRAMMER; SINGER, 2003) e CascadeML (PAKRASHI; NAMEE, 2019) são métodos baseados em Redes Neurais Artificiais (RNAs) (HAYKIN, 2011). BP-MLL adapta a função de erro utilizada no algoritmo *Missing Label Prediction* (MLP)(HAYKIN, 2011) para dar suporte a múltiplos rótulos. *Multi-class Multi-layer Perceptron* é uma família de algoritmos online[9] capaz de ranquear rótulos baseado em perceptrons. De maneira semelhante ao Binary Relevance que usa um classificador para cada rótulo do conjunto, MMP usa um perceptron para cada rótulo, mas para a atualização de cada perceptron considera o desempenho do conjunto como um todo (GIBAJA; VENTURA, 2014; SOROWER, 2010).

O método CascadeML evolui automaticamente a rede neural e o algoritmo de treinamento para classificação multirrótulo. O algoritmo utiliza a função de erro de BP-MLL e

[8] A função protótipo calcula o protótipo de um conjunto de instâncias. Um protótipo é uma instância de dados representativa dos dados. Por exemplo, o protótipo pode ser calculado pela média do conjunto de instâncias (HASTIE; TIBSHIRANI; FRIEDMAN, 2001).

[9] Este tipo de algoritmo trabalha com entrada de dados parcial (apenas uma parte está disponível) pois alguns dados de entrada chegam apenas no futuro, portanto, esses dados futuros não estão disponíveis no momento para serem utilizados (ALBERS, 2003).

considera as correlações entre os rótulos. Em um processo de duas fases, CascadeML aumenta a arquitetura da rede neural incrementalmente, conforme aprende os pesos usando o algoritmo de gradiente de primeira ordem adaptativo, não sendo necessário informar o número de camadas ocultas, nós e taxa de aprendizado (PAKRASHI; NAMEE, 2019).

Entre os métodos baseados em Máquinas de Vetores de Suporte (*Support Vector Machines* (SVMs)) (CERVANTES et al., 2020) o mais popular é o *Rank-SVM* (ELISSEEFF; WESTON, 2001). O algoritmo minimiza a função de ranking loss[10] usando um conjunto de classificadores lineares, os quais também lidam com casos não lineares utilizando funções de kernel (ZHANG; WU, 2014).

Zhang e Zhou (2007) propuseram o *Multi-Label k-Nearest Neighboors* (ML-kNN), um método baseado em instâncias. Este algoritmo adapta técnicas do *k-Nearest Neighboors* (k-NN) (ALTMAN, 1992) para lidar com múltiplos rótulos. Primeiro, para cada instância, o algoritmo encontra os k vizinhos mais próximos e em seguida conta o número de ocorrências de cada rótulo na vizinhança. A contagem é então combinada com as probabilidades anteriores de cada rótulo para realizar as predições (READ, 2010). Assim como o MMP, ML-kNN também se assemelha ao processo realizado pelo Binary Relevance pois realiza a contagem separada para cada rótulo.

Modelos probabilísticos generativos são focados na categorização de texto. Ueda e Saito (2003) propuseram um método deste tipo para classificação multirrótulo, denominado *Parametric Mixture Models* (PMM). Para automatizar a classificação dos documentos de texto, PMM estima as probabilidades de rótulos a partir dos termos que aparecem no documento (HERRERA et al., 2016).

Goncalves, Plastino e Freitas (2013) propuseram o *Genetic Algorithm for ordering Classifier Chains* (GACC), um método evolucionário que otimiza a ordem dos classificadores em cadeia e tenta tornar o modelo obtido mais interpretável. Outro método que pode ser citado é o *Multi-Label Ant-Miner* (MuLAM) proposto por Chan e Freitas (2006) que é uma extensão do algoritmo baseado em colônias de formigas e trata o problema multirrótulo como uma tarefa de otimização (HERRERA et al., 2016). O trabalho de Goncalves, Freitas e Plastino (2018) apresenta uma categorização das abordagens de algoritmos evolucionários para classificação multirrótulo encontradas na literatura.

Métodos Associativos integram regras de associação e classificação. O *Multi-Class Multi-Label Associative Classification* (MMAC) é um método associativo proposto por Thabtah, Cowling e Peng (2004). O algoritmo começa minerando um conjunto inicial de regras no conjunto de treinamento e em seguida remove as instâncias associadas a este conjunto. A partir das instâncias restantes, um novo conjunto de regras de associação é minerado. Este processo é repetido até que não haja mais itens frequentes. Ao final, cada conjunto de regras encontrado é mesclado e uma nova instância é classificada de acordo com o suporte da regra que se aplica a ela (SOROWER, 2010).

[10] Porcentagem de pares de rótulos que são ordenados incorretamente (READ, 2010)

Os métodos apresentados mostraram que ao se criar novos algoritmos, aspectos como desbalanceamento, escalabilidade, dimensionalidade e modelagem de correlações podem ser consideradas desde o princípio da construção de um novo algoritmo, o que não necessariamente significa que não ocorrerão. Na modificação de um algoritmo existente, alguns desses aspectos podem se sobressair, ou outros problemas podem surgir, e será necessário adicionar recursos para solucioná-los.

Os métodos deste tipo de abordagem trabalham geralmente com partições globais. Há ainda outros algoritmos que utilizam métodos que não se enquadram na Figura 12. A Tabela 11 apresenta o sumário dos métodos listados nesta seção, onde o SIM indica métodos que modelam e exploram as correlações explicitamente e o NÃO são métodos que não o fazem. No entanto, como os métodos globais lidam com todos os rótulos ao mesmo tempo, eles conseguem identificar algumas correlações inerentes aos dados, mas não signfica que conseguirão explorá-las e usufruir corretamente deste conhecimento, como já foi explicado no Capítulo 1.

Tabela 11 – Resumo dos Métodos Basados na Abordagem Global.

Método	Partição	Modelagem de Correlações	Algoritmo
BP-MLL	Global	Sim	RNAs
GACC	Global	Sim	Evolutivo
ML-C4.5	Global	Não	Árvores de Decisão
ML-kNN	Local	Não	kNN
MMAC	Global	Não	Associativo
MMP	Local	Não	RNAs
MuLAM	Global	Não	Evolutivo
PCT	Global	Não	Árvores de Decisão
PMM	Global	Sim	Probabilístico
Rank-SVM	Global	Sim	SVM

2.2 Combinação de Classificadores Multirrótulo

Tanto para os métodos da abordagem global, quanto para os métodos da abordagem local, é possível utilizar uma combinação de classificadores multirrótulo (*Ensemble of Multi Label Classifiers* (EMLCs)) para tentar melhorar a classificação, o ranqueamento multirrótulo, diminuir o *overfitting*[11] ou o *underfitting*[12], modelar correlações, minimizar o desbalanceamento e a escalabilidade (HERRERA et al., 2016).

Para a classificação multirrótulo, alguns métodos independentes e dependentes de algoritmo fazem uso de alguma combinação de classificadores. Portanto, considera-se combinações de classificadores apenas aqueles métodos que combinam vários métodos e são capazes de lidar com dados multirrótulo (MOYANO et al., 2018). Em uma combinação de classificadores, cada classificador base é executado separadamente, e então as predições de cada um são agregadas para realizar uma predição final. O próprio método *Binary*

[11] o modelo de aprendizado se adapta muito bem aos dados de treinamento, mas não generaliza bem para novos dados

[12] o modelo de aprendizado não se adapta aos dados de treinamento

Relevance, por exemplo, combina vários classificadores binários para resolver o problema multirrótulo.

Um EMLC clássico baseado em classificadores binários é o *Ensemble of Classifier Chains* (ECC) (READ et al., 2009; READ; PFAHRINGER, 2021), o qual possui o CC como classificador base. No método CC, a cadeia de classificadores é construída em uma ordem específica, portanto a propagação de erros de classificação ao longo da cadeia impacta diretamente o resultado. O método ECC foi proposto para corrigir este problema, permitindo que a cadeia de classificadores seja construída de forma aleatória. Cada CC que compõe o ECC é construído sobre uma seleção aleatória de instâncias com substituição. Um conjunto de predições é obtido para cada rótulo, os quais são considerados como votos e um ranking é gerado (MOYANO et al., 2020; MADJAROV et al., 2012). Ensemble of Classifier Chains trabalha com partição local pois herda essa característica do CC, o qual também é baseado no BR, um método de partição local.

O método proposto por Tsoumakas e Vlahavas (2007) denominado *Random k-labELsets* (RAkEL) é baseado no método criação de rótulos. RAkEL constrói um conjunto de classificadores Label Powerset em que cada classificador é treinado para um pequeno - e diferente - subconjunto de rótulos aleatórios (k-labelsets), o que permite aprender todos os conjuntos de rótulos do conjunto de dados e modelar correlações.

Dois parâmetros do RAkEL devem ser ajustados: o número de classificadores n, e o número de conjuntos de rótulos k desejados. Se $k = 1$ e $n = |\mathcal{L}|$, então o comportamento será igual ao método Binary Relevance, e se $k = |\mathcal{L}|$ e $n = 1$, então o comportamento será como o *Label Powerset*. As predições são combinadas e o sistema de votos majoritário é utilizado para cada rótulo, gerando um ranking (MOYANO et al., 2018; ROKACH; SCHCLAR; ITACH, 2014). RAkEL gera partições locais pois é baseado no método Label Powerset. Além disso, RAkEL exige dois parâmetros, o número de classificadores e número de conjuntos de rótulos, os quais limitam a criação livre e diversa das partições que estão entre as locais e globais.

Outro EMLC clássico baseado em rótulos, denominado *Ensemble of Pruned Sets* (EPS), foi proposto por (READ, 2008), o qual estende o método PPT, corrige o problema de *overffiting* durante a poda, e treina N modelos independentes, cada qual sobre um subconjunto do conjunto de treinamento sem substituição. Assim como RAkEL, EPS também aprende todos os conjuntos de rótulos e modela correlações.

Kocev et al. (2007) propôs o *Random Forest of Predictive Clustering Trees* (RF-PCT), um EMLC baseado no algoritmo de Predictive Clustering Trees. A técnica de bagging (BREIMAN, 1996) é usada para selecionar diferentes conjuntos de instâncias para cada classificador e, para cada nó da árvore, RF-PCT seleciona a melhor característica de um subconjunto aleatório de instâncias, fornecendo diversidade para os classificadores base. Uma instância de teste é classificada por meio da média dos valores de confiança de todos os classificadores para cada rótulo.

Moyano (2019) propôs um EMLC baseado em métodos evolucionários denominado *Evolutionary Algorithm Multi-Label* (EME). EME gera automaticamente conjuntos de classificadores multirrótulo que consideram o desbalanceamento, a modelagem de correlações e a alta dimensionalidade do espaço de rótulos em sua construção. Cada classificador portanto aprende as características de um subconjunto de rótulos aleatório, o qual evolui com o algoritmo e busca combinações de melhor desempenho. Para modelar as correlações, um operador de mutação busca por indivíduos onde os rótulos estão mais relacionados. O desbalanceamento é tratado com uma função que considera tanto o desempenho preditivo quanto a quantidade de vezes que cada rótulo é considerado no conjunto. Por fim, os conjuntos evoluem selecionando os classificadores com base no desempenho geral.

Métodos baseados em conjuntos tentam melhorar o desempenho preditivo final, além de permitir a modelagem de correlações entre rótulos, facilitando a escalabilidade e paralelização. Apesar disto, de acordo com Moyano et al. (2020), selecionar um classificador base para um EMLC não é trivial e, algumas vezes, esta escolha pode levar o EMLC a ter desempenho inferior ao de um classificador base. Mesmo que EMLCs facilitem a escalabilidade, há casos em que isso pode não ocorrer. Isso dependerá se o classificador base escolhido é capaz de lidar com um grande número de rótulos. Como os EMLCs lidam com o tipo de partição do método base, os mesmos não lidam com o conceito de partições híbridas introduzido nesta pesquisa.

Há ainda outros EMLCs que utilizam métodos que não se enquadram na Figura 12 os quais podem ser encontrados na Tabela 12, e que também sumariza os métodos apresentados nesta subseção. Tanto os métodos dependentes de algoritmo, quanto os independentes, e também os EMLCs, podem aplicar diferentes técnicas para explorar as correlações entre rótulos. Portanto, a seção 2.3 apresentará algumas definições e métodos propostos na literatura para modelar correlações entre rótulos.

Tabela 12 – Resumo dos EMLCs.

Método	Partição	Modelagem Correlações	Algoritmo
ECC	Local	Sim	CC
EME	Local	Sim	Evolutivo
EPS	Local	Sim	PPT
RAkEL	Local	Sim	LP
RF-PCT	Local	Sim	PCT

2.3 Correlações entre Rótulos

Em todos os problemas multirrótulo, de acordo com Read (2010), existem correlações entre os rótulos. Trabalhos como os dos autores (BAREZI; KWOK; RABIEE, 2017) e (ZHU; KWOK; ZHOU, 2018) mostraram que explorar as correlações entre rótulos aumenta o poder preditivo do classificadores. A modelagem das correlações entre rótulos

pode facilitar as predições, ou seja, um rótulo pode ser predito corretamente devido à sua correlação com outro rótulo.

A literatura traz algumas propostas para organizar e categorizar diferentes tipos de correlações entre rótulos. As correlações entre rótulos são classificadas como correlações globais ou locais por Huang e Zhou (2012). As correlações globais assumem que as correlações entre rótulos são compartilhadas por todas as instâncias do conjunto de dados, ou seja, dois ou mais rótulos são correlacionados se classificarem todas as instâncias do conjunto de dados. As correlações locais assumem que as correlações de rótulos só podem ser compartilhadas por um subconjunto de instâncias, não todas, e que dois ou mais rótulos são correlacionados se classificarem um subconjunto de instâncias. O método *Multi-Label Learning by Exploiting Label Correlations Locally* (ML-LOC)(HUANG; ZHOU, 2012) é um exemplo de método que explora correlações locais, enquanto *Global and local multi-label correlations* (GLOCAL) (ZHU; KWOK; ZHOU, 2018) é um exemplo de método que explora correlações locais e globais.

Os autores usaram dependência condicional e incondicional para representar correlações entre rótulos no trabalho apresentado em (DEMBCZYŃSKI et al., 2012). A dependência condicional captura as dependências entre os rótulos em uma instância específica e prediz a probabilidade de os rótulos ocorrerem juntos. A dependência incondicional modela a probabilidade de certos rótulos ocorrerem juntos em todo o conjunto de dados.

Zhang e Wu (2014) introduziram o conceito de correlações entre rótulos de primeira ordem, segunda ordem e alta ordem. As correlações de primeira ordem incluem todos os métodos multirrótulo que ignoram as correlações entre os rótulos, como (ZHANG; ZHOU, 2007) e (BOUTELL et al., 2004). As correlações de segunda ordem são modeladas usando pares de rótulos, como (ELISSEEFF; WESTON, 2001) e (FÜRNKRANZ et al., 2008). Por fim, as correlações de alta ordem são modeladas usando todos os rótulos ou subconjuntos de rótulos, como em (TSOUMAKAS; VLAHAVAS, 2007) e (READ; PFAHRINGER; HOLMES, 2008).

Alguns trabalhos modelam as correlações entre rótulos usando apenas o espaço de rótulos. Como exemplos, em (YE et al., 2015), a medida de similaridade cosseno foi aplicada ao espaço de rótulos para descobrir a relação entre os rótulos. Shi et al. (2015) usaram o algoritmo Apriori considerando rótulos como conjuntos de itens para modelar as correlações.

Outros estudos usam o espaço de atributos para modelar as correlações entre rótulos, como em (ZHANG, 2015), onde os autores propuseram um método chamado *MultiLabel Learning with Label Specific Features* (LIFT). Eles alegaram que, tal como as correlações entre rótulos, os atributos discriminativos para cada rótulo de classe podem ser usados no processo de aprendizagem. Eles empregaram abordagens de agrupamento para criar atributos distintos que capturam as propriedades exclusivas de cada rótulo. Weng et al. (2018) propos uma estratégia baseada no LIFT que aborda o problema do desequilíbrio

de classes.

Há também pesquisas que levam em consideração espaços de atributos de entrada e rótulos para descobrir correlações de rótulos, como em (LI; YANG, 2021). Os autores apresentaram uma rede neural completamente conectada que faz uso do aprendizado das correlações. No espaço de atributos de entrada, foi utilizado o pooling de covariâncias, enquanto que no espaço de rótulos, foi utilizada uma Rede Convolucional de Grafos. No trabalho de (XU et al., 2021) foi proposta uma nova estratégia de agrupamento para aprender com o espaço de atributos de entrada, e também uma rede convolucional de grafos de dois fluxos para aprender com o espaço de rótulos.

Existem certas semelhanças entre as várias formas de correlações entre rótulos apresentadas. Por lidarem com todos os rótulos, as correlações globais, as dependências condicionais e as correlações de ordem superior são comparáveis. Por considerarem grupos de rótulos, a correlação local, a dependência incondicional e as correlações de segunda ordem são comparáveis. Embora algumas comparações e semelhanças sejam identificadas, a literatura ainda não é clara ao definir diferentes tipos de correlações de rótulos. Então, pode-se resumir as características acima da seguinte forma:

- *Ordem*: categoriza as correlações de rótulos com base na ordem: primeiro (sem correlação), segundo (pares de rótulos) ou alto (todos os rótulos);
- *Instâncias*: categoriza as correlações de rótulo com base no espaço da instâncias (atributos de entrada e saída): global (todas as instâncias) ou local (subconjunto de instâncias);
- *Dependência*: categoriza as correlações de rótulo com base em probabilidades: condicional (dependente da instância) ou incondicional (conjunto de dados inteiro);
- *Espaço do conjunto de dados*: categoriza as correlações de rótulo com base no espaço do conjunto de dados: atributos de entrada, instâncias ou espaço de rótulos.

Com base nesse sumário, pode-se organizar os métodos para modelar correlações entre rótulos em quatro categorias, para caracterizar adequadamente as várias correlações: 1) que não exploram correlações; 2) que exploram correlações entre pares de rótulos; 3) que exploram correlações em agrupamentos de rótulos; e 4) que exploram correlações em todos os rótulos.

Por exemplo, a correlação global pode ser modelada da seguinte maneira: 1) usando apenas do espaço de rótulos, sem o espaço de atributos de entrada; 2) usando todo o conjunto de dados; 3) capturando correlações entre rótulos a partir da similaridade entre os atributos; 4) capturando correlações do espaço de instâncias; 5) capturando as correlações de cada espaço separadamente e combinando-as.

Algo semelhante pode ser feito para modelar correlações locais: 1) agrupando o espaço de instâncias (entrada e saída); 2) agrupando espaço de rótulos; 3) agrupando o espaço

de atributos; 4) agrupar cada espaço separadamente e combiná-los posteriormente. Além disso, é necessário considerar aqui os agrupamentos de rótulos disjuntos e não disjuntos. Medidas de similaridades, redes complexas, otimização, probabilidades e outras técnicas podem ser empregadas para capturar esses diferentes tipos de correlações.

Por exemplo, o espaço de rótulos pode ser utilizado para calcular uma medida de similaridade, que é principalmente uma medida de similaridade pareada, obtendo assim as correlações entre cada par de rótulos. Outra maneira seria agrupar o espaço de rótulos primeiro e depois aplicar alguma técnica em cada grupo para obter as correlações locais. Assim, as correlações entre rótulos podem ser exploradas mesclando essas categorizações, diferentes espaços de conjuntos de dados e técnicas.

2.4 Medidas de Avaliação

Como mencionado na introdução deste capítulo, um classificador multirrótulo gera como saída uma bipartição ou uma lista ordenada de rótulos (ranking). Diferente da classificação monorrótulo, em que a predição pode ser correta ou incorreta, a predição na classificação multirrótulo pode ser parcialmente correta, totalmente correta ou totalmente incorreta. Portanto, para a classificação multirrótulo, definir qual erro de classificação é o mais (ou menos) grave não é simples como na classificação monorrótulo. Por exemplo, uma instância de teste pode ser classificada incorretamente em dois rótulos, enquanto que outras cinco instâncias de teste podem ser classificadas corretamente em apenas um rótulo (WU; ZHOU, 2017; CHARTE et al., 2018).

Este aspecto da classificação multirrótulo levou ao desenvolvimento de várias medidas de avaliação de desempenho as quais estão sintetizadas na Figura 13. Com essa diversidade de medidas, o desempenho dos classificadores pode variar para cada uma delas. Avaliar um classificador multirrótulo usando uma única medida é restritivo, pois cada medida avalia um ou outro aspecto em particular, o que pode levar a análises e conclusões incompletas e até mesmo errôneas. O recomendado é usar um conjunto dessas medidas, obtendo assim uma visão mais coerente do desempenho sob diferentes aspectos (WU; ZHOU, 2017).

As Subseções 2.4.1 e 2.4.2 apresentam as principais medidas de avaliação para problemas multirrótulo. O símbolo $\uparrow$ antecedendo o nome da equação indica que quanto maior o valor resultante, melhor o desempenho, enquanto que o símbolo $\downarrow$ indica que quanto menor o valor resultante, melhor o desempenho.

2.4.1 Bipartições

As medidas baseadas em bipartições podem ser divididas em baseadas em rótulos e basedas em instâncias. Para a bipartição gerada pelo classificador multirrótulo, as medidas de avaliação baseadas em instâncias calculam a diferença média entre os conjuntos

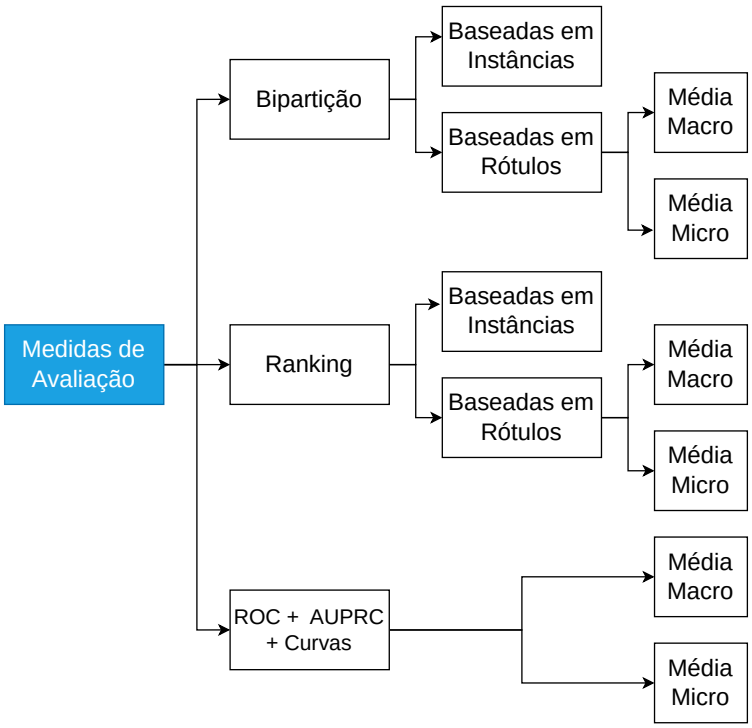


Figura 13 – Taxonomia de Medidas de Avaliação Multirrótulo. Fonte: Elaborado pela autora com base em (ZHANG; WU, 2014).

de rótulos preditos e os conjuntos de rótulos reais em cada instância do conjunto de teste. As baseadas em rótulos avaliam o desempenho preditivo para cada rótulo separadamente e depois calculam a média do desempenho de todos os rótulos. As medidas baseadas em rótulo podem ainda usar duas estratégias diferentes: *micro-averaging* (Média Micro) e *macro-averaging* (Média Macro) (MADJAROV et al., 2012; TSOUMAKAS, 2010).

2.4.1.1 Medidas Baseadas em Instâncias

A eficácia geral de um classificador multirrótulo é avaliada pela Acurácia (A), Equação 2. Esta métrica calcula a proporção do número de rótulos corretamente preditos em comparação ao número total de rótulos (verdadeiros e preditos) para uma instância, fazendo em seguida a média sobre todas as instâncias (GIBAJA, 2015; HERRERA et al., 2016).

$$\uparrow A = \frac{1}{m}\sum_{i=1}^{m}\frac{|\ Y_i \cap \hat{Y}_i\ |}{|\ Y_i \cup \hat{Y}_i\ |} \tag{2}$$

Exact Match Ratio (EMR)[13] também conhecida como *Classification Accuracy* (CA) ou *Subset Accuracy* (SA), ignora os rótulos parcialmente corretos, atuando como a acurácia da classificação monorrótulo. Como EMR avalia apenas as instâncias classificadas

[13] Proporção de correspondência exata, Acurácia de Classificação ou subconjunto de acurácia

corretamente, ela é bastante restritiva. De forma similar, a Equação 4, chamada *0/1 Loss*, mede a diferença entre os rótulos verdadeiros e os rótulos preditos, ao invés da igualdade como na Equação 3 (GIBAJA; VENTURA, 2014; PEREIRA et al., 2018).

Conhecida como *Hamming Loss* (HL), a Equação 5 avalia a fração de rótulos classificados erroneamente, isto é, quando um rótulo negativo é predito como positivo e quando um rótulo positivos é predito como negativo. Esta métrica calcula a diferença simétrica entre o conjunto de rótulos predito e o conjunto de rótulos verdadeiro, onde $Y_i \Delta \hat{Y}_i = (Y_i - \hat{Y}_i) \cup (\hat{Y}_i - Y_i)$. O valor ótimo é `HL=0`, ou seja, nenhum erro (WU; ZHOU, 2017; PEREIRA et al., 2018).

$$\uparrow EMR = \frac{1}{m} \sum_{i=1}^{m} I\Big(Y_i = \hat{Y}_i\Big) \quad (3) \qquad I = \begin{cases} 1 & \text{Se } \hat{Y}_i = Y_i \\ 0 & \text{caso contrário} \end{cases}$$

$$\downarrow 0/1L = \frac{1}{m} \sum_{i=1}^{m} I\Big(Y_i \neq \hat{Y}_i\Big) \quad (4)$$

$$\downarrow HL = \frac{1}{m} \frac{1}{l} \sum_{i=1}^{m} \mid Y_i \Delta \hat{Y}_i \mid \quad (5)$$

Duas métricas que permitem medir a eficácia de um classificador para recuperar rótulos positivos são a Precisão (P) e a Revocação (R). A primeira calcula a fração de rótulos preditos que realmente são relevantes (Equação 6), enquanto que a segunda calcula a fração de rótulos relevantes verdadeiros que também são preditos (Equação 7). A média harmônica da precisão e revocação é calculada pela Equação 8, (READ, 2010; GIBAJA; VENTURA, 2014; READ, 2010; PEREIRA et al., 2018) sendo denominada F1.

$$\uparrow P = \frac{1}{m} \sum_{i=1}^{m} \frac{\mid Y_i \cap \hat{Y}_i \mid}{\hat{Y}_i} \quad (6) \qquad \uparrow R = \frac{1}{m} \sum_{i=1}^{m} \frac{\mid Y_i \cap \hat{Y}_i \mid}{Y_i} \quad (7)$$

$$\uparrow F1 = \frac{1}{m} \sum_{i=1}^{m} \frac{2 \mid Y_i \cap \hat{Y}_i \mid}{\mid \hat{Y}_i \mid + \mid Y_i \mid} \quad (8)$$

2.4.1.2 Medidas Baseadas em Rótulos

Para as medidas baseadas em rótulos considere l o número total de rótulos; y_j o j-ésimo rótulo; tp_j (*true positives*) o número de rótulos verdadeiros positivos para y_j; tn_j (*true negatives*) o número de rótulos verdadeiros negativos para y_j; fp_j (*false positives*) o número de rótulos falsos positivos para y_j e fn_j (*false negatives*) o número de rótulos falsos negativos para y_j (HERRERA et al., 2016).

Na Equação 9 (Precisão-Macro (PMa)), Equação 10 (Revocação-Macro (RMa)) e Equação 11 (Macro-F1 (F1Ma)), as medidas média-macro primeiro calculam o desempenho de cada rótulo individualmente e somente depois a média entre todos os rótulos é calculada, atribuindo assim pesos iguais para os rótulos - independente se o rótulo

é frequente, infrequente ou raro. Nas Equações média-micro - Equação 12: Precisão-Micro (PMi), Equação 13: Revocação-Micro (RMi) e Equação 14: Micro-F1 (F1Mi) - os rótulos são calculados todos juntos. O desempenho de rótulos raros acaba por influenciar medidas média-macro, enquanto que as medidas média-micro são mais influenciadas pelos rótulos mais comuns (HERRERA et al., 2016).

Para mensurar predições errôneas de rótulos, três medidas foram propostas por Rivolli, Soares e Carvalho (2018). A primeira medida apresentada na Equação 15, denominada *Wrong Label Prediction* (WLP), mede quando o rótulo pode ser predito para algumas instâncias, mas essas predições estão sempre erradas. A Equação 16 permite calcular a proporção de rótulos que nunca são preditos sendo denominada MLP. A terceira métrica, *Constant Label Problem* (CLP), mede quando o mesmo rótulo é predito para todas as instâncias (RIVOLLI; SOARES; CARVALHO, 2018). Para CLP, WLP, e MLP[14] o valor de retorno ideal é zero, indicando que não há ocorrências destes problemas nas predições dos rótulos.

$$\uparrow PMa = \frac{1}{\mid l \mid} \sum_{j=1}^{\mid l \mid} \frac{tp_j}{tp_j + fp_j} \quad (9) \qquad \uparrow RMa = \frac{1}{\mid l \mid} \sum_{j=1}^{\mid l \mid} \frac{tp_j}{tp_j + fn_j} \quad (10)$$

$$\uparrow F1Ma = \frac{2 \times PMa \times RMa}{PMa + RMa} \quad (11) \qquad \uparrow PMi = \frac{\sum_{j=1}^{l} tp_j}{\sum_{j=1}^{l} tp_j + \sum_{j=1}^{l} fp_j} \quad (12)$$

$$\uparrow RMi = \frac{\sum_{j=1}^{l} tp_j}{\sum_{j=1}^{l} tp_j + \sum_{j=1}^{l} fn_j} \quad (13) \qquad \uparrow F1Mi = \frac{2 \times PMi \times RMi}{PMi + RMi} \quad (14)$$

$$\downarrow WLP = \frac{1}{l} \sum_{j=1}^{l} I\Big(tp_j == 0\Big) \quad (15) \qquad \downarrow MLP = \frac{1}{l} \sum_{j=1}^{l} I\Big(tp_j + fp_j == 0\Big) \quad (16)$$

$$\downarrow CLP = \frac{1}{l} \sum_{j=1}^{l} I\Big(tn_j + fn_j == 0\Big) \quad (17)$$

2.4.2 Ranking

Para as equações desta subseção considere L um rótulo de $\mathcal{L}$[15], $f(\mathbf{x}_i, L)$ uma função de valor real que retorna a confiança de L ser um rótulo de $\mathbf{x}_i$ e $rank(\mathbf{x}_i, L)$ uma função que, para uma instância $\mathbf{x}_i$, mapeia o valor real de $f(\mathbf{x}_i, L)$ para a posição do rótulo ($L \in \mathcal{L}$) no ranking. Um rótulo L_1 é ranqueado em uma posição mais alta que um outro rótulo L_2 se $f(\mathbf{x}_i, L_1) > f(\mathbf{x}_i, L_2)$, o que implica $rank(\mathbf{x}_i, L_1) < rank(\mathbf{x}_i, L_2)$. Considere ainda que $rank^*(\mathbf{x}_i, L)$ é o ranking verdadeiro; $\overline{Y}_i$ é o conjunto complementar de Y_i com respeito à $\mathcal{L}$; $[[\pi]]$ uma função que retorna 1 se a proposição p é verdadeira e 0 caso contrário; e, por

[14] Problema de rótulo constante (clp), errado (wlp) e faltante (mlp)
[15] conforme definido no início deste capítulo $\mathcal{L} = \{L_1, L_2, ..., L_l\}$ e $Y \subseteq \mathcal{L}$

fim, $\mathbf{x}'$, $\mathbf{x}''$, L' e L'' duas instâncias e dois rótulos, respectivamente, não necessariamente diferentes.

2.4.2.1 Medidas Baseadas em Instâncias

A Precisão-Média (PM) - Equação 18 - permite obter, em média, o número de posições que precisam ser verificadas antes de um rótulo não relevante ser encontrado. Portanto, determina para cada rótulo em uma instância, a proporção de rótulos relevantes que são ranqueados acima dela no ranking predito (TSOUMAKAS, 2010; SOROWER, 2010; MADJAROV et al., 2012; ZHANG; WU, 2014; GIBAJA; VENTURA, 2014; GIBAJA, 2015; HERRERA et al., 2016; WU; ZHOU, 2017; PEREIRA et al., 2018).

$$\uparrow PM = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{\mid Y_i \mid}\sum_{L\in Y_i}\frac{\mid \{L' \in Y_i | rank_i(L') \leq rank_i(L)\} \mid}{rank_i(L)} \tag{18}$$

A Equação 19 - Cobertura (C) - conta o número de passos necessários para percorrer o ranking fornecido até que todos os rótulos relevantes sejam encontrados, retornando o número médio de passos. Esta medida é influenciada pelo tamanho do espaço de rótulos de cada conjunto de dados multirrótulo. Quanto maior o espaço de rótulos, maior o número de passos para se percorrer (TSOUMAKAS, 2010; SOROWER, 2010; MADJAROV et al., 2012; ZHANG; WU, 2014; GIBAJA; VENTURA, 2014; GIBAJA, 2015; HERRERA et al., 2016; WU; ZHOU, 2017; PEREIRA et al., 2018).

$$\downarrow C = \frac{1}{m}\sum_{i=1}^{m}\max_{L\in Y_i} rank_i(L) - 1 \tag{19}$$

É possível verificar se o ranqueamento está correto pela Equação 20. Se *Is Error* (IE) retornar 0, o ranqueamento está correto, caso contrário retorna 1, independente do quão incorreto ou correto esteja o ranking (GIBAJA; VENTURA, 2014; GIBAJA, 2015).

$$IE = \frac{1}{m}\sum_{i=1}^{m}\left[\left[\sum_{L\in\mathcal{L}} \mid rank_i^*(L) - rank_i(L) \mid \neq 0\right]\right] \tag{20}$$

O número de posições entre os rótulos positivos e negativos com pior ranking pode ser calculado pela Equação 21. *Margin Loss* (MLoss) está relacionada ao ranqueamento incorreto de rótulos (GIBAJA; VENTURA, 2014; GIBAJA, 2015).

$$\downarrow MLoss = \frac{1}{m}\sum_{i=1}^{m}\max\Big(0, \max(\{rank(L) \mid L \in Y_i\}) - \min(\{rank(L') \mid L' \notin Y_i\})\Big) \tag{21}$$

Para medir o número de vezes que um rótulo no topo do ranking não está no conjunto de rótulos relevantes para a instância, utiliza-se a Equação 22 - *One Error* (OE) - que avalia também a fração de instâncias cujo rótulo mais confiável é irrelevante. Enquanto a métrica cobertura (Equação 19) considera o rótulo menos relevante entre todos os rótulos

relevantes, one error considera somente o rótulo mais relevante (TSOUMAKAS, 2010; SOROWER, 2010; MADJAROV et al., 2012; ZHANG; WU, 2014; GIBAJA; VENTURA, 2014; GIBAJA, 2015; HERRERA et al., 2016; WU; ZHOU, 2017; PEREIRA et al., 2018).

Ranking Error (RE) retorna a soma do quadrado das diferenças de posição para cada rótulo no ranking predito e no ranking verdadeiro. Se o ranking for idêntico ao verdadeiro, RE=0, se for invertido RE=1 (GIBAJA; VENTURA, 2014; GIBAJA, 2015).

$$\downarrow OE = \frac{1}{m}\sum_{i=1}^{m}\delta(\arg\min_{L\in\mathcal{L}} rank_i(L)) \quad (22) \qquad \delta(L) = \begin{cases} 1 & L \notin Y_i \\ 0 & \text{caso contrário} \end{cases}$$

$$\downarrow RE = \frac{1}{m}\sum_{i=1}^{m}\sum_{L\in\mathcal{L}} \mid rank_i^*(L) - rank_i(L) \mid^2 \quad (23)$$

O número de vezes que rótulos irrelevantes são classificados acima de rótulos relevantes é calculado pela Equação 24, a qual é denominada *Ranking Loss* (RL). A métrica considera todas as possíveis combinações de rótulos relevantes (L_j) e não relevantes (L_k) para uma instância e conta quantas vezes um rótulo não relevante é ranqueado acima de um rótulo relevante na predição (TSOUMAKAS, 2010; SOROWER, 2010; MADJAROV et al., 2012; ZHANG; WU, 2014; GIBAJA; VENTURA, 2014; GIBAJA, 2015; HERRERA et al., 2016; WU; ZHOU, 2017; PEREIRA et al., 2018).

$$\downarrow RL = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{\mid Y_i \mid\mid \overline{Y_i} \mid} \mid E \mid \quad (24)$$

$$E = \left\{(L_j, L_k) \mid rank_i(L_j) > rank_i(L_k), (L_j, L_k) \in Y_i \times \overline{Y_i}\right\}$$

2.4.2.2 Medidas Baseadas em Rótulos

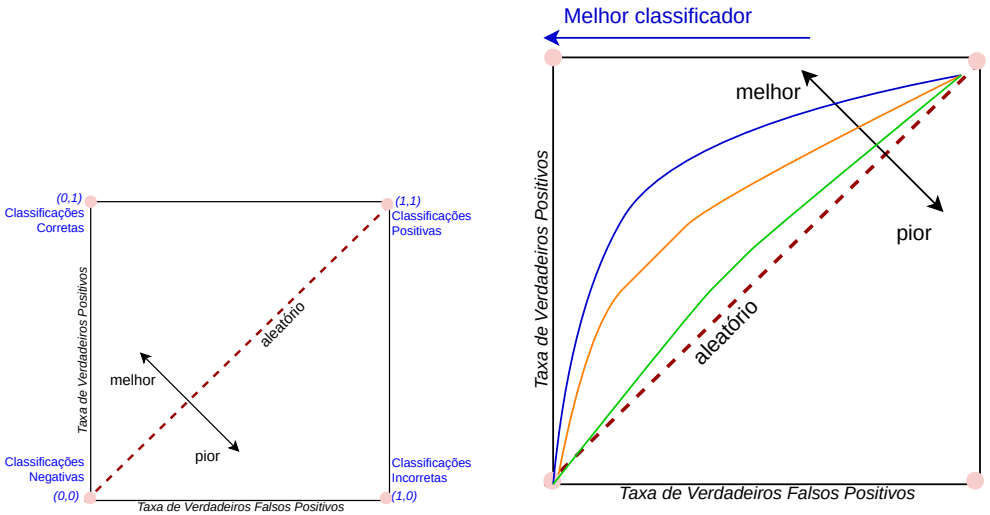
As medidas de avaliação de ranqueamento que serão apresentadas nesta seção não utilizam uma matriz de confusão como as medidas apresentadas na subseção 2.4.2.2. Estas medidas de ranking são baseadas no gráfico *Receiver Operating Characteristic* (ROC) que é um gráfico bidimensional onde o eixo X representa a taxa de falsos positivos e o eixo Y a taxa de verdadeiros positivos, um ponto representa um modelo de classificação e o ponto é calculado pela taxa de verdadeiros positivos e falsos positivos (matriz de confusão). O gráfico ROC pode ser utilizado como uma ferramenta para visualizar, organizar e selecionar classificadores com base em seu desempenho (FACELI et al., 2011; GUVENIR; KURTCEPHE, 2013).

Um classificador que produz um único rótulo como saída gera apenas um ponto no espaço ROC, enquanto que para os classificadores que produzem valores reais ou probabilidades, vários pontos são gerados, o que resulta em uma curva (SILVA, 2006). A *Area Under the Curve* (AUC) - Area abaixo da curva - é derivada da curva ROC medindo a área bidimensional abaixo de toda a curva ROC retornando um valor entre 0 e 1, que

mostra como o modelo se comporta para diferentes valores de limiares (FAWCETT, 2006; FACELI et al., 2011; ALER; HANDL; KNOWLES, 2013).

A medida Média-AUC-ROC veio da necessidade de se calcular a média dos pontos ROC usando uma variável independente (pois seu valor pode ser controlado diretamente) como, por exemplo, a lista de valores reais retornada por um modelo de ranqueamento. Neste caso, a amostragem do gráfico é feita com base no ranqueamento, ao invés da posição no espaço, e para cada threshold é calculado primeiro o ponto correspondente na curva e em seguida a média entre eles (BRADLEY, 1997; FAWCETT, 2006).

A Figura 14(a) ilustra o gráfico ROC, enquanto que a Figura 14(b) ilustra o gráfico da CURVA-ROC. Recapitulando e finalizando, um classificador é considerado melhor do que o outro se o seu ponto no espaço ROC estiver acima e à esquerda do respectivo ponto do segundo classificador. A Curva ROC é plotada através das predições probabilísticas, e não das predições bińarias. No entanto, as predições probabilísticas se tornam predições binárias após passar por um corte (threshold) que decide se aquela predição se tornará o valor 1 ou 0.



(a) ROC (b) CURVA-ROC

Figura 14 – Exemplo do gráfico ROC e de gráfico da CURVA-ROC. Elaborado pela autora com base em (DAVIS; GOADRICH, 2006; FAWCETT, 2006; BRADLEY, 1997; COOK; RAMADAS, 2020)

Exemplo: se ($\hat{y}$ >=0.5) então $\hat{y} = 1$, caso contrário $\hat{y} = 0$. O melhor corte pode ser escolhido através do teste de vários valores e plotando o gráfico da curva para verificar qual se aproxima mais do melhor resultado. A área embaixo da curva ROC é uma medida extraída da curva ROC. Quanto mais próximo de 1, melhor o resultado. Exemplo: comparando-se 3 algoritmos, 3 curvas são geradas e podem ser comparadas. Assim, se a AUC-ROC for: classificador 1 = 0.8, classificador 2 = 0.6, e classificador 3 = 0.9, então o

classificador 3 é superior com relação aos outros dois classificadores. É possível também calcular as médias-micro-macro para as curvas ROC.

Dentro deste contexto, e de acordo com Herrera et al. (2016), a Média-AUC-Macro-ROC e a Média-AUC-Micro-ROC, para classificação multirrótulo, podem ser calculadas conforme as Equações 25 e 26.

$$\uparrow AUC_{macro_roc} = \frac{1}{l}\sum_{j=1}^{l} \frac{\mid \{(\mathbf{x}', \mathbf{x}'') \mid f(\mathbf{x}', L_j) \geq f(\mathbf{x}'', L_j), (\mathbf{x}', \mathbf{x}'') \in Z_j \times \overline{Z_j}\} \mid}{\mid Z_j \mid\mid \overline{Z_j} \mid} \tag{25}$$

$$\text{onde: } Z_j = \{\mathbf{x}_i \mid L_j \in Y_i, 1 \leq i \leq m\} \quad \text{e} \quad \overline{Z_j} = \{\mathbf{x}_i \mid L_j \notin Y_i, 1 \leq i \leq m\}$$

$$\uparrow AUC_{micro_roc} = \frac{\mid \{(\mathbf{x}', \mathbf{x}'', L', L'') \mid f(\mathbf{x}', L') \geq f(\mathbf{x}'', L''), (\mathbf{x}', L') \in S^+, (\mathbf{x}'', L'') \in S^-\} \mid}{\mid S^+ \mid\mid S^- \mid} \tag{26}$$

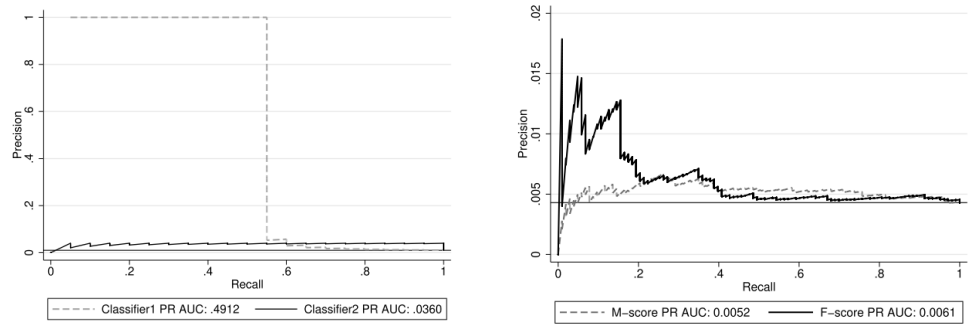
$$\text{onde: } S^+ = \{(\mathbf{x}_i, L) \mid L \in Y_i, 1 \leq i \leq m\} \quad \text{e} \quad S^- = \{(\mathbf{x}_i, L) \mid L \notin Y_i, 1 \leq i \leq m\}$$

2.4.3 AUPRC

A área sob a curva de precisão e revocação (**A**rea **U**nder the **P**recision-**R**ecall **C**urve - AUPRC) pode mostrar o trade-off entre precisão e revocação em todos os possíveis valores de limite. As medidas AUPRC-Micro e AUPRC-Macro calculam a precisão média das predições probabilísticas. A precisão média resume uma curva de precisão/revocação como a média ponderada das precisões alcançadas em cada limite, com o aumento na revocação do limite anterior usado como peso.

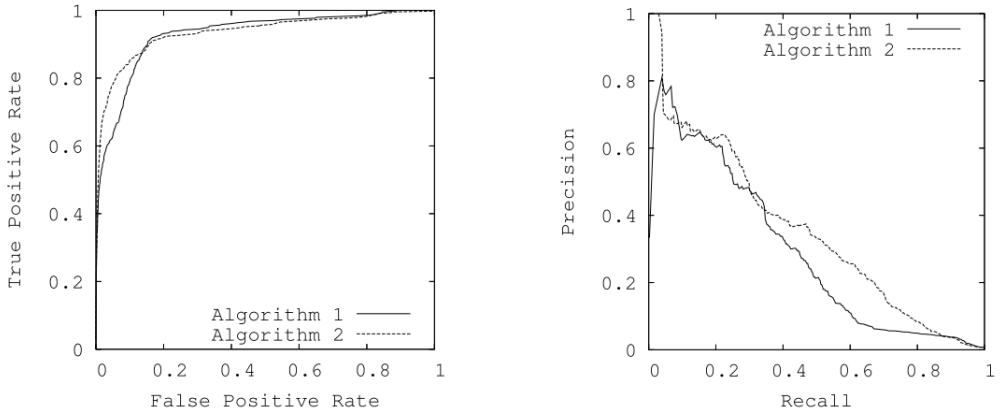
No método de micro-média, as métricas são calculadas globalmente, ou seja, agregam as contribuições de todas as classes e depois calcula a média, enquanto no método de macro-média, as métricas são calculadas para cada rótulo individualmente primeiro e depois a média não ponderada é calculada. Portanto, a AUPRC é um número único que sumariza a informação na curva de precisão e revocação, mostrando a precisão em função da revocação.

A compensação entre precisão e revocação pode ser útil na seleção de um limite, bem como na avaliação do modelo (OZENNE; SUBTIL; MAUCORT-BOULCH, 2015; COTTAM et al., 2020; ZHOU et al., 2021; HERRERA et al., 2016). A Figura 15 apresenta duas curvas de revocação-precisão, as quais foram retiradas do artigo de (COOK; RAMADAS, 2020). Já a Figura 16 apresenta a diferença entre os gráficos plotados no espaço ROC e no espaço de revocação-precisão, a qual foi retirada do artigo de (DAVIS; GOADRICH, 2006)



(a) Exemplo 1 (b) Exemplo 2

Figura 15 – Exemplos de gráficos da curva de precisão e revocação. Fonte: (COOK; RAMADAS, 2020).



(a) Comparison in ROC space (b) Comparison in PR space

Figura 16 – Exemplos de gráficos da Curva-ROC e da curva de precisão e revocação. Fonte: (DAVIS; GOADRICH, 2006).

2.5 Dimensionalidade, Escalabilidade e Desbalanceamento

Outros três aspectos importantes da classificação multirrótulo são a Dimensionalidade, Escalabilidade e Desbalanceamento. Dimensionalidade é relativo à dimensão do espaço de atributos, enquanto que escalabilidade trata do processamento de um grande número de rótulos no espaço de rótulos. O desbalanceamento refere-se basicamente à distribuição dos rótulos (READ, 2010; HERRERA et al., 2016).

Alguns conjuntos de dados multirrótulo podem apresentar alta dimensão no espaço de atributos como, por exemplo, o conjunto de dados `Wiki10-31K` (BHATIA et al., 2015) que possui 132.876 atributos sendo 101.938 atributos de entrada, 30.938 rótulos (atributos de saída) e 20.762 instâncias. Reduzir o número de atributos em conjunto de dados com alta dimensionalidade, em datasets como o `Wiki10-31K`, pode facilitar

o processamento, simplificar o modelo, tornar o modelo mais interpretável, diminuir o tempo de aprendizado e aumentar a capacidade de generalização evitando o overfitting (KASHEF; NEZAMABADI-POUR; NIKPOUR, 2018).

Para reduzir o espaço de atributos, estratégias foram propostas na literatura e uma revisão pode ser encontrada em (KASHEF; NEZAMABADI-POUR; NIKPOUR, 2018). Algumas dessas estratégias podem ser diretamente aplicáveis para a classificação multirrótulo, enquanto outras podem ser estendidas, de maneira similar ao que que ocorre com as abordagens dependente e independente de algoritmo. Seleção de atributos e extração de atributos são as principais estratégias utilizadas para realizar a redução. A seleção de atributos remove atributos redundantes ou irrelevantes - selecionando assim aqueles atributos que fornecem informação útil para a construção de um modelo - enquanto que a extração de atributos obtém novos atributos por meio de combinações e transformações do conjunto original (TSOUMAKAS, 2010; GIBAJA; VENTURA, 2014; HERRERA et al., 2016).

A seleção de atributos pode ainda ser dividida em três estratégias diferentes: filtros, *wrappers* e embutidas. A estratégia de filtros é independente de algoritmo, portanto, métodos como o Binary Relevance e o Label Powerset podem ser usados para gerar um ranking de atributos para cada rótulo. A partir do ranking, e de um critério de eliminação, atributos com pontuações insuficientes são eliminados e aqueles com as melhores pontuações são selecionados, filtrando assim os atributos mais significativos para o modelo (GIBAJA; VENTURA, 2014; KASHEF; NEZAMABADI-POUR; NIKPOUR, 2018).

Wrappers são diretamente aplicáveis à classificação multirrótulo e tiram vantagem do algoritmo de aprendizado como parte do processo de seleção. Portanto, dado o algoritmo, a estratégia consiste em procurar por subconjuntos de atributos que otimizem uma função de perda em um conjunto de avaliação. Nas estratégias embutidas a seleção de atributos é realizada como parte do processo de construção do modelo e exploram os pontos fortes das estratégias de filtros (baixa complexidade computacional) e wrappers (apresenta melhores resultados comparado aos filtros) (TSOUMAKAS, 2010; KASHEF; NEZAMABADI-POUR; NIKPOUR, 2018).

A extração de atributos pode ser categorizada em dois principais tipos: Supervisionada e Não Supervisionada. Métodos não supervisionados podem ser diretamente aplicados à classificação multirrótulo, não precisam de dados rotulados e tentam reduzir a dimensionalidade preservando determinadas características dos atributos. *Principal Component Analysis* (PCA)(ABDI; WILLIAMS, 2010) pode ser citado como um método de extração de atributos não supervisionado tradicional. Os métodos supervisionados necessitam de adaptação, precisam dos dados rotulados e analisam correlações entre os atributos e a classe, sendo o *Canonical Correlation InformAtion* (CCA)(YIN, 2004) um método tradicional (HERRERA et al., 2016; KASHEF; NEZAMABADI-POUR; NIKPOUR, 2018).

Nem sempre é possível retirar um rótulo do conjunto, pois as correlações entre os ró-

tulos e outras informações relevantes podem ser perdidas. Neste caso, o espaço de rótulos pode ser dividido em espaços menores, de maneira que as informações e as correlações não se percam no processo. Uma maneira de fazer isto é selecionando grupos de rótulos, processando-os separadamente e então reconstruindo o problema original ao final. Como exemplos podem ser citados os métodos *Pruned Sets*, que ao eliminar conjuntos de rótulos menos frequentes reduz a dimensão do espaço, e HOMER que facilita o processamento ao transformar o problema original em uma hierarquia de problemas menores (GIBAJA; VENTURA, 2014; HERRERA et al., 2016).

Estes métodos de redução tentam simplificar a tarefa multirrótulo, o que leva a outro problema já mencionado em seções anteriores, a escalabilidade. Quanto maior o número de rótulos no espaço de rótulos maior o custo computacional do treinamento, assim como mais memória será necessária para gerar o modelo. Dependendo da dimensionalidade do espaço de rótulos, o algoritmo pode não terminar a execução, mesmo que ele seja dividido em vários espaços menores, pois o número total de instâncias do conjunto também pode influenciar no processamento (SOROWER, 2010).

Outro problema que pode ser gerado com a alta dimensionalidade do espaço de rótulos é o desbalanceamento. Quando o número total de rótulos do espaço de rótulos é muito alto, é possível que o número de instâncias positivas para determinados rótulos seja bem pequeno e o número de instâncias negativas alto, assim como alguns rótulos podem ser mais frequentes que outros. Isto é conhecido como desbalanceamento de classes. Além disso, pode haver um número alto de instâncias associadas a conjuntos de rótulos frequentes e também um alto número de instâncias associadas a conjuntos de rótulos menos frequentes, o que é conhecido como *label skew* (READ, 2010; HERRERA et al., 2016).

(TSOUMAKAS; KATAKIS; VLAHAVAS, 2008) e (WENG et al., 2018) podem ser citados como exemplos de métodos que tratam do desbalanceamento de classes. O trabalho de Tsoumakas, Katakis e Vlahavas (2008) mantém uma distribuição uniforme do conjunto de rótulos em subconjuntos disjuntos, de modo que rótulos semelhantes são agrupados e rótulos dissimilares separados. No trabalho de Weng et al. (2018), duas instâncias positivas (ou negativas) são selecionadas do conjunto de treinamento, e uma nova instância positiva (negativa) é gerada pelo cálculo da média aritmética entre as duas.

Como já mencionado na Seção 2.1, os aspectos aqui abordados podem ser tratados diretamente no desenvolvimento de um novo algoritmo, ou precisam ser adaptados em algoritmos existentes. Conhecer o conjunto de dados, analisando o balanceamento, quantidade de conjuntos de rótulos frequentes e infrequentes, entre outras características, pode auxiliar na resolução do problema multirrótulo. A Seção 2.6 apresenta métricas capazes de extrair estas informações dos conjunto de dados.

2.6 Características de Dados Multirrótulo

Conjuntos de dados multirrótulo possuem diversas propriedades que podem ser mensuradas e as quais caracterizam este tipo de dado. É possível obter desde informações simples como número total de rótulos, número total de instâncias, número total de conjuntos de rótulos distintos, até informações mais complexas como o nível de desbalanceamento do conjunto, esparsidade do espaço de rótulos e de instâncias, entre outros. Estas informações auxiliam na resolução do problema multirrótulo, podendo ajudar na escolha das técnicas a serem utilizadas e também na escolha do conjunto de dados mais adequado pra o problema em questão (READ, 2010; HERRERA et al., 2016).

A média de rótulos por instância é dada pela Equação 27, denominada cardinalidade de rótulos (Card) (TSOUMAKAS; KATAKIS, 2007). Quanto maior o valor de Card, maior é o número de rótulos relevantes por instância. Um valor baixo indicará que a maioria das instâncias possui apenas um rótulo relevante (HERRERA et al., 2016).

$$Card(\mathcal{D}) = \frac{1}{m}\sum_{i=1}^{m} \mid Y_i \mid \tag{27}$$

A Equação 28, denominada densidade de rótulos (Dens), normaliza a cardinalidade pelo número de rótulos (GIBAJA, 2015). Quanto maior o valor de Dens, melhor a representação dos rótulos em cada instância. Quanto menor, mais dispersão há, indicando que a maioria das instâncias é representada por um pequeno sub-conjunto de rótulos (HERRERA et al., 2016).

$$Dens(\mathcal{D}) = \frac{Card(\mathcal{D})}{n} \tag{28}$$

A proporção de combinações de rótulos que são únicas (PUnic) no número total de instâncias é dada pela Equação 33 (READ, 2010).

$$PUnic(\mathcal{D}) = \frac{\mid \{L \mid \exists!\mathbf{x} : (\mathbf{x}, L) \in D\} \mid}{m} \tag{29}$$

A Equação 30 calcula a proporção de ocorrências do conjunto de rótulos com a frequência máxima (PMax), representando dessa forma a proporção de instâncias associadas aos conjuntos de rótulos que ocorrem com mais frequência. $conta(L, \mathcal{D})$ é a frequência com que L aparece combinado com outro rótulo em $\mathcal{D}$ (READ, 2010).

$$PMax(\mathcal{D}) = \max_{L|(x,L)\in\mathcal{D}} \frac{conta(L, \mathcal{D})}{m} \tag{30}$$

A porcentagem de instâncias rotuladas com um único rótulo (PMin) é dada pela Equação 31. Um valor entre 0 e 1 é retornado e um valor de PMin próximo de 1 indica uma alta proporção de instâncias com um único rótulo (HERRERA et al., 2016).

$$PMin(\mathcal{D}) = \sum_{L'\in Y_i/|L'|=1} \frac{\mid L' \mid}{m} \tag{31}$$

Ambas as Equações 27 e 28 fornecem informações sobre frequência de rótulos enquanto que as Equações 29 e 30 fornecem informações de contagem de ocorrências, e neste caso, a contagem das combinação de rótulos únicos e frequência máxima. Um valor alto de PUnic indica que em $\mathcal{D}$ o número de conjuntos de rótulos diferentes é alto. Se o valor de Pmax e PUnic forem ambos altos, então a maioria dos rótulos no conjunto de dados possui apenas alguns exemplos positivos (CHARTE et al., 2018; HERRERA et al., 2016).

A medida apresentada na Equação 32 calcula o *número de conjuntos de rótulos distintos* (Distinto) presentes em $\mathcal{D}$. Caso o valor retornado seja alto, então o espaço de rótulos tem alta dimensionalidade. *Distinto* é limitada pelo número de instâncias do conjunto e quanto mais alto o valor, mais irregular os rótulos aparecem em $\mathcal{D}$. A proporção de conjuntos de rótulos distintos (PD) pode ser calculada pela Equação 33. (SOROWER, 2010; HERRERA et al., 2016).

$$Distinto(\mathcal{D}) =\mid L \subseteq \mathcal{L} \mid \exists(x, L) \in D \mid \tag{32}$$

$$PD(\mathcal{D}) = \frac{Distinto(\mathcal{D})}{m} \tag{33}$$

As medidas apresentadas nas Equações 34, 35 e Equação 36 avaliam o nível de desbalanceamento de um conjunto de dados (CHARTE et al., 2018). O nível de desbalanceamento de um rótulo específico (IRLbl) (L_j) pode ser calculado usando a Equação 34. Um valor de IRLbl igual a 1 indica rótulos com maior frequência, enquanto que um valor maior que 1 indica rótulos menos frequentes. Portanto, quanto maior o valor de IRLbl, mais rara é a presença do rótulo em $\mathcal{D}$.

A proporção do rótulo mais comum em relação ao mais raro (MaxIR) pode ser calculada pela Equação 36, cujo objetivo é obter a razão de desbalanceamento máximo, e para obter a razão de desbalanceamento médio de cada rótulo (MeanIR) aplica-se a Equação 35 (HERRERA et al., 2016; CHARTE et al., 2018).

$$IRLbl(L) = \frac{max_{L_j \in \mathcal{L}}(\sum_{i=1}^{m}[[L_j \in Y_i]])}{\sum_{i=1}^{m}[[L \in Y_i]]} \tag{34}$$

$$MeanIR = \frac{1}{n}\sum_{L \in \mathcal{L}} IRLbl(L) \tag{35}$$

$$MaxIR = \max_{L \in \mathcal{L}}(IRLbl(L_j)) \tag{36}$$

Se o valor de IRLbl for alto para muitos rótulos, ou se o nível de desbalanceamento for extremo para alguns rótulos, então o valor de MeanIR é alto, e o IRLblσ retorna o desvio padrão dos IRLbl(L). Um coeficiente de variação para a taxa de desbalanceamento médio (CVIR) pode ser calculado pela Equação 37. CVIR ajuda a identificar a causa do alto valor de MeanIR (HERRERA et al., 2016).

$$CVIR = \frac{IRLbL\sigma}{MeanIR'} \tag{37}$$

$$IRLbl\sigma(\mathcal{L}) = \sqrt{\frac{1}{n-1}\sum_{L\in\mathcal{L}}(IRLBl(L) - MeanIR)^2} \tag{38}$$

As métricas $Scumble$ (Equação 39) e $Scumble_i$ (Equação 40) avaliam o nível de concorrência entre os rótulos minoritários e majoritários, indicando também se todos os rótulos na instância têm frequências similares ou não (GIBAJA, 2015).

$$Scumble(\mathcal{D}) = \frac{1}{m}\sum_{i=1}^{m} Scumble_i \tag{39}$$

$$Scumble_i = 1 - \frac{1}{\overline{IRLbl_i}}(\prod_{L\in\mathcal{L}} IRLbl_{iL})^{\frac{1}{n}} \tag{40}$$

Uma medida que calcula o produto do número de atributos, número de rótulos e número de combinações diferentes de rótulos é dada pela Equação 41 (CHARTE et al., 2018). O valor retornado por *Theoretical Complexity Score* (TCS) indica a dificuldade em aprender um modelo preditivo do conjunto de dados: quanto mais alto o valor, mais difícil é o aprendizado. Na Equação 41, Ls indica o número total de combinações de rótulos (CHARTE et al., 2016).

$$TCS(\mathcal{D}) = log(m \times n \times Ls) \tag{41}$$

Por fim, o nível de dependência incondicional de rótulos *Unconditional Label Dependency* (ULD) também pode ser calculado. ULD é a média da correlação de rótulos ponderada pelo número de instâncias comuns e é dada pela Equação 42 (LUACES et al., 2012).

$$ULD(\mathcal{L}) = \frac{\sum_{i<j}\rho(L_i, L_j)|L_i \cap L_j}{\sum_{i<j}|L_i \cap L_j|} \tag{42}$$

2.7 Considerações Finais

Este capítulo apresentou a fundamentação teórica deste trabalho que corresponde à classificação multirrótulo. A introdução do capítulo apresentou a definição formal de classificação multirrótulo e também apresentou rapidamente o ranqueamento. Na seção 2.1 foram apresentados os principais métodos para resolver problemas de classificação multirrótulo que são divididos em duas principais abordagens: Independente (Local) e Dependente de algoritmo (Global). Já a seção 2.2 apresentou algumas das mais conhecidas combinações de classificadores multirrótulo da literatura, enquanto que a seção 2.3 apresentou os principais conceitos sobre a modelagem das correlações entre rótulos. As

medidas para avaliar o desempenho preditivo dos classificadores multirrótulo foram apresentadas na seção 2.4 e as características dos dados multirrótulo na seção 2.6. Outros aspectos como a dimensionalidade, escalabilidade e desbalanceamento foram discutidos na seção 2.5. No Capítulo 3 são apresentados os trabalhos correlatos.

Capítulo 3

Trabalhos Correlatos

Este capítulo apresenta trabalhos relacionados à esta pesquisa. Semelhanças e diferenças de cada trabalho com o HPML são apresentadas ao longo do capítulo. A Tabela 13 apresenta as principais características dos estudos revisados. A referência e os domínios de aplicação são apresentados na primeira e segunda colunas. A terceira coluna apresenta as técnicas usadas para modelar as correlações entre rótulos, enquanto que a quarta coluna apresenta o método de particionamento utilizado. Por fim, os classificadores usados para avaliar o trabalho correlato são mostrados na quinta coluna.

Um framework chamado *Group Sensitive Classifier Chains* (GCC) que explora correlações locais foi proposto em (HUANG et al., 2015). Na fase de treinamento, o framework GCC primeiro realiza o agrupamento do conjunto de treinamento inteiro para somente depois aprender as correlações entre os rótulos, capturando portanto as correlações entre rótulos a partir da similaridade entre as instâncias, o que gera grupos de instâncias similares, e não grupos disjuntos de rótulos correlacionados. O método apresentado nesta tese também possui uma versão onde é usada uma estratégia para encontrar grupos disjuntos de rótulos correlacionados a partir da similariade entre as instâncias. No entanto, nenhum rótulo deve repetir nos subconjuntos de rótulos para cada partição híbrida gerada e, portanto, é feito um ajuste nos rótulos dos grupos para que esta restrição seja satisfeita. A Figura 17 ilustra as diferenças entre o método GCC e as partições híbridas.

No GCC, o k-means é aplicado para encontrar os grupos , em seguida, um grafo de dependência de rótulos *Label Dependency Graph* (LDG) é modelado para cada grupo e, por fim, k-classifier chains são construídos com base em cada grafo de dependência de rótulos aprendido. O valor de k para o k-means precisa ser estimado, enquanto que para as partições híbridas um número fixo de grupos dentro de cada partição é indesejado - permite construir e avaliar diversas configurações de partições híbridas, com subconjuntos

Tabela 13 – Sumário dos Trabalhos Correlatos

Referência	**Domínio**	**Correlação**	**Particionamento**	**Classificador**
Huang et al. (2015)	Diversos	*k*-Means	*k*-Means	BSVM, CC, BCC, ML-LOC
Szymański, Kajdanowicz e Kersting (2016)	Diversos	Grafo de Co-Ocorrência de Rótulos	Métodos de Detecção de Comunidades	Ra*k*el
Barezi, Kwok e Rabiee (2017)	Diversos (Extreme Multi-label clasification)	Otimização	Otimização	LEML, WSABIE, CPLST, CS, ML-CSSP, Fast-XML, LPSR
Papanikolaou, Tsoumakas e Katakis (2018)	Diversos	Hierarquia de Rótulos	k-Means Balanceado, FastOptics e Slink	BR-SVM e Label LDA
Nikoloski, Kocev e Džeroski (2018)	Diversos	Hierarquia de Rótulos	k-Means Balanceado, Algoritmo de Agrupamento Hierárquico Aglomerativo, PCTs e GLMM	Clus
Abeyrathna (2018)	Diversos	Co-Ocorrência de Rótulos	C3M	BR e LP
Wang et al. (2019)	Farmacêutico	Cramér's V e GCG	Métodos de Detecção de Comunidades	ERT, RF, SVM, ECG e MLP
Wang, Zhu e Ye (2019)	Farmacêutico	Cramér's V e GCG	Métodos de Detecção de Comunidades	ERT, RF, SVM, ECG e LightGBM
Moyano et al. (2020)	Diversos	Algoritmo Evolutivo	Nenhum	EME, BR, LP, CC, GACC, PS, LPBR, LIFT
Chu et al. (2020)	Farmacêutico	Grafo de Co-Ocorrência de Rótulos com Peso	Métodos de Detecção de Comunidades	LP
Lin, Chen e Lee (2020)	Diversos	k-Modes	GLMM	GLMM
Pliakos, Vens e Tsoumakas (2021)	Farmacêutico	k-Means	k-Means	DTs e ERT
Basgalupp et al. (2021)	Diversos	Nenhum	Bell Number	GA e DT
Chen, Wang e Li (2022)	Diversos	DNN	DNN	DNN

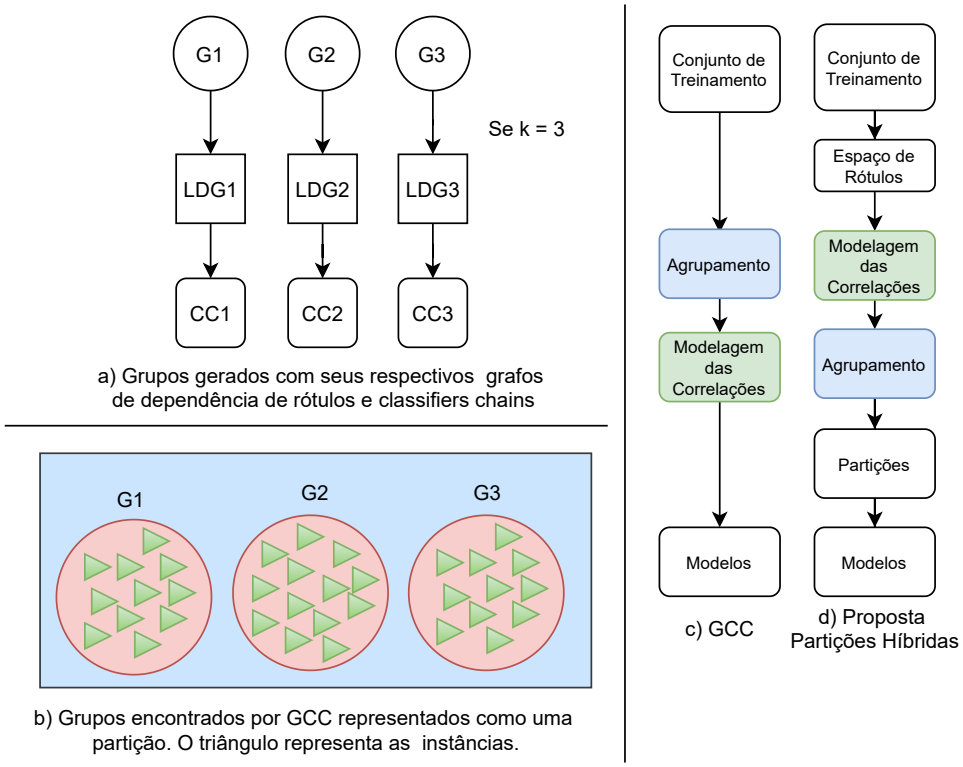


Figura 17 – Comparação entre GCC e as Partições Híbridas. Fonte: Elaborado pela autora com base em Huang et al. (2015).

de rótulos diversos, e assim levar a escolha da partição híbrida mais adequada. Outra diferença entre o GCC e as partições híbridas, é que o primeiro modela as correlações de forma local enquanto que o método para encontrar as partições híbridas desta tese, modela as correlações de froma global.

Na fase de teste, o framework GCC encontra o grupo g_n mais próximo para a instância de teste $\mathbf{x}_t$. Os autores assumiram que $\mathbf{x}_t$ compartilha as mesmas correlações de rótulos que as instâncias que pertencem a g_n. Assim, o modelo de classifier chains construído para g_n é usado para testar $\mathbf{x}_t$. Para as partições híbridas aqui apresentadas, a fase de teste utiliza a partição híbrida vencedora, entre todas as geradas, para treinar e testar o modelo final.

Outro método similar às partições híbridas é apresentado em (MOYANO et al., 2020). O método denominado *Evolutionary AlGorithm for Multi-Label Ensemble OpTimization* (EAGLET) é baseado em algoritmo evolutivo e tem como objetivo gerar uma combinação de classificadores multirrótulo onde cada um dos seus membros é um classificador multirrótulo base focado em um subconjunto de rótulos. Na criação da população inicial tanto rótulos frequentes quanto infrequentes são considerados, sendo que cada rótulo é forçado a aparecer um número mínimo de vezes. Comparando com as partições híbridas, cada indivíduo da população poderia ser considerado como um grupo de uma partição, pois para cada um desses grupos e indivíduos é necessário um classificador. Cada indi-

víduo do método EAGLET é decomposto em um novo conjunto de dados que consiste apenas dos k rótulos ativos naquele indivíduo.

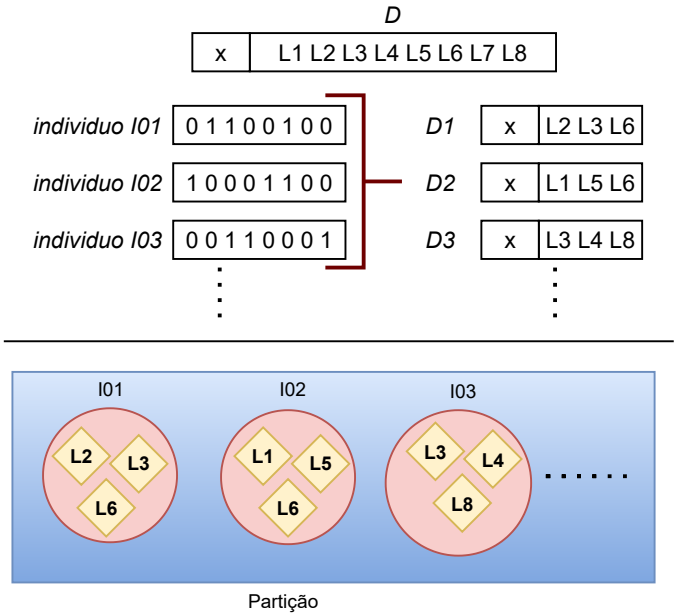


Figura 18 – Indivíduos encontrados pelo método EAGLET representados como uma partição. Fonte: Elaborado pela autora baseado em Moyano et al. (2020).

O número de indivíduos, o número de gerações, o número de classificadores e o número de rótulos de cada classificador devem ser informados pelo usuário para o método EAGLET. Novamente, prefere-se que para encontrar as partições híbridas, o número de rótulos para cada subconjunto da partição não seja determinado. Como consequência disso, o número de classificadores variará conforme o número de grupos disjuntos de rótulos encontrados para cada partição híbrida. Os indivíduos encontrados pelo método EAGLET são representados como uma partição na Figura 18.

No método EAGLET, novos indivíduos herdam rótulos dos pais nas fases de crossover e mutação, e dessa forma as correlações locais entre os rótulos são consideradas. A medida de avaliação F1 foi usada como função de *fitness* para avaliar e escolher o melhor indivíduo. De maneira similar, para as partições híbridas, critérios como a medida de avaliação Macro-F1 e coeficiente da silhueta são usados para selecionar a melhor partição híbrida entre todas as geradas. Enquanto no método EAGLET, o classificador que obtiver o melhor desempenho preditivo (F1) e diversidade é selecionado, nas partições híbridas a partição híbrida com o melhor desempenho preditivo - ou melhor coeficiente da silhueta - é selecionada para o teste final.

O algoritmo proposto em Papanikolaou, Tsoumakas e Katakis (2018) considera a similaridade entre os rótulos e os particiona usando um algoritmo de agrupamento, é um método com características comuns às partições híbridas (modelar correlaçãoes e particioná-las). No entanto, a versão de Papanikolaou, Tsoumakas e Katakis (2018) é um

HOMER aprimorado (TSOUMAKAS; KATAKIS; VLAHAVAS, 2008), que permite que a geração de uma hierarquia baseada na similaridade entre os rótulos e os nós folhas são compostos por mais de um rótulo. A Figura 19 apresenta o HOMER.

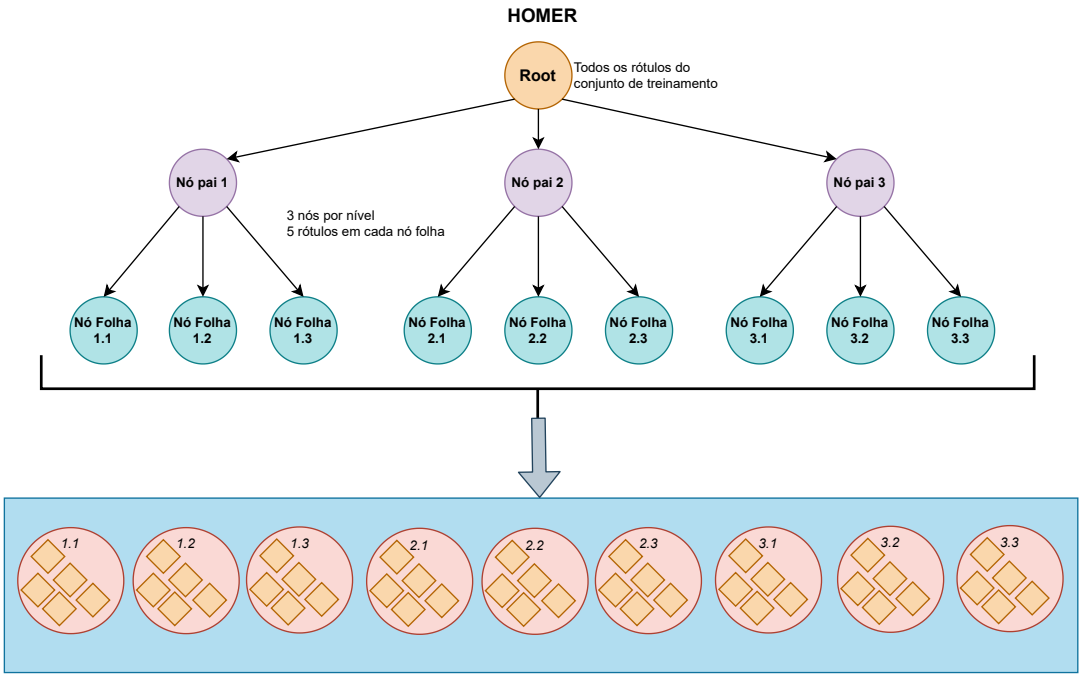


Figura 19 – HOMER representado como uma partição. Fonte: Elaborado pela autora com base em Papanikolaou, Tsoumakas e Katakis (2018).

O algoritmo *Balanced k Means* foi usado para o agrupamento e o mesmo exige que o usuário determine os parâmetros k (número de grupos) e também n_{max} (número de rótulos em cada nó folha), o que já é uma diferença com relação às partições híbridas. O método HOMER gera uma partição similar à partição híbrida, pois cada nó folha contem sub-conjuntos de rótulos similares, no entanto, essa partição não pode ser considerada partição híbrida pois a quantidade de nós e rótulos dentro de cada nó é limitada pelo usuário. Além disso, uma partição híbrida é composta por grupos disjuntos de rótulos correlacionados, onde cada grupo pode conter um único rótulo, ou um par de rótulos correlatos, ou um subconjunto de rótulos correlacionados, o que não é o caso do HOMER.

Outro método encontrado na literatura que é bastante similar às premissas das partições híbridas é o *Multi-Label Classification Label Clusters* (MLC-LC), proposto por Abeyrathna (2018) e ilustrado na Figura 20. Este método usa uma versão modificada do algoritmo C^3M (CAN; OZKARAHAN, 1990) para modelar as correlações entre os rótulos e particionar o espaço de rótulos, já que o mesmo calcula o número ideal de grupos dentro da partição. Portanto, ambos os trabalhos modelam correlações de alta ordem, isto é, modelam as correlações entre os rótulos usando todo o espaço de rótulos de uma única vez e um número de grupos de rótulos não é fornecido pelo usuário.

Pode-se dizer que MLC-LC segue os mesmos passos que são necessários para encontrar

as partições híbridas, no entanto, existem diferenças. A principal diferença entre os dois trabalhos está na geração das partições híbridas: MLC-LC não gera diversas partições híbridas. Outra diferença está na forma de treinar os modelos. MLC-LC usa o classificador *Label PowerSet*[1] para treinar grupos com mais de um rótulo, significando que para cada grupo desses serão gerados novos rótulos de acordo com as combinações de rótulos existentes.

Para as partições híbridas, se o método de desempenho for escolhido como critério de validação das partições híbridas, então, para cada grupo de rótulos correlacionados formado por mais de um rótulo, um classificador multirrótulo é aplicado de modo que se aprenda as correlações que foram identificadas naquele grupo em questão de uma única vez. Mesmo que o Label PowerSet aprenda as correlações encontradas, o método o faz de uma maneira mais custosa, o que não é interessante.

A Figura 20a apresenta o processo do MLC-LC que, ao usar o M^3C é capaz de gerar agrupamentos de rótulos correlacionados para uma partição, enquanto que a Figura 20b apresenta o que foi determinado para o as partições híbridas: a geração de várias partições que são compostas por grupos disjuntos de rótulos correlacionados.

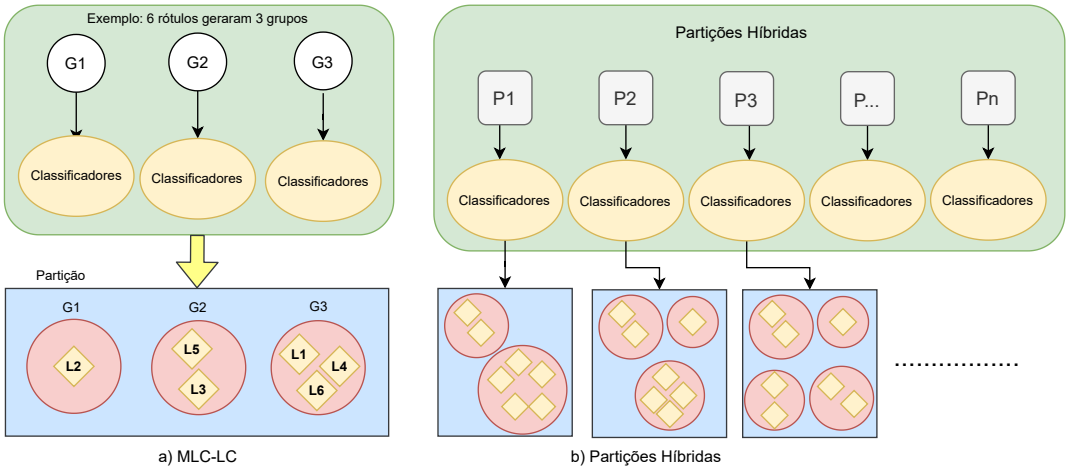


Figura 20 – Comparando MLC-LC com as Partições híbridas. Fonte: Elaborado pela autora baseado em Abeyrathna (2018).

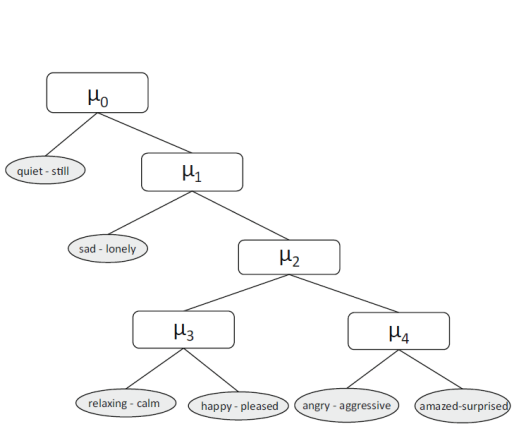
Assim como o HOMER, o trabalho apresentado em Nikoloski, Kocev e Džeroski (2018) também tem como objetivo gerar hierarquia de rótulos considerando as correlações existentes entre eles, o que é diferente das partições híbridas que são compostas por grupos disjuntos de rótulos e não por hierarquia de rótulos. Diferente dos outros trabalhos já apresentados, e também das partições híbridas, que são tarefas de classificação multirrótulo, o trabalho de Nikoloski, Kocev e Džeroski (2018) aborda a classificação multirrótulo como uma tarefa de classificação hierárquica multi-rótulo.

De acordo com os autores, as hierarquias de rótulos permitem modelar a estrutura de dependência e os rótulos interdependentes, sendo construídas usando um ranking dos

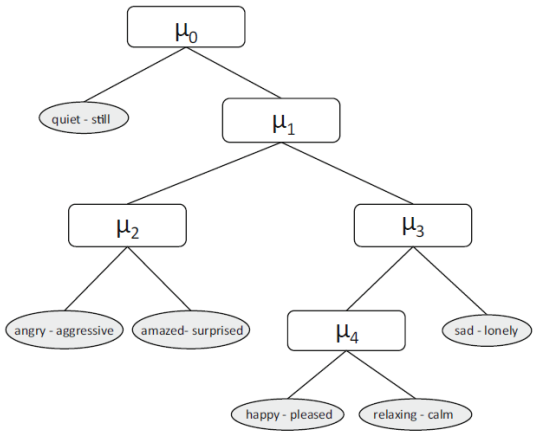
[1] *Label PowerSet* foi discutido na subseção 2.1.1 do Capítulo 2

atributos e não o espaço de rótulos. Portanto, primeiro os atributos são ranqueados usando *Random Forests* - Florestas Aleatórias (RF), depois a hierarquia é construída e, por fim, os modelos são induzidos e testados. A hierarquia encontrada é usada para transformar o conjunto de dados multirrótulo plano em conjuntos de dados multirrótulo hierárquicos de treino e teste. Um algoritmo de agrupamento hierárquico aglomerativo com ligação simples e com ligação completa, o k-means balanceado, e PCTs foram usados para gerar as hierarquias e o CLUS foi usado no treino e teste.

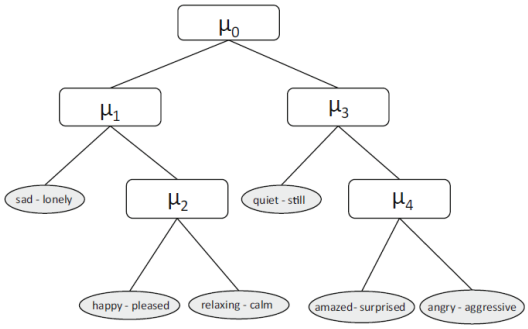
A Figura 21 foi retirada do artigo original e apresenta exemplos de hierarquias geradas para o conjunto de dados *emotions*. O símbolo μ indica o nó da hierarquia, assim como o grupo a que pertence o rótulo. Os autores também investigaram se é possível construir hierarquias de rótulos usando o ranqueamento do espaço de rótulos e qual dos métodos aplicados produz a melhor hierarquia de acordo com o ranking. Criar e avaliar ranqueamentos ou hierarquias a partir de ranqueamentos não é objetivo das partições híbridas.



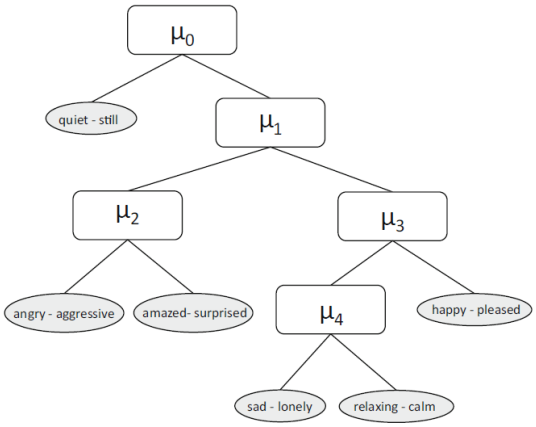
(a) Ligação Simples



(b) Ligação Completa



(c) k-means balanceado



(d) *Predictive Clustering Trees*

Figura 21 – Hierarquias geradas pelos quatro algoritmos. Fonte: (NIKOLOSKI; KOCEV; DžEROSKI, 2018). Retirado do artigo.

Outro trabalho que se assemelha às partições híbridas, denominado *Network-based*

Label Space Partition (NLSP), é apresentado em Szymański, Kajdanowicz e Kersting (2016). NLSP propõe uma abordagem alternativa ao particionamento aleatório do espaço de rótulos, a qual é orientada a dados[2].

A Figura 22 ilustra as diferenças entre NLSP e as partições híbridas. Como pode ser observado na Figura 22, os autores primeiro utilizam o Ra*k*el para gerar 250 partições aleatórias dos dados e, a partir destas partições, modelam os grafos de co-ocorrência, aplicam os métodos de detecção de comunidades, e então executam a classificação. No caso das partições híbridas, elas são obtidas a partir do espaço de rótulos original e não por vários espaços aleatórios diferentes, no entanto a metodologia é similar:

Outra diferença está no objetivo dos dois trabalhos. O NLSP avalia como o particionamento do espaço de rótulos, usando abordagens orientadas a dados, pode melhorar o particionamento aleatório na classificação multirrótulo. Isto é diferente de encontrar partições híbridas no espaço de rótulos para melhorar o desempenho de qualquer classificador multirrótulo que venha a ser utilizado.

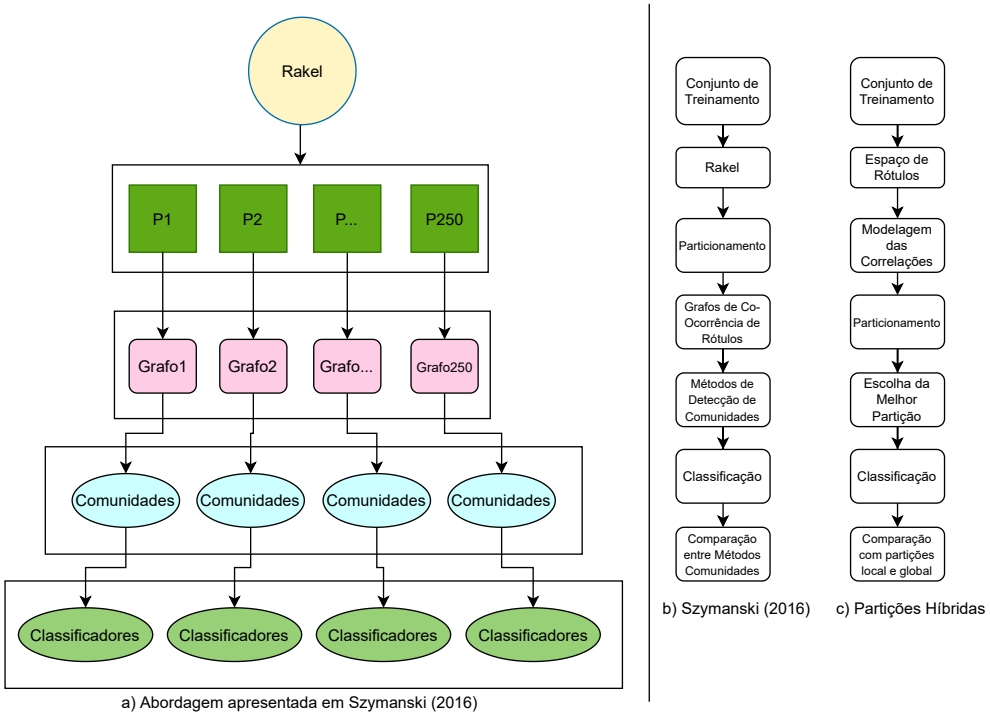


Figura 22 – Comparando o NLSP (esquerda) com as partições híbridas (direita). Elaborado pela autora com base em Szymański, Kajdanowicz e Kersting (2016).

Nos trabalhos de Wang et al. (2019) e Wang, Zhu e Ye (2019) os autores extenderam a ideia principal do NLSP para resolver problemas mais específicos na área de farmacêutica: o *Anatomical Therapeutic Chemical* (ATC) e o *Network-Based Label Space Partition method for predicting the Specificity of Membrane Transporter Substrates* (STS-NLSP). O primeiro empregou o NLSP para prever classes ATC de uma determinada substância,

[2] Considera os dados ao invés da aleatoriedade ao gerar grupos. Por exemplo, ao invés de selecionar subespaços aleatórios dos dados, seleciona subespaços com base na co-ocorrência dos dados

sugerindo uma técnica para categorizar produtos farmacêuticos multirrótulo com base em propriedades terapêuticas, farmacêuticas e químicas. O método ATC-NLSP modela as correlações de rótulos usando a estatística de Cramners'V para criar um grafo de co-ocorrência de rótulos e, em seguida, particionar o espaço de rótulos usando métodos de detecção de comunidade. Apesar de o ATC-NLSP ter obtido bons resultados, os autores notaram que o número de comunidades detectadas pelo NLSP foi muito baixo.

O segundo utilizou NLSP para predizer a especificidade de um substrato de transportador de membrana. Também usaram um recurso híbrido de similaridade estrutural e semântica, baseado na estratégia estabelecida no ATC-NLSP, para encontrar relações significativas entre os compostos. O STS-NLSP visa prever se um substrato pode reconhecer especificamente uma das treze categorias de transportadores de drogas, desde o cassete de ligação de ATP até as famílias de transportadores de soluto. O STS-NLSP pontuou mal para alguns conjuntos de dados, de acordo com os autores, devido ao desbalanceamento de classes.

Ainda na área de farmacêutica, Chu et al. (2020) desenvolveu uma abordagem chamada *Multi-Label Learning with Community Detection Method for Identifying Drug-Target Interactions Prediction* (DTI-MLCD), que está relacionada com NLSP, ATC-NLSP e STS-NLSP. O objetivo do DTI-MLCD é prever novos alvos para medicamentos existentes, bem como novas terapias para atributos-alvos previamente identificados. Usando grafos de co-ocorrência de rótulos ponderados e métodos de detecção de comunidades, o DTI-MLCD divide o espaço de rótulos em vários subespaços, aplicando classificadores multirótulos em cada um e combinando as predições finais. Os autores compararam os resultados do algoritmo k-means com os da abordagem de detecção de comunidades, concluindo que os resultados desta última foram superiores em termos de desempenho e interpretabilidade. O DTI-MLCD, assim como o STS-NLSP, pode ser influenciado pelo desbalanceamento dos rótulos.

A estratégia adotada pelos métodos NLSP, ATC-NLSP, STS-NLSP e DTI-MLCD são bastante semelhantes às partições híbridas. As etapas propostas podem ser resumidas da seguinte forma: 1) modelagem das correlações de rótulos; 2) particionamento do espaço de rótulo de acordo com 1; 3) construção dos respectivos conjuntos de dados para cada partição, e 4) treinamento e teste dos modelos. A principal diferença entre esses trabalhos e as partições híbridas é o objetivo para o qual foram projetados. Esta tese reporta um estudo onde o objetivo é encontrar, construir e avaliar várias partições com grupos disjuntos de rótulos correlacionados para uma variedade de dados multi-rótulo. NLSP, ATC-NLSP, STS-NLSP e DTI-MLCD usam o particionamento de rótulos para melhorar as predições de um domínio específico, ou apenas para analisar as correlações entre rótulos, o qual também é baseado na aleatoriedade para melhorar a abordagem orientada a dados. A Figura 22 pode ser usada como referência comparativa entre estes trabalhos e as partições híbridas.

Continuando na área de bioinformática, outro trabalho usou metodologia semelhante à das partições híbridas. Pliakos, Vens e Tsoumakas (2021) apresentou a abordagem *Multi-Label Classification with Label-Feature Encoding* (MLCLE), que usa correlações entre rótulos e particionamento de rótulos para organizar proteínas-alvo (espaço de saída) em grupos biologicamente significativos. O objetivo no entanto é bem claro: desenvolver um novo framework de classificação multirrótulo para interações Drug-Target. O MLCLE primeiro divide as proteínas-alvo em k grupos e, em seguida, constrói características codificando os resultados do agrupamento. A Figura 23 foi retirada do artigo original.

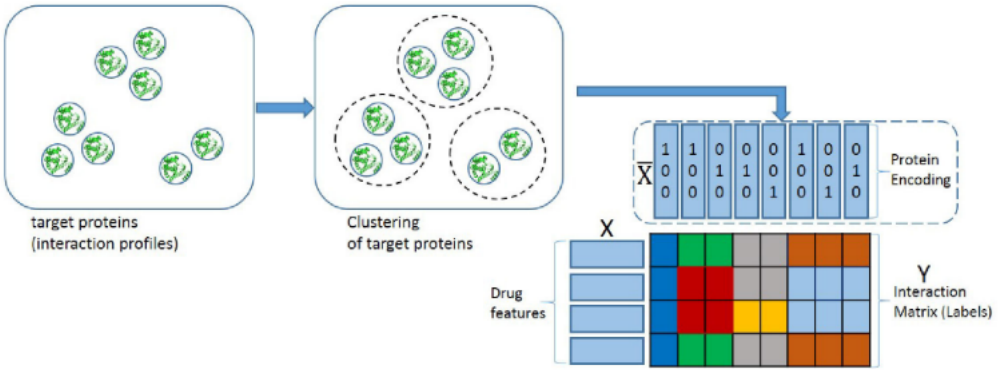


Figura 23 – MLCLE. Fonte: Pliakos, Vens e Tsoumakas (2021). Retirado do artigo. A Figura pode ser melhor visualizada no artigo original.

O estágio final é criar um modelo baseado em árvore biclustering (não foi usado para as partições híbridas). Este método tira proveito das correlações entre rótulos e cria **uma única partição**, e não muitas. Os autores também se preocuparam com a interpretabilidade, uma vez que um agrupamento de atributos-alvos pode definir uma família específica de proteínas ou até mesmo cumprir a mesma função. Como resultado, eles usaram *Decision Trees* - Árvores de Decisão (DTs), enfatizando que elas são simples de compreender e permitem a aquisição de novos conhecimentos e insights importantes. Outra descoberta do artigo foi que utilizar o particionamento de rótulos com correlações de rótulos pode ajudar a melhorar o desempenho dos classificadores.

Internet das Coisas é outra área em que uma metodologia similar às partições híbridas foi utilizada. Lin, Chen e Lee (2020) propuseram uma meta-aprendizagem híbrida baseada em rótulos com o Modelo Misto Linear Generalizada (MMLG) para auxiliar no reconhecimento de atividades de sensores de IoT. HybridL-BGLM, nome do método, usa o algoritmo k-Modes para particionar o espaço de rótulos e o MMLG para modelar as correlações, funcionando em quatro passos: i) agrupar os rótulos para reduzir o número de rótulos e detectar a correlação entre eles; ii) construir subclassificadores entre os grupos (between-group) via MMLG; iii) construir sub-classificadores dentro de cada (within-group) grupo via MMLG; iv) construir a combinação de classificadores final, isto é, combinar as predições de ii) e iii).

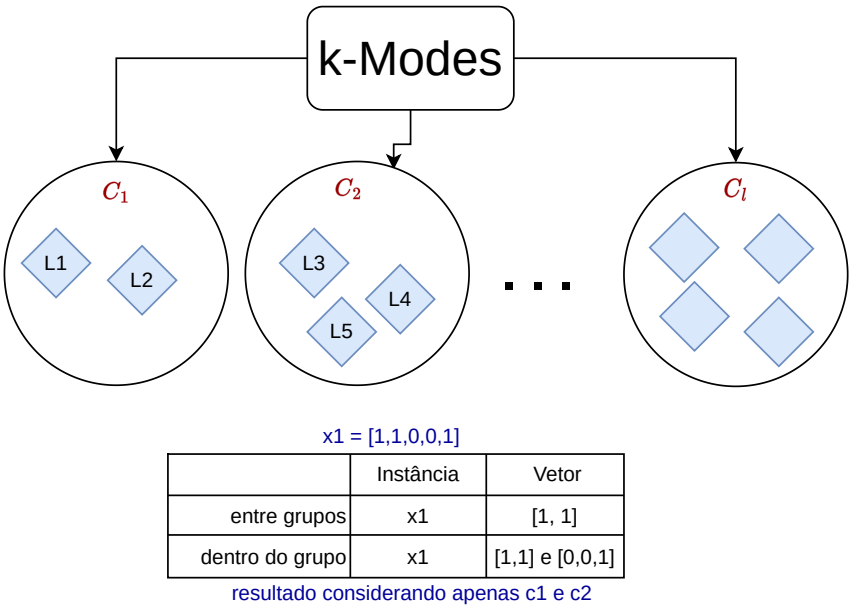


Figura 24 – HybridLBGLM. Elaborado pela autora com base em Lin, Chen e Lee (2020).

O HybridL-BGLM calcula um vetor de rótulos entre grupos e também um vetor de rótulos dentro de cada grupo para cada uma das instâncias do conjunto de dados. Dessa forma, é verificado para cada grupo de rótulos gerado pelo k-Modes: i) uma instância tem um vetor de dados entre grupos (between-group) se ela pertence a qualquer um dos rótulos em um grupo; ii) uma instância tem um vetor de dados dentro do grupo (within-group) se ela pertence a um rótulo específico dentro do grupo. Pelos passos acima mencionados, é possível notar que o HybridLBGLM particiona o espaço de rótulos, mas não produz partições híbridas da maneira que foi definida nesta tese. A Figura 24 ilustra o método HybridL-BGLM. A metodologia para as partições híbridas (verificar Figura 44 e Figura 17) possui mais passos, um deles, a escolha da melhor partição híbrida, algo que não ocorre na metodologia do HybridLBGLM.

Outro trabalho correlacionado é da área de área de Extreme Multi-Label Classification, que trabalha com conjuntos de dados multirrótulo com um alto número de rótulos. O método apresentado por Barezi, Kwok e Rabiee (2017) divide o espaço de rótulos em subgrupos de rótulos independentes com o objetivo de paralelizar o aprendizado. Um processo de otimização é usado para encontrar subgrupos independentes de rótulos com densas intradependências e também esparsas dependências entre diferentes grupos.

O HPML encontra mais de uma partição (híbrida) do espaço de rótulos, onde cada partição é composta por mais de um subgrupo de rótulos. Além disso, o HPML não se concentra apenas nos conjuntos de dados multirrótulo com um número muito grande de rótulos (extreme multilabel classification).

Por fim, Basgalupp et al. (2021) apresentou uma ideia semelhante às partições híbridas. Os autores investigaram se havia uma partição de rótulos capaz de melhorar o desempenho

do classificador em comparação ao uso das abordagem global e local. Cada subconjunto da partição de rótulos é tratado como um problema de predição. Os autores fizeram um teste minucioso e abrangente para descobrir a melhor divisão utilizando um método baseado em algoritmo genético para encontrar as partições. Nesta tese não foi utilizada nenhum método ou algoritmo genético. Outra diferença que pode ser apontada é que em Basgalupp et al. (2021) os autores se concentraram em tarefas de regressão e classificação, enquanto que aqui a preocupação é apenas com a classificação multi-rótulo. Finalmente, uma diferença significativa é que os autores não consideraram as correlações entre rótulos na investigação, enquanto que aqui o objetivo é modelar as correlações e particioná-las.

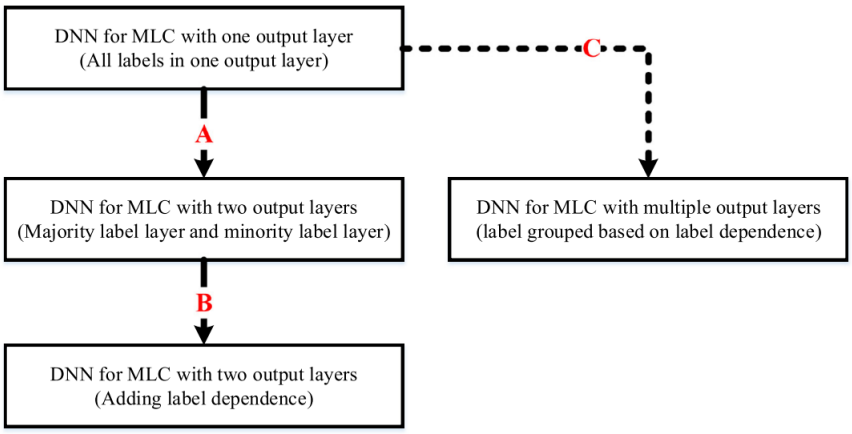


Figura 25 – Esquema do método apresentado por Chen, Wang e Li (2022). Retirado do artigo.

Chen, Wang e Li (2022) conclui esta seção de trabalhos correlatos, onde configurações especiais de redes neurais foram desenvolvidas para melhorar o desempenho da classificação multirrótulo baseado em Redes Neurais Profundas (*Deep neural networks* (DNN)) e considerando tanto o desbalanceamento dos dados quanto as correlações entre rótulos. Os autores desenvolveram três estratégias para tratar das correlações, conforme mostra a Figura 25. Na Etapa A os rótulos são separados em rótulos majoritários e rótulos minoritários, enquanto na Etapa B dependências de rótulos são adicionadas alterando os rótulos nos grupos de rótulos resultantes da Etapa A. Por fim, a Etapa C separa os rótulos diretamente com base na dependência de rótulo. Apesar de tanto o método apresentado por (CHEN; WANG; LI, 2022) quanto o método aqui proposto buscarem melhorar o desempenho preditivo considerando as correlações, o trabalho de (CHEN; WANG; LI, 2022) também trata do desbalanceamento de classes, o que não é tratado aqui. Além disso, o objetivo do método aqui proposto não é otimizar parâmetros de redes neurais como ocorre em (CHEN; WANG; LI, 2022), mas sim encontrar partições que possam melhor otimizar o desempenho.

Concluindo, este capítulo apresentou alguns trabalhos correlatos às partições híbridas, mostrando que pesquisadores têm investido nos temas do particionamento do espaço de

rótulos e também modelagem das correlações. Apesar dos princípios serem similares em cada trabalho correlato, esta tese vai além disso. Todos os trabalhos apresentados têm algo em comum com o HPML, mas nenhum explorou as correlações da forma que se propõe nesta pesquisa: uma abordagem em sete passos que gera várias partições de rótulos e escolhe uma entre elas que é considerada a mais adequada para aumentar o poder preditivo do classificador. Portanto, como resultado, nenhum desses trabalhos foi capaz de produzir as (várias) partições híbridas conforme se propõe nesta tese. Esta é a principal distinção entre os trabalhos apresentados e o HPML.

Capítulo 4

Partições Híbridas para Classificação Multirrótulo

Este capítulo apresenta a abordagem proposta assim como todas as suas variações, que é denominado aqui como HPML. O principal objetivo é encontrar partições de grupos disjuntos de rótulos correlacionados localizados entre as tradicionais partições global e local. Para isto, diferentes técnicas de particionamentos baseadas nas correlações entre rótulos modeladas são usadas, as quais geram diferentes partições híbridas, cada uma composta por diferentes números de grupos, e diferentes números de rótulos correlacionados dentro de cada grupo. A Figura 26 apresenta a visão geral da abordagem desenvolvida, a qual é dividida em sete passos principais:

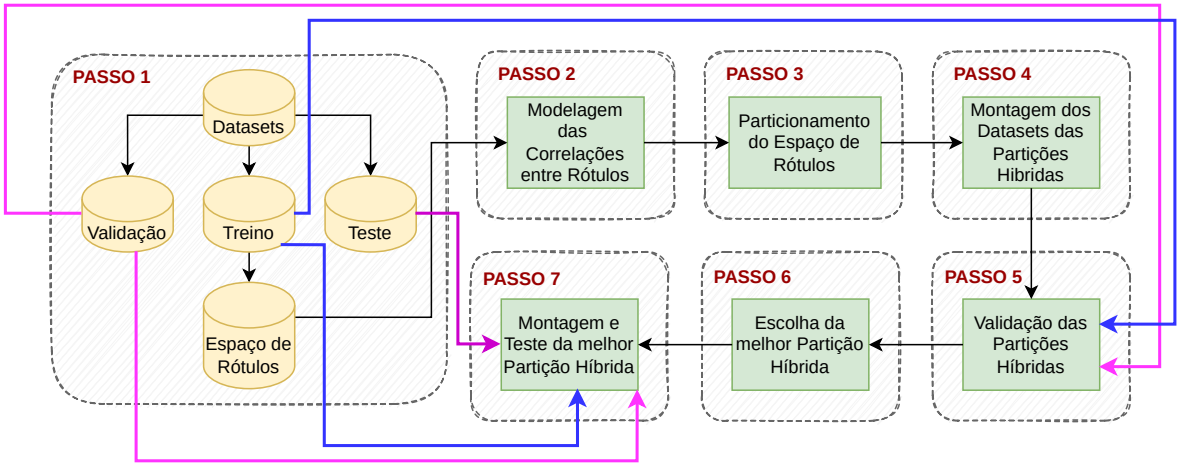


Figura 26 – Visão geral da abordagem HPML. Elaborado pela autora.

- **Passo-1:** *Pré-processamento.* Primeiro, o conjunto de dados multirrótulo é dividido em conjuntos de validação, treinamento e teste;

- **Passo-2:** *Modelagem das correlações entre rótulos.* As correlações entre rótulos são modeladas usando apenas o espaço de rótulos do conjunto de treinamento. Três diferentes técnicas são usadas para modelar as correlações: medidas de similaridades, mapas auto organizáveis (redes neurais) e grafos de co-ocorrência de rótulos (métodos de detecção de comunidades). Essas técnicas não são usadas em conjunto, mas sim separadamente;

- **Passo-3:** *Particionamento do espaço do rótulos.* O particionamento do espaço de rótulos é feito com base nas correlações modeladas e corresponde a encontrar as partições híbridas. Três diferentes técnicas são usadas neste passo: algoritmo de agrupamento aglomerativo hierárquico, mapas auto organizáveis e métodos de detecção de comunidades. Neste caso, é importante ressaltar que: i) se as correlações forem modeladas com os mapas auto-organizáveis de Kohonen, o particionamento também deve ser feito através dele, ii) se as correlações forem modelas com medidas de similaridade, pode-se então particionar usando o algoritmo de agrupamento hierárquico aglomerativo, ou então modelar grafos de co-ocorrência de rótulos e em seguida particioná-los usando os métodos de detecção de comunidade; iii) não há como aplicar agrupamento hierárquico em redes neurais, nem usar redes neurais com grafos.

- **Passo-4:** *Montagem dos datasets das partições híbridas.* Para cada grupo de cada partição híbrida encontrada é construído o dataset correspondente. Todas as instâncias do dataset são usadas nos grupos e apenas os rótulos são selecionados de acordo com o grupo ao qual pertencem. Isto é feito para os conjuntos de treinamento, teste e validação;

- **Passo-5:** *Validação das partições híbridas.* Duas diferentes abordagens podem ser usadas para validar todas as partições híbridas encontradas no Passo-3 e construídas no Passo-4: 1) modelos de classificação são induzidos usando o conjunto de treinamento e o desempenho é avaliado com o conjunto de validação; 2) o coeficiente da silhueta é calculado usando o espaço de rótulos do conjunto de treinamento. Essas abordagens podem ser usadas em conjunto ou separadamente, de qualquer forma, essas partições selecionadas são testadas e comparadas entre elas e entre as partições local, global e aleatória;

- **Passo-6:** *Escolha da melhor partição híbrida.* Para escolher a melhor partição híbrida dois critérios são usados: 1) a Macro-F1 que é uma medida de avaliação de um classificador multirrótulo, e 2) o coeficiente da silhueta que é uma medida de

qualidade de agrupamento. Para ambos os critérios, a partição escolhida como a melhor é aquela que obtém o maior valor entre todas as partições validadas;

- **Passo-7:** *Montagem e teste da melhor partição híbrida.* Finalmente, usando o conjunto de teste, classificadores são induzidos nos grupos da melhor partição híbrida escolhida e o desempenho final do modelo é obtido.

Três principais versões foram desenvolvidas para HPML e estão organizadas na Tabela 14. As principais diferenças entre essas versões estão nas técnicas usadas para modelar as correlações e particionar o espaço de rótulos. Além disso, três versões estendidas do HPML.A, baseadas no conceito de conjunto de cadeia de classificadores também foram elaboradas. Essa versão é denominada HPML.D e suas características estão sumarizadas na Tabela 15. As próximas seções detalham cada uma dessas variações.

Tabela 14 – Variações da abordagem HPML

	HPML.A	**HPML.B**	**HPML.C**
Passo 2	*Medidas de Similaridade* - Jaccard Index - Rogers Tanimoto	*Redes Neurais* - SOM/Kohonen	*Grafo de Co-Ocorrência de Rótulos* - Jaccard Index - Rogers Tanimoto
Passos 3 e 4	*Algoritmo de Agrupamento Hierárquico Aglomerativo:* - Coeficiente Aglomerativo mais alto - Corte dos dendrogramas	- Corte do Mapa de Neurônios - Transformar Partições Kohonen em Partições Híbridas	*Métodos de Detecção de Comunidades:* - Modularidade mais alta - Corte dos dendrogramas
Passos 5 e 6	- Macro-F1 - Coeficiente da Silueta	- Macro-F1 - Coeficiente da Silueta	- Macro-F1 - Coeficiente da Silueta
Passo 7	Classificadores Binários e Multirrótulo	Classificadores Binários e Multirrótulo	Classificadores Binários e Multirrótulo

4.1 HPML versão A

A versão HPML.A modela as correlações usando medidas de similaridade e particiona o espaço de rótulos usando um algoritmo de agrupamento hierárquico aglomerativo. Para validar as partições híbridas o método usa tanto um classificador, quanto calcula o coeficiente da silhueta. A seguir detalhes de como o método HPML.A funciona serão explicados.

Tabela 15 – Variações da abordagem HPML com Cadeias de Classificadores - HPML.D

	Encadeamento Interno	**Encadeamento Externo**	**Encademaneto Interno e Externo**
Passo 2	*Medida de Similaridade* Jaccard Index	*Medida de Similaridade* Jaccard Index	*Medida de Similaridade* Jaccard Index
Passos 3 e 4	*Algoritmo de Agrupamento Hierárquico Aglomerativo:* - Ward.D2 - Corte do dendrograma	*Algoritmo de Agrupamento Hierárquico Aglomerativo:* - Ward.D2 - Corte do dendrograma	*Algoritmo de Agrupamento Hierárquico Aglomerativo:* - Ward.D2 - Corte do dendrograma
Passos 5 e 6	Coeficiente da Silhueta	Coeficiente da Silhueta	Coeficiente da Silhueta
Passo 7	Um ECC é induzido em cada grupo	Os grupos são encadeados e um classificador multirrótulo é induzido em cada grupo	Os grupos são encadeados e em cada um deles um ECC é induzido

4.1.1 Modelagem das Correlações

Medidas de similaridade visam quantificar até que ponto as instâncias ou rótulos de um conjunto de dados se assemelham. As medidas de similaridade podem ser aplicadas a dados binários (representados pela presença ou ausência de instâncias/atributos/rótulos), dados numéricos (representados por vetores de números reais) e dados estruturados (representados por árvores e grafos). Portanto, é possível utilizar alguma medida de similaridade existente, no espaço de rótulos, para quantificar a semelhança entre esses rótulos (LESOT; RIFQI; BENHADDA, 2009).

As medidas de similaridade, portanto, são capazes de medir o grau de similaridade entre pares de rótulos (JUNIOR, 2019) dentro de um conjunto de treinamento. Para demonstrar como elas podem ser aplicadas no método proposto considere o espaço de rótulos do conjunto de dados apresentado no Capítulo 2 e representado nesta subseção pela Tabela 16. Cada linha da Tabela 16 representa um conjunto de rótulos Y_i associado à i-ésima instância $\mathbf{x}_i$ do conjunto de dados multirrótulo. Assim, o espaço de rótulos de $\mathcal{D}_e$ é representado por uma matriz binária $\mathcal{M}$, onde cada uma de suas células ($m_{i,j}$) recebe o valor 1 se a instância $\mathbf{x}_i$ é classificada na classe y_j, e 0 caso contrário.

Tabela 16 – HPML.A: Espaço de rótulos de $\mathcal{D}_e$

Label1	**Label2**	**Label3**	**Label4**	**Label5**
1	0	1	1	0
1	1	1	0	0
0	1	0	1	0
1	0	0	0	1
1	0	1	1	0

Dado um espaço de rótulos estruturado como o da Tabela 16, uma matriz de contingência pode ser construída com o número de ocorrências e co-ocorrências associadas a

cada par de rótulos (p_i, p_j) no conjunto de dados. A tabela de contingência apresentada na Figura 27 para um par de rótulos (p_i, p_j) é construída da seguinte forma:

- ❏ a: corresponde ao número total de ocorrências simultâneas de p_i e p_j;
- ❏ b: corresponde ao número total de ocorrências de p_i sem p_j;
- ❏ c: corresponde ao número total de ocorrências de p_j sem p_i;
- ❏ d: corresponde ao número total de ocorrências sem p_i e sem p_j.

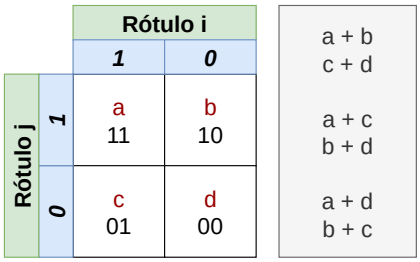


Figura 27 – HPML.A: Tabela de Contingência

A Figura 27 mostra ainda as somas de diagonais, linhas e colunas. Uma vez que a tabela de contingência é preenchida, é possível calcular a similaridade dos pares de rótulos (p_i, q_j), sendo o processo executado para todos os rótulos do espaço de rótulos de $\mathcal{D}_e$. Neste trabalho duas medidas de similaridades reconhecidas e relevantes na literatura foram utilizadas: Índice de Jaccard (Equação 43) e Rogers Tanimoto (Equação 44). A principal diferença entre Rogers e Jaccard está no uso do atributo **d** da matriz de contingência no cálculo: a segunda considera, a primeira não.

$$Jaccard = \frac{a}{a+b+c} \tag{43}$$

$$Rogers = \frac{(a+d)}{a+(2*(b+c))+d))} \tag{44}$$

Depois de calcular todas as semelhanças de pares de rótulos usando as medidas, uma matriz de similaridade é obtida, como a apresentada na Tabela 17. Um par de rótulos onde $p_i = p_j = 1$ indica máxima similaridade e, $p_i = p_j = 0$ indica nenhuma similaridade. A Figura 28 ilustra os passos 1 e 2 para o HPML.A.

4.1.2 Particionando o espaço de rótulos e gerando os datasets das partições híbridas

Dada uma matriz com todas as similaridades dos pares, todos os valores de similaridades devem ser convertidos em valores de dissimilaridade para que seja possível usar a

Tabela 17 – HPML.A: Matriz de Similaridade para o índice de Jaccard ($\mathcal{D}_e$)

	Label1	**Label2**	**Label3**	**Label4**	**Label5**
Label1	1.00	0.20	0.75	0.40	0.25
Label2	0.20	1.00	0.25	0.25	0.00
Label3	0.75	0.25	1.00	0.50	0.00
Label4	0.40	0.25	0.50	1.00	0.00
Label5	0.25	0.00	0.00	0.00	1.00

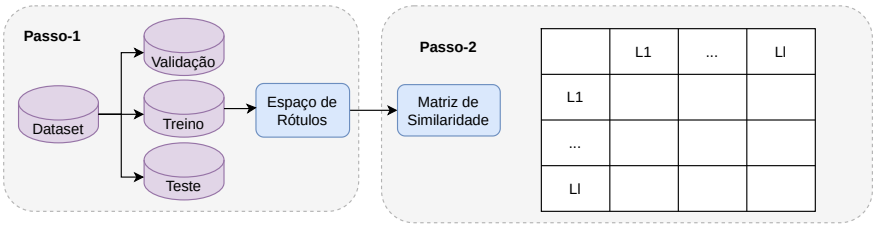


Figura 28 – Passos 1 e 2 da variação HPML.A: divisão do dataset e modelagem das correlações.

matriz em um algoritmo de agrupamento hierárquico. A Tabela 18 apresenta a matriz de dissimilaridade para o índice de Jaccard obtida após a aplicação da Equação 45 à Tabela 17.

$$d(Jaccard) = (1 - Jaccard) \tag{45}$$

$$d(Rogers) = (1 - Rogers) \tag{46}$$

Tabela 18 – HPML.A: Matriz de Dissimilaridade Jaccard ($\mathcal{D}_e$)

	Label1	**Label2**	**Label3**	**Label4**	**Label5**
Label1	0.00	0.80	0.25	0.60	0.75
Label2	0.80	0.00	0.75	0.75	1.00
Label3	0.25	0.75	0.00	0.50	1.00
Label4	0.60	0.75	0.50	0.00	1.00
Label5	0.75	1.00	1.00	1.00	0.00

Um algoritmo de agrupamento hierárquico é um processo de organização de objetos (rótulos, instâncias, etc.) em grupos aninhados. Como resultado desse processo, uma hierarquia de agrupamentos aninhado (dendrograma), similar a uma estrutura de árvore, é gerada. Diferentes partições de dados podem ser encontradas quando o dendrograma é cortado nos níveis onde os grupos se juntam (LUKASOV, 1978; DASH et al., 2003; THEODORIDIS, 2009; MITTAL et al., 2022). Este é o principal motivo pelo qual este algoritmo foi escolhido neste trabalho.

Estes algoritmos são divididos em dois tipos: aglomerativos e divisivos. No caso de algoritmos de agrupamento hierárquico divisivos, os rótulos começam todos juntos (como a partição global) e então, a cada iteração vão sendo divididos até que cada rótulo esteja em um grupo separado (como a partição local). Em contraste, o método aglomerativo utilizado no algoritmo de agrupamento aglomerativo hierárquico começa com n grupos

(partição local onde cada rótulo é um grupo separado) e prossegue por fusões consecutivas até que apenas um grupo seja obtido contendo todos os rótulos (partição global) (KAUFMAN; ROUSSEEUW, 1990; MURTAGH; CONTRERAS, 2017; MITTAL et al., 2019; GOLALIPOUR et al., 2021).

Durante o processo de aglomeração, os grupos são mesclados de acordo com métricas de ligação que calculam a distância entre dois grupos (XU; TIAN, 2015). Cinco métricas de ligação podem ser usadas para construir dendrogramas (FACELI et al., 2011; TAN; STEINBACH; KUMAR, 2005):

- **Simples**: calcula a distância mínima ou o vizinho mais próximo. A distância entre dois grupos é definida como a distância entre seus dois membros mais próximos. Produz grupos nos quais os rótulos são adicionados sequencialmente a um único grupo (FRIGUI, 2008; THEODORIDIS; KOUTROUMBAS, 2009);

- **Completa**: calcula a distância máxima ou vizinho mais distante. A distância entre dois grupos é definida como a distância entre seus dois membros mais distantes. Produz grupos bem separados e compactos (FRIGUI, 2008; THEODORIDIS; KOUTROUMBAS, 2009);

- **Média**: calcula a distância média entre dois grupos de forma que esses dois grupos tenham uma influência igual no resultado final. A distância entre dois grupos é definida como a distância média entre cada um dos seus membros (FRIGUI, 2008; THEODORIDIS; KOUTROUMBAS, 2009);

- **McQuitty**: Produz grupos com características entre as métricas de ligação simples e completa (MCQUITTY, 1966; THEODORIDIS; KOUTROUMBAS, 2009);

- **Ward.D**: O método de variância mínima de Ward visa encontrar aglomerados esféricos compactos. O método é baseado no valor ótimo de uma função objetiva para escolher o par de grupos que serão mesclados a cada iteração (WARD, 1963; MURTAGH; LEGENDRE, 2014; THEODORIDIS; KOUTROUMBAS, 2009). O pacote utilizado neste trabalho fornece duas implementações diferentes para esta métrica, Ward.D e Ward.D2, e ambas foram testadas. A diferença entre elas está na implementação do critério de agrupamento de Ward de 1963, a primeira não considera, enquanto a segunda sim. Em Ward.D2 as dissimilaridades são elevadas ao quadrado antes da atualização do cluster[1] .

A Figura 29 apresenta os dendrogramas resultantes de cada uma das métricas de ligação para ($\mathcal{D}_e$). Um coeficiente pode ser usado para escolher um entre os cinco dendrogramas que, de acordo com este coeficiente, é considerado o melhor. O dendrograma

[1] https://www.rdocumentation.org/packages/fastcluster/versions/1.2.3/topics/hclust

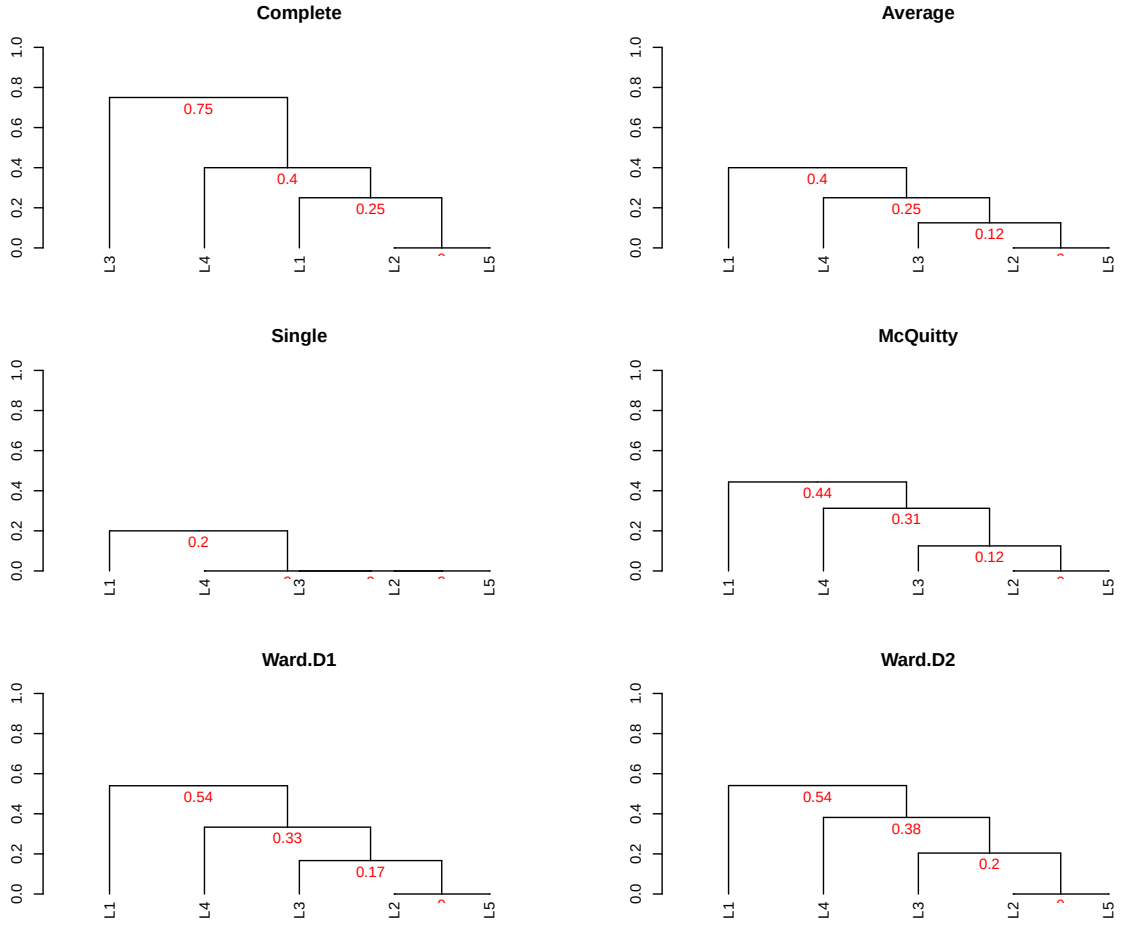


Figura 29 – Dendrogramas para a Matriz de Dissimilaridade de Jaccard da variação HPML.A

escolhido é então cortado em níveis para produzir as partições. O Coeficiente Aglomerativo (CA) usa a matriz de dissimilaridade para medir a qualidade dos dendrogramas gerados. Considere dois grupos C_r e C_q, e $d(M_i)$ a dissimilaridade entre os grupos C_r e C_q quando eles são mesclados na i-ésima etapa do agrupamento aglomerativo.

Para cada processo de fusão M_i envolvendo dois grupos na i-ésima etapa da aglomeração, a dissimilaridade $d(M_i)$ na etapa i é dividida pela dissimilaridade calculada na etapa $i-1$. O valor do coeficiente aglomerativo é, portanto, a média de todas as dissimilaridades $1 - d(M_i)$. Quanto maior o valor do coeficiente aglomerativo, melhor é o dendrograma, assim o maior CA entre as três métricas é usado para escolher o dendrograma que será cortado (KAUFMAN; ROUSSEEUW, 1990).

Uma vez que o melhor dendrograma é escolhido, ele é cortado para produzir as partições híbridas - cada corte representa uma partição diferente. A Tabela 19 ilustra as

partições produzidas cortando o dendrograma da Figura 29 para a métrica de ligação completa em todos os níveis possíveis. O espaço de rótulos do dataset de exemplo é composto por 5 rótulos, portanto, o dendrograma pode ser cortado em um total de 5 níveis, produzindo portanto 5 diferentes particionamentos, que consequentemente são todas as possíveis. Dentro desses 5 particionamentos estão a partição local (P5), a partição global (P1) e três partições híbridas (P2, P3 e P4).

As colunas da Tabela 19 representam partições e as linhas representam os rótulos. Os números em cada célula representam os grupos (G) ao quais os rótulos nas linhas foram atribuídos. Nota-se que cortar o dendrograma no nível 1 gera a partição local ($P1$), enquanto que um corte no nível 5 gera a partição global ($P5$). Todos os outros cortes geram partições híbridas.

Tabela 19 – Partições $\mathcal{D}_e$ encontradas pela variação HPML.A.

	P5	**P4**	**P3**	**P2**	**P1**
Label1	1	1	1	1	1
Label2	1	1	1	2	2
Label3	1	1	2	3	3
Label4	1	2	3	4	4
Label5	1	1	1	1	5

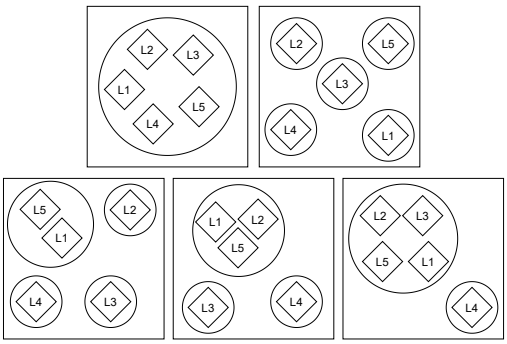


Figura 30 – Representação das Partições $\mathcal{D}_e$ encontradas pela variação HPML.A

Cortar o dendrograma no nível 3 resulta em uma partição com três grupos de rótulos[2]: $G_1 = \{Label1, Label2, Label5\}$, $G_2 = \{Label3\}$ e $G_3 = \{Label4\}$. Esta partição representa diferentes correlações de rótulos se comparada à partição global onde todos os rótulos estão em um único grupo ($G1 = \{Label1, Label2, Label3, Label4, Label5\}$), e à partição local onde cinco grupos estão presentes, cada um com um rótulo diferente ($G1 = \{Label1\}$, $G2 = \{Label2\}$, $G3 = \{Label3\}$, $G4 = \{Label4\}$, e $G5 = \{Label5\}$).

A Figura 31 ilustra o processo do Passo-3 e a Figura 32 ilustra os datasets construídos para cada uma das partições da Figura 30. Os datasets são construídos considerando todas as instâncias correspondentes ao conjunto de dados utilizado - ou treino, ou validação ou teste dependendo da fase - e apenas os rótulos são selecionados conforme os grupos de

[2] G_i corresponde a um subconjunto de rótulos

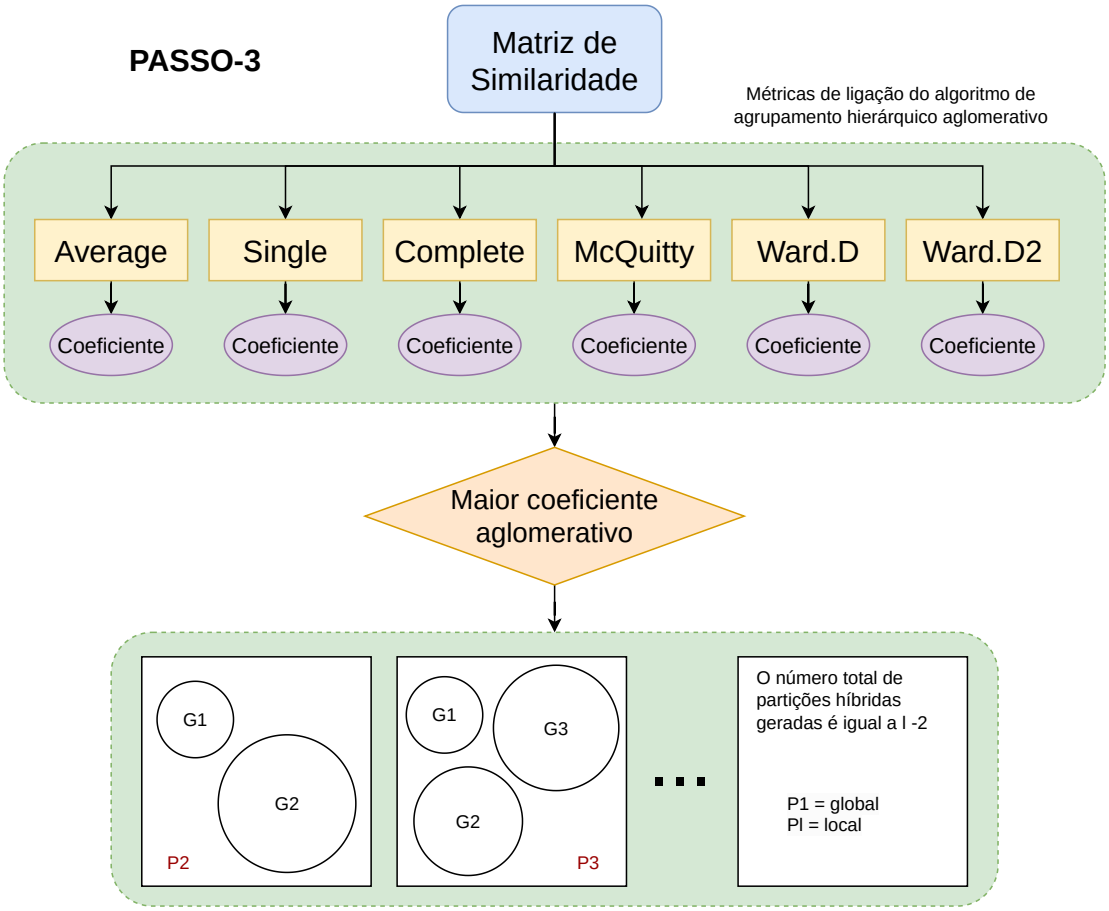


Figura 31 – Passo 3 da variação HPML.A ilustrado: particionamento do espaço de rótulos. O número total de partições híbridas geradas é igual ao número total de rótulos (l) existente no espaço de rótulos menos 2, que corresponde às partições local e global que também são encontradas durante o processo.

cada partição. Na implementação das versões, para facilitar a codificação dos diversos tipos de particionamento, análises e gerar menos confusão, os nomes das partições foram renomeados. A partição 4 que originalmente tem 2 grupos foi renomeada como P2, a partição 2 que tem 4 grupos foi renomeada como P4, e assim por diante. Dessa forma o número de grupos (G) da partição (P) combina com o nome da partição P_G e, por este motivo as partições híbridas serão referenciadas como P2 = 2 grupos, P3 = 3 grupos, etc.

4.1.3 Validação e escolha da melhor partição com um classificador

Neste trabalho, optou-se por validar as partições de duas formas: 1. melhor desempenho preditivo; e 2. melhor coeficiente de silhueta. Para exemplificar como as partições são validadas usando o critério de melhor desempenho preditivo, considere que a partição $P4$ (que na implementação se torna $P2$) da Tabela 19 está sendo validada. Esta partição é formada por 2 grupos: $G_1 = \{Label1, Label2, Label3, Label4\}$ e $G_2 = \{Label5\}$. Nesse

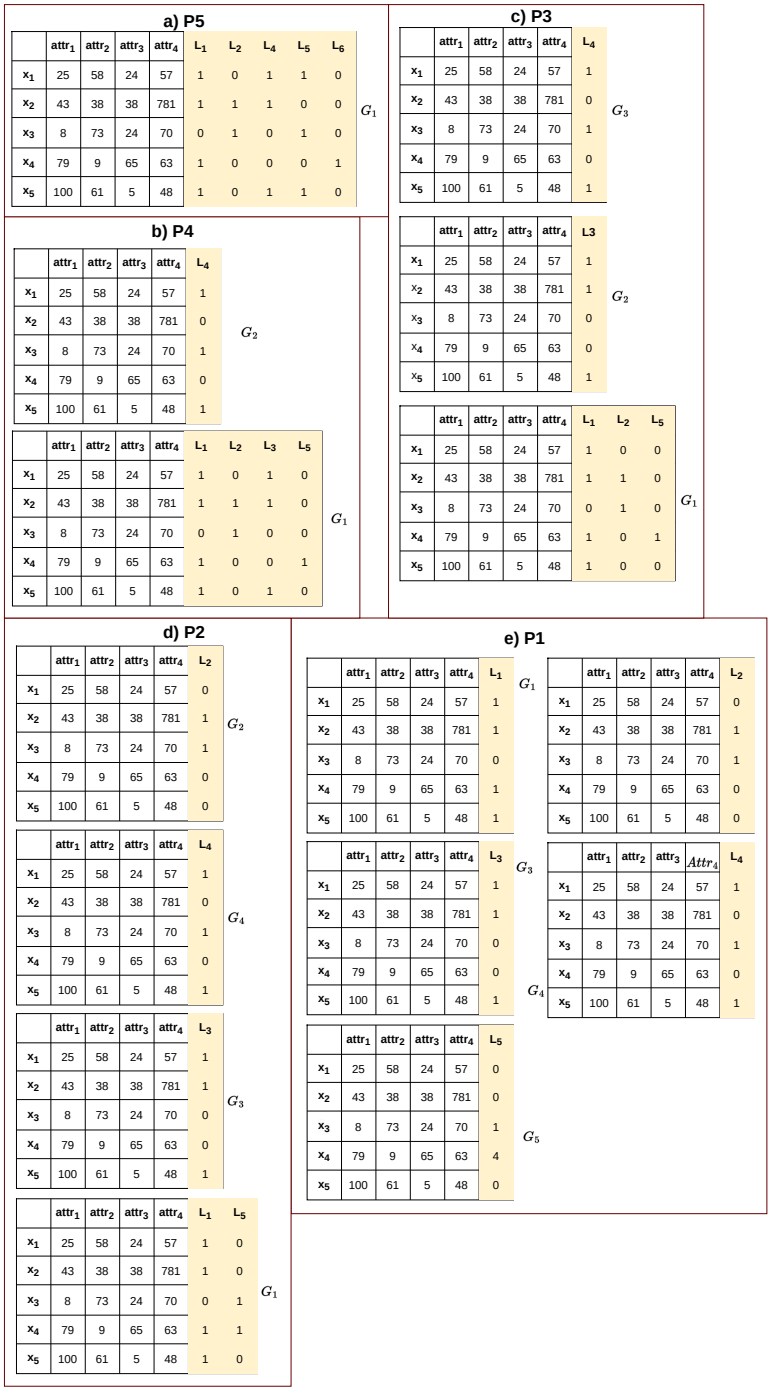


Figura 32 – Datasets para as partições da Figura 30

caso, um classificador multirrótulo é treinado com todas as instâncias que pertencem aos rótulos de G_1, e um classificador binário é treinado com todas as instâncias que pertencem ao rótulo de G_2.

Uma vez que os classificadores foram treinados com o conjunto de treinamento, suas predições são obtidas usando um conjunto de validação separado e então combinadas. No caso da partição $P4$, suponha que uma instância de validação $\mathbf{x}_i$ obtém as seguintes predições individuais: $\hat{Y}_{G_1} = \{1, 0, 1, 0\}$ e $\hat{Y}_{G_2} = \{1\}$. Uma predição final multirrótulo para

$\mathbf{x}_i$ é obtida combinando $\hat{Y}_{G_1}$ e $\hat{Y}_{G_2}$, resultando em $\hat{Y}_{P4} = \{1, 0, 1, 0, 1\}$. Isso é executado para todas as instâncias em um conjunto de dados de validação. A melhor partição gerada é aquela que resulta no melhor desempenho no conjunto de dados de validação, de acordo com alguma medida de avaliação multirrótulo. A Macro-F1 é uma boa medida para problemas multirrótulo, pois considera os desempenhos individuais em cada classe e por isto foi escolhida como critério de seleção nesta fase.

O Clus Framework (VENS et al., 2008), que induz árvores de decisão binárias e multi-rótulo baseadas em PCTs e é considerado um dos métodos do estado da arte na literatura foi escolhido como classificador para esta versão. Um PCT binário é treinado para cada rótulo e suas saídas são combinadas para formar a predição final multirrótulo para a partição local. Para a partição global, apenas um PCT multirrótulo é necessário. As partições híbridas são validadas usando um conjunto de PCTs multirrótulo, ou uma combinação de PCTs binários e multi-rótulo, dependendo de quantos rótulos estão nos clusters da partição que está sendo validada. As saídas individuais são então combinadas para formar a predição multirrótulo final. A Figura 33 ilustra o processo dos passos 5 e 6 com classificador para o HPML.A. Do lado esquerdo são ilustrados os detalhes de como cada partição é validada e do lado direto há uma visão geral com todas as partições. Este critério de seleção de partição híbrida pode ser utilizado por todas as variações da abordagem HPML.

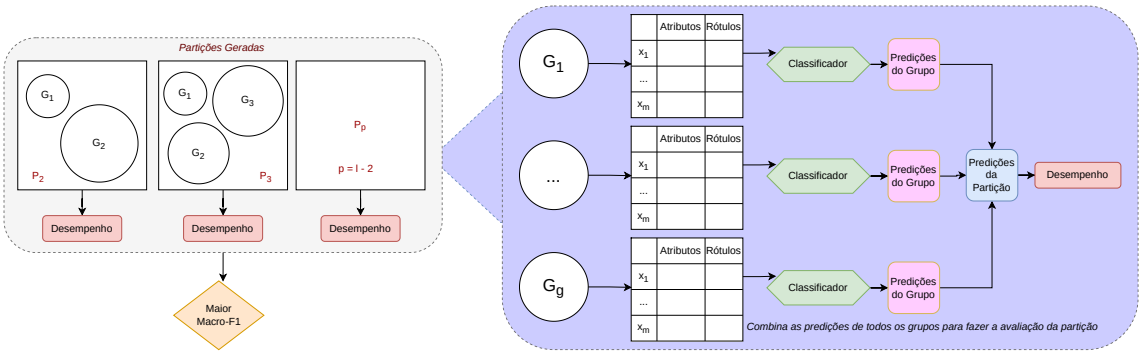


Figura 33 – Validação com Classificador. Para grupos compostos com um único rótulo é induzida a versão local do classificador, enquanto em grupos compostos por mais de um rótulo, a versão global é induzida.

4.1.4 Validação e escolha da melhor partição híbrida com o coeficiente da silhueta

O coeficiente da silhueta foi escolhido como critério de seleção de melhor partição híbrida por ser um método de validação de agrupamento e a Figura 34 ilustra este processo. No caso das partições híbridas, a qualidade do grupo pode ser definida a partir da proximidade entre os rótulos de um determinado grupo e da distância entre esses rótulos e o

grupo mais próximo. Além disso, o coeficiente da silhueta pode ser usado para escolher um número ideal de grupos (ROUSSEEUW, 1987).

A Figura 35 ilustra o conceito do coeficiente da silhueta. Para cada objeto i do conjunto de dados, é necessário calcular $a(i)$ que é a dissimilaridade média entre i e todos os outros pontos do grupo ao qual i pertence. Se i for único em seu grupo então a similaridade é zero, $s(i) := 0$. Para todos os outros grupos é necessário calcular o $d(i, C)$ que é a dissimilaridade média de i para todas as observações do grupo. O menor deles é $b(i) := min_C d(i, C)$, e pode ser entendido como a dissimilaridade entre i e seu grupo vizinho. O cálculo da similariadde $s(i)$ é apresentado na Equação 47 (ROUSSEEUW, 1987).

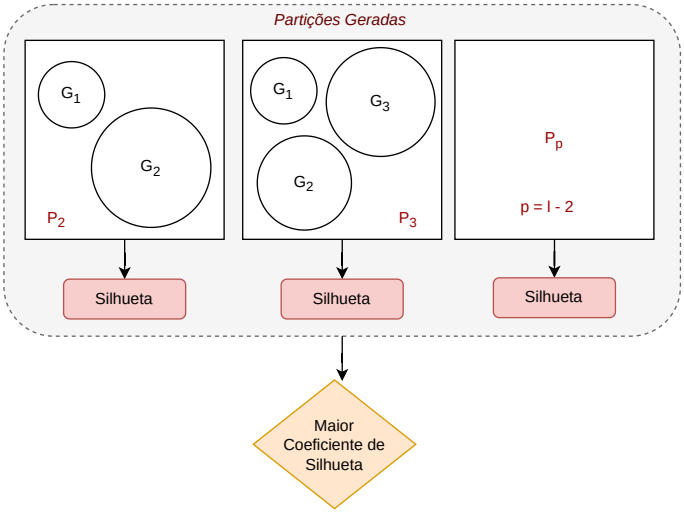


Figura 34 – Validação com coeficiente da silhueta

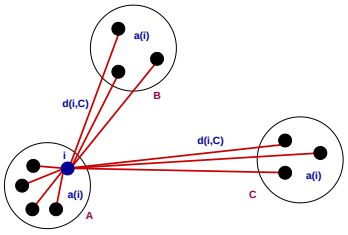


Figura 35 – Coeficiente da silhueta: na Figura estão ilustrados três grupos de uma mesma partição híbrida, os quais compostos por objetos que neste contexto são os rótulos.

$$s(i) := \frac{b(i) - a(i)}{max(a(i), b(i))} \tag{47}$$

Usando este método, é necessário transpor o espaço de rótulos e adicionar a informação do grupo de cada rótulo, visto que se deseja obter o coeficiente da silhueta para grupos de rótulos e não grupos de atributos de entrada. Somente após isso é possível efetuar

o cálculo. A partição híbrida com o coeficiente de silhueta mais alto é então escolhida para o teste. Este critério de seleção de partição híbrida pode ser utilizado por todas as variações da abordagem HPML.

4.1.5 Teste da Partição Híbrida Escolhida

A Figura 36 ilustra o processo de teste para as partições escolhidas tanto com o coeficiente da silhueta quanto com um classificador. Depois de selecionar a melhor partição usando os conjuntos de dados de treinamento e validação, um classificador é induzido na melhor partição híbrida com o conjunto de teste. Os resultados do teste são comparados com os resultados obtidos pelos classificadores usando as partições locais e globais convencionais, e também aleatórias. Este processo é idêntico para todas as três variações da abordagem: HPML.A, HPML.B e HPML.C.

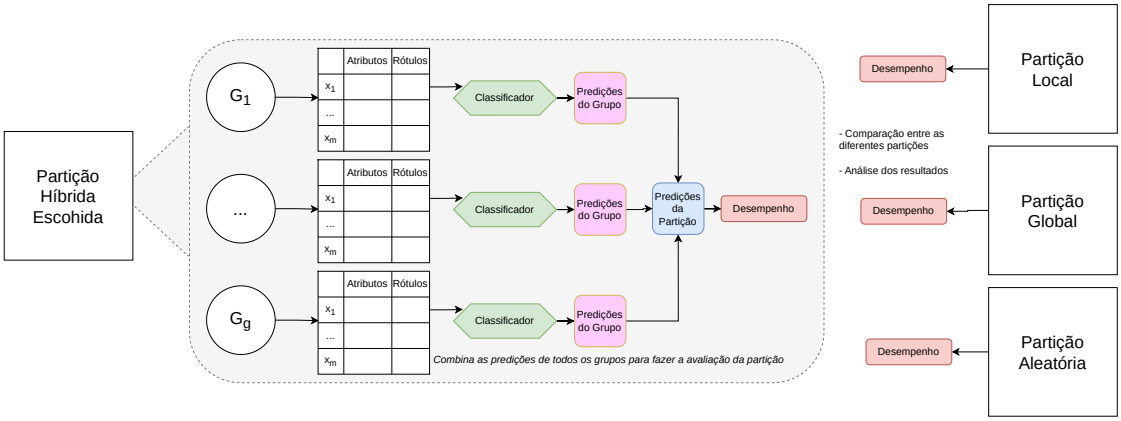


Figura 36 – Teste da partição híbrida escolhida. Este processo é idêntico para todas as três variações da abordagem: HPML.A, HPML.B e HPML.C

4.2 HPML versão B

A versão HPML.B modela as correlações usando o Mapa Auto Organizável de Kohonen e particiona o espaço de rótulos cortando o mapa em grupos de neurônios. No entanto, com o Kohonen as partições híbridas não são obtidas diretamente. Na verdade obtem-se partições que são compostas por grupos de neurônios com instâncias similares e, portanto, é necessário transformar essas partições de instâncias em partições de rótulos. Assim como HPML.A, para validar as partições híbridas usa tanto um classificador, quanto calcula o coeficiente da silhueta e, por este motivo, estes passos não serão re-explicados nesta seção, visto que já foram detalhados anteriormente. A seguir detalhes de como o método HPML.B funciona serão explicados.

4.2.1 Modelagem das Correlações

Em 1982, Kohonen propôs um método competitivo denominado *Self-Organizing Map* (SOM). Os neurônios são posicionados em uma grade bidimensional e, após o aprendizado, forma-se algo como um mapa topográfico dos padrões de entrada, semelhante à maneira como os estímulos sensoriais humanos são mapeados em diferentes partes do cérebro. Para garantir que o processo de auto-organização ocorra adequadamente, é necessário que todos os neurônios da rede sejam expostos a um número suficiente de diferentes padrões de entrada. Por esse motivo cada neurônio da camada de entrada está conectado com todos os neurônios da camada de saída. A Figura 37 ilustra uma arquitetura de rede neural artificial para o SOM (HAYKIN, 2011).

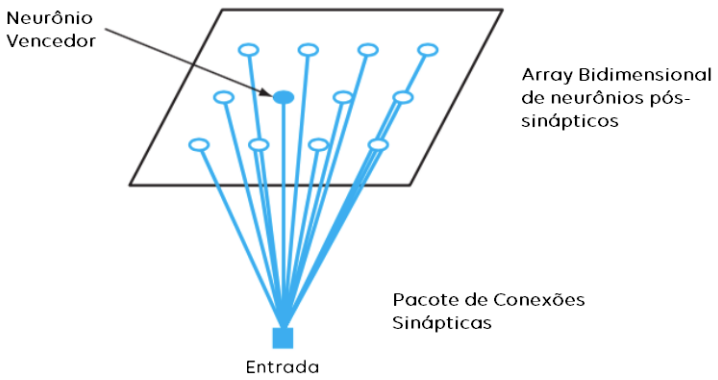


Figura 37 – Mapa Auto-Organizável de Kohonen. Fonte: Haykin (2011).

O principal objetivo do SOM, portanto, é mapear um padrão de entrada de dimensão arbitrária em um mapa bidimensional, de maneira adaptativa e ordenada topologicamente (HAYKIN, 2011). Devido a essas características dos mapas auto-organizáveis, juntamente com o fato de que esses algoritmos realizam o agrupamento de dados sem a necessidade do fornecimento inicial de número fixo de grupos para cada partição híbrida - ainda que o número de neurônios seja fornecido - é que o SOM foi escolhido como uma das técnicas iniciais para investigar o particionamento do espaço de rótulos.

No SOM os pesos da rede recebem, inicialmente, valores aleatórios e, após inicializados, três processos são executados: competição, cooperação e adaptação sináptica. Na competição, para cada padrão de entrada, os neurônios calculam seus respectivos valores da função discriminante, a qual fornece a base para a competição entre os neurônios, sendo o neurônio vencedor aquele com o melhor valor dessa função.

Na cooperação, o neurônio vencedor determina a localização espacial de uma vizinhança topológica de neurônios excitados, o que fornece base para cooperação dessa vizinhança. Para definir a vizinhança, diferentes funções podem ser utilizadas, como por exemplo a função Gaussiana. Na Adaptação Sináptica, os neurônios excitados melhoram seus valores individuais da função discriminante, sendo o ajuste dos pesos feito para melhorar a resposta do neurônio vencedor à aplicação de um padrão de entrada similar.

Esse processo é dividido ainda entre outras duas fases: auto-organização, em que ocorre a ordenação topológica dos vetores de pesos, e convergência, em que ocorre o ajuste fino dos pesos do mapa (HAYKIN, 2011).

Portanto, uma função de vizinhança deve ser definida para a rede SOM, assim como a sua arquitetura e taxa de aprendizado. O algoritmo deve ser inicializado, em seguida ocorre a amostragem, em que as instâncias do espaço de instâncias (ou o espaço de rótulos), são apresentados à rede de forma aleatória. Depois é preciso encontrar o neurônio que melhor combina com o padrão selecionado na amostragem (melhor função discriminante) e, por fim, é feita a atualização dos pesos. Todo esse processo é repetido um número de vezes (fornecido pelo usuário) até se obter o mapa final.

Neste trabalho, o mapa de Kohonen recebeu como entrada apenas o espaço de rótulos pois o que se deseja é mapear apenas os rótulos. A entrada para o mapa de Kohonen pode ser considerada uma matriz, onde cada linha dessa matriz é tratada como um vetor do conjunto de dados que será mapeado. Assim, para o espaço de rótulos do conjunto de dados multirrótulo, o vetor é binário e a entrada do espaço de rótulos pode ser representada como $\mathbf{x}_i = [Label_1, Label_2, ..., Label_l]$. Portanto, para o mapa de Kohonen o espaço de rótulos pode ser tratado como uma matriz composta por valores $0s$ e $1s$. A Figura 38 ilustra os Passos 1 e 2 para o HPML.B.

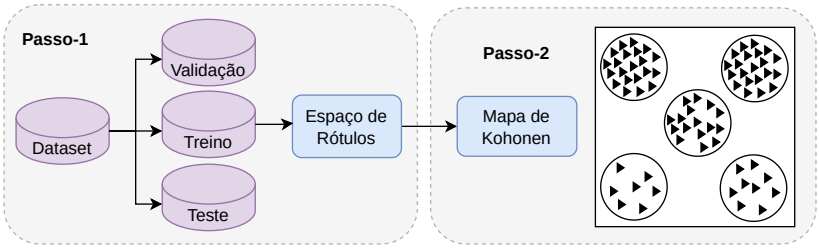


Figura 38 – Passos 1 e 2 da variação HPML.B ilustrados: divisão do dataset e modelagem das correlações. Triângulos representam as instâncias e os círculos neurônios compostos por instâncias similares.

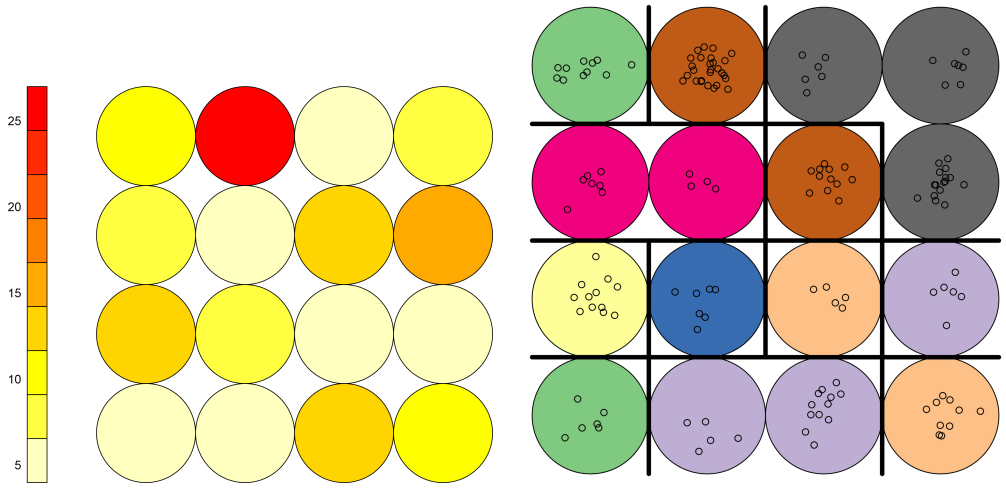
As linhas são as instâncias do espaço de rótulos e as colunas são os rótulos. O mapa de Kohonen atribui instâncias (após a competição) similares a cada neurônio vencedor do mapa. Essas instâncias mapeadas são compostas por vários rótulos diferentes, sendo necessário averiguar quais rótulos estão presentes em cada neurônio, podendo haver repetição desses rótulos em cada um dos neurônios.

Para demonstrar com mais clareza como Kohonen pode ser aplicado na estratégia aqui proposta, considere o conjunto de dados Flags (GONCALVES; PLASTINO; FREITAS, 2013) que tem 7 rótulos e 194 instâncias. A Figura 39 apresenta os gráficos de contagem (a) e mapeamento (b). Um gráfico de contagem (Figura 39(a)) mostra o número de instâncias mapeadas para cada neurônio sendo que as unidades vazias são representadas na cor cinza, enquanto que um gráfico de mapeamento (Figura 39(b)) mostra em quais

neurônios as instâncias são mapeadas.

O mapa de Kohonen utilizado para gerar os gráficos exige que se defina o número de neurônios do mapa em duas dimensões (x e y). A dimensão é então definida tendo como base o gráfico de contagem: se a dimensão escolhida gerar neurônios cinzas no mapa de contagem, a dimensão deve ser redefinida até que se encontre uma dimensão onde não haja neurônios vazios.

Considerando isto, no gráfico de contagem aqui apresentado não há nenhum neurônio na cor cinza. Como pode ser observado, o mapa foi configurado com 16 neurônios (dimensão 4×4). Para a Figura 39(a) se o mapa tivesse a dimensão 5×5, haveriam neurônios na cor cinza, indicando assim que não foram mapeadas instâncias para esses neurônios.



(a) Gráfico de Contagem (b) Gráfico de Mapeamento

Figura 39 – Mapas de Kohonen. O gráfico de contagem ilustra as instâncias mapeadas em cada neurônio, enquanto que o gráfico de mapeamento mostra o mapa de kohonen cortado gerando agrupamentos de neurônios.

4.2.2 Particionando o Espaço de Rótulos

A Figura 39(a) apresenta o mapa de kohonen encontrado para o conjunto de dados Flags e a Figura 39(b) ilustra os cortes no mapa, como se fossem cortes em um dendrograma. No entanto, as cores definem os grupos de neurônios e as linhas tentam apenas separar os neurônios conforme as cores. O mapa foi dividido em 8 grupos de neurônios (8 cores) e cada cor representa um grupo de instâncias similares.

Para chegar nesses grupos de neurônios, um método de corte similar ao que é aplicado no dendrograma do índice Jaccard é usado no mapa de kohonen. No HPML-A é necessário primeiro usar o algoritmo de agrupamento aglomerativo hierárquico e depois aplicar o corte, mas no caso do Kohonen aplica-se apenas o corte diretamente no mapa, portanto, o mapa não é submetido ao algoritmo de agrupamento aglomerativo hierárquico. As

partições do espaço de rótulos são obtidas ao se cortar o mapa de kohonen em vários níveis, como se fose um dendrograma, onde cada corte produz grupos de neurônios.

Os níveis do corte correspondem à dimensão do mapa. Por exemplo, o mapa da Figura 39(b) pode ser cortado em 16 níveis diferentes pois o mapa foi construído com 16 neurônios. Isto é diferente da abordagem do HPML.A onde os cortes no dendrograma ocorrem de acordo com o número total de rótulos no espaço de rótulos. Além disso, o primeiro e último corte geram a partições global e local e, portanto, são descartadas. Esse descarte das partições local e global ocorrem em todas as abordagens já que todas elas geram essas partições.

Diferentemente do HPML-A, as partições obtidas ao cortar o mapa de Kohonen não possuem agrupamentos disjuntos de rótulos, ou seja, um rótulo em Kohonen pode ser mapeado para mais de um neurônio simultaneamente. Portanto, é necessária uma estratégia para selecionar quais rótulos permanecerão em cada grupo. Uma estratégia simples foi adotada e é ilustrada na Figura 40: i) identificar quais rótulos estão presentes em cada grupo; ii) obter a frequência com que cada rótulo aparece em cada grupo; iii) comparar as frequências de cada rótulo em cada grupo, e iv) manter o rótulo no grupo correspondente à sua maior frequência.

Por exemplo, para a partição da Figura 39(b) os 8 grupos iniciais poderão se tornar apenas 2 ou 3. O número de grupos mudará como resultado dessa estratégia, e o número de grupos para cada partição não ficará claro, o que é um recurso útil, visto que é interessante ter várias maneiras de obter partições híbridas. A Figura 41 ilustra o Passo-3 para o HPML.B.

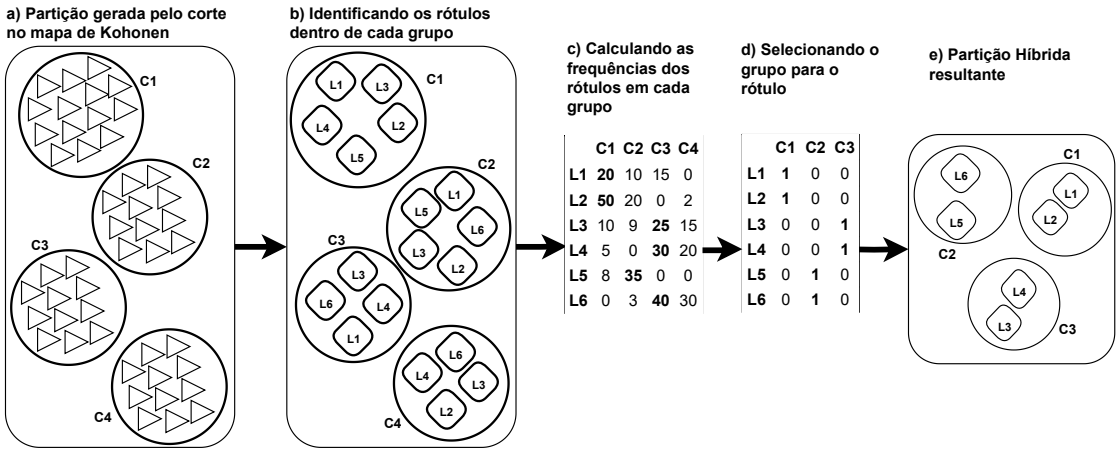


Figura 40 – Estratégia de transformação das partições geradas com o Kohonen para partições híbridas.

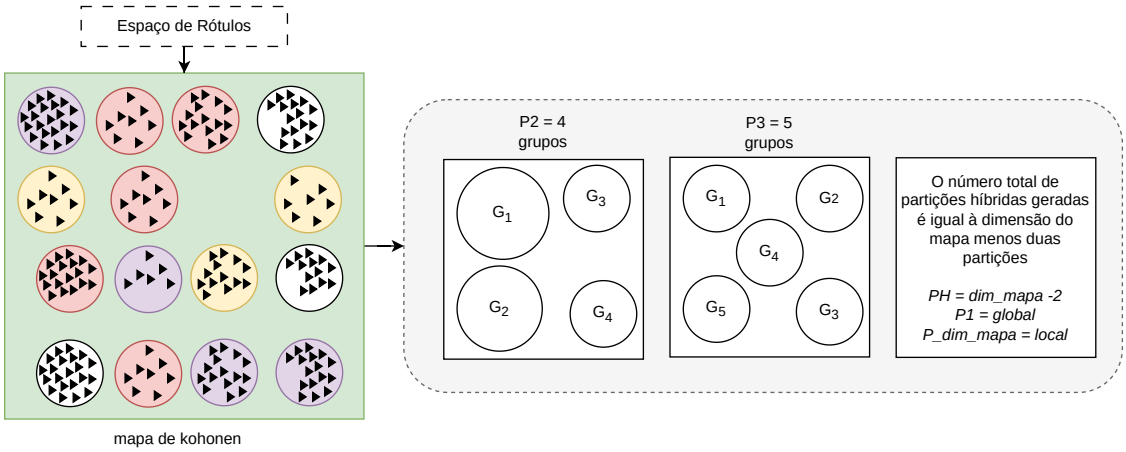


Figura 41 – Passo 3 da abordagem HPML.B ilustrado: particionamento do espaço de rótulos.

4.3 HPML versão C

A terceira versão das partições híbridas é a HPML.C que faz uso dos métodos de detecção de comunidades para modelar as correlações, a qual tem sua origem na teoria dos grafos. A motivação para usar métodos de detecção de comunidade é que a topologia da rede pode codificar interações entre os dados sistematicamente e encontrar relacionamentos entre eles, e nas partições híbridas são os relacionamentos entre rótulos (MITTAL; BHATIA, 2020; SILVA; ZHAO, 2016). Novamente, os passos de construção dos datasets, validação, escolha e teste das partições não serão explicados aqui pois é o mesmo processo das abordagens HPML.A e HPML.B.

Os métodos de detecção de comunidade também são algoritmos de particionamento de grafos: eles dividem os vértices em grupos minimizando o número de arestas entre eles. A grosso modo, uma comunidade (grupo) é um conjunto de vértices com muitas arestas dentro da comunidade e algumas arestas fora dela, característica desejada para as partições híbridas (SILVA; ZHAO, 2016). Os rótulos então podem ser considerados vértices e as correlações entre eles, as arestas. Dessa forma, um grafo de rótulos correlacionados pode ser construído e particionado para encontrar comunidades de rótulos.

4.3.1 Modelagem das Correlações

Para gerar partições híbridas usando métodos de detecção de comunidade um grafo de co-ocorrências de rótulos deve ser gerado a partir de uma matriz de similaridade de rótulos. Essa matriz representa um grafo completo modelado com as correlações onde uma aresta conecta cada um dos pares de vértices distintos. As matrizes de similaridades já foram explicadas na seção 4.1 e, portanto aqui será descrito como elas são usadas para gerar os grafos (HUANG et al., 2021; MITTAL; BHATIA, 2020).

Segundo Silva e Zhao (2016), é necessário uma matriz de similaridade ou dissimilaridade para construir uma rede a partir de dados vetoriais, o que possibilita estabelecer ligações entre pares de rótulos com pesos de acordo com aquela matriz. No entanto, ligações com pesos pequenos podem levar a resultados ruins e podem ser considerados como ruídos que podem fornecer informações erradas ao algoritmo de aprendizado de máquina.

Para evitar isso, deve-se aplicar a *esparcificação*, uma técnica de corte de arestas. A esparcificação tem como objetivo remover arestas que não agregam valor ao grafo e, portanto, é uma segunda fase na modagem do grafo, sendo a primeira correspondente ao cálculo das similaridades. Um tipo tradicional de esparcificação é baseado no conceito de k vizinhos mais próximos, onde os cortes ocorrem nas arestas que não fazem parte da vizinhança. Outro tipo de esparcificação permite um limite (threshold) que corta alguma porcentagem das arestas e ambos os tipos são usados aqui (SILVA; ZHAO, 2016).

Exemplificando, considere o espaço de rótulos da Tabela 16. O mesmo é composto por 5 rótulos e como consequência a matriz de similariade terá tamanho 5 x 5. Em cada uma das 25 posições da matriz estarão os valores de similaridades calculados para cada par de rótulos possível, conforme já explicado.

No entanto, essa matriz não pode ser utilizada desta forma para modelar o grafo, ela precisa ser transformada em uma tabela composta por vértice de origem, vértice de destino, valor de similaridade e valor do peso da aresta (o qual pode ser o mesmo da similaridade). A matriz de similaridade apresentada na Tabela 17 ficará como a Tabela 20 que aqui será referida como *Tabela de Similaridade*. Ao fim de cada rótulo e seus respectivos pares na tabela, uma linha foi adicionada, permitindo melhor visualização das conexões.

A esparcificação por $k-$NN modifica a importância da aresta. Uma forma de se aplicar esta esparcificação é escolher os k maiores valores para uma aresta, isto é, para cada vértice são considerados apenas os k maiores valores de similaridade que estão na coluna pesos da Tabela 20. Os pares de vértices correspondentes a esse critério ficarão intactos, enquanto o restante terá o seu valor alterado para zero (nenhuma importância). Por exemplo, define-se que apenas os 3 maiores valores de pesos serão considerados ($k-\text{NN} = 3$), aplicando isto à Tabela 20 obtem-se como resultado a Tabela 21.

Os self-loops também devem ser retirados pois esses pares de rótulos geram arestas para si mesmos e não conectam com outros, então eles não são úteis na criação do grafo e não agregam informação relevante. Neste exemplo, as conexões entre $Label1 - Label1$, $Label2 - Label2$, $Label3 - Label3$, $Label4 - Label4$ e $Label5 - Label5$ são os self loops. Depois de aplicar a esparcificação o grafo pode ser finalmente construído. A Figura 42 apresenta os grafos construídos para a Tabela 20 com e sem self-loops. Observe que os vértices do grafo da Figura 42a possui laços, isto é, arestas que retornam a eles mesmos, enquanto que o grafo da Figura 42b não.

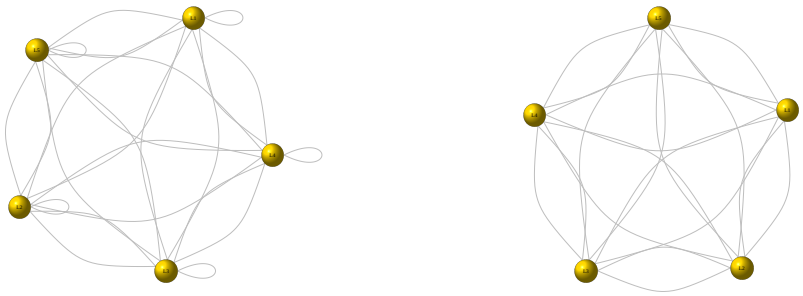
Ainda com relação à esparcificação com k-NN, outros pontos devem ser levados em consideração para gerar os grafos completos para os datasets multirrótulo: a) Valores de k:

Tabela 20 – Tabela de Similaridade

Origem	Destino	Jaccard	Peso
Label1	Label1	1.00	1.00
Label1	Label2	0.20	0.20
Label1	Label3	0.75	0.75
Label1	Label4	0.40	0.40
Label1	Label5	0.25	0.25
Label2	Label1	0.20	0.20
Label2	Label2	1.00	1.00
Label2	Label3	0.25	0.25
Label2	Label4	0.25	0.25
Label2	Label5	0.00	0.00
Label3	Label1	0.75	0.75
Label3	Label2	0.25	0.25
Label3	Label3	1.00	1.00
Label3	Label4	0.50	0.50
Label3	Label5	0.00	0.00
Label4	Label1	0.40	0.40
Label4	Label2	0.25	0.25
Label4	Label3	0.50	0.50
Label4	Label4	1.00	1.00
Label4	Label5	0.00	0.00
Label5	Label1	0.25	0.25
Label5	Label2	0.00	0.00
Label5	Label3	0.00	0.00
Label5	Label4	0.00	0.00
Label5	Label5	1.00	1.00

Tabela 21 – Sparcificação com k-NN

Origem	Destino	Jaccard	Peso
Label1	Label5	0.25	0.00
Label1	Label2	0.20	0.00
Label2	Label1	0.20	0.00
Label2	Label5	0.00	0.00
Label3	Label2	0.25	0.00
Label3	Label5	0.00	0.00
Label4	Label2	0.25	0.00
Label4	Label5	0.00	0.00
Label5	Label2	0.00	0.00
Label5	Label3	0.00	0.00
Label5	Label4	0.00	0.00
Label2	Label3	0.25	0.25
Label2	Label4	0.25	0.25
Label5	Label1	0.25	0.25
Label1	Label4	0.40	0.40
Label4	Label1	0.40	0.40
Label3	Label4	0.50	0.50
Label4	Label3	0.50	0.50
Label1	Label3	0.75	0.75
Label3	Label1	0.75	0.75
Label1	Label1	1.00	1.00
Label2	Label2	1.00	1.00
Label3	Label3	1.00	1.00
Label4	Label4	1.00	1.00
Label5	Label5	1.00	1.00



(a) Grafo com self-loops

(b) Grafo sem self-loops

Figura 42 – Exemplos de Grafos gerados com a abordagem HPML.C. Fonte: Elaborado pela autora.

é dependente do tamanho do espaço de rótulos, mas para todos os datasets 5 valores são utilizados neste trabalho; e b) Tamanho do espaço de rótulos: neste trabalho, se o dataset multirrótulo possui menos de 5 rótulos, k é gerado sequencialmente e, para espaços de rótulos maiores que estes, k é gerado aleatoriamente. Essas regras foram definidas para garantir que o grafo gerado seja completo.

Um valor máximo de k que corresponde a 70% do total de rótulos no espaço de rótulos é calculado: $k.max = (0.7 \times l)$ e o valor mínimo é sempre $k.min = 1$. Se $k.max < 5$, então os valores de k são $1, 2, 3$ e 4. Mas se $k.max >= 5$, então 5 valores aleatórios entre

1 e $k.max$ são gerados e utilizados como os k vizinhos mais próximos.

Em cada iteração, um grafo diferente é gerado de acordo com o corte de k. Todos os rótulos permanecem como vértices, mas o número de arestas conectadas a cada um muda já que apenas as k arestas mais próximas serão conectadas a eles. Para o dataset de exemplo utilizado aqui $l = 5$, $k.max = 4$ e $k = \{1, 2, 3, 4\}$. A Tabela 22 apresenta o resultado da esparcificação com k-NN para o dataset de exemplo. Note que quando $k = 1$ apenas um vizinho (uma linha) é considerado para o rótulo (vértice) e em todas as conexões seguintes o peso tem o valor zero. No caso de $k = 4$, todas as conexões são consideradas pois, no caso deste exemplo, um rótulo pode ter no máximo 4 vizinhos.

Como este dataset de exemplo tem poucas instâncias, claramente há pesos com valor zero pois a similaridade original tem esse valor. Neste caso, a esparcificação $k = 4$ não faz sentido e, portanto, o corte máximo de vizinhos está em $k = 3$ para o dataset de exemplo.

Tabela 22 – HPLM.C: Tabelas de Similaridades para os cortes do k-NN da abordagem HPLM.C

k = 1				**k = 2**				**k = 3**				**k = 4**			
origem	**destino**	**jaccard**	**peso**	**origem**	**destino**	**jaccard**	**peso**	**origem**	**destino**	**jaccard**	**peso**	**origem**	**destino**	**jaccard**	**peso**
L1	L3	0.75	0.75	L1	L3	0.75	0.75	L1	L3	0.75	0.75	L1	L3	0.75	0.75
L1	L4	0.4	0	L1	L4	0.4	0.4	L1	L4	0.4	0.4	L1	L4	0.4	0.4
L1	L5	0.25	0	L1	L5	0.25	0	L1	L5	0.25	0.25	L1	L5	0.25	0.25
L1	L2	0.2	0	L1	L2	0.2	0	L1	L2	0.2	0	L1	L2	0.2	0.2
L2	L3	0.25	0.25	L2	L3	0.25	0.25	L2	L3	0.25	0.25	L2	L3	0.25	0.25
L2	L4	0.25	0	L2	L4	0.25	0.25	L2	L4	0.25	0.25	L2	L4	0.25	0.25
L2	L1	0.2	0	L2	L1	0.2	0	L2	L1	0.2	0.2	L2	L1	0.2	0.2
L2	L5	0	0	L2	L5	0	0	L2	L5	0	0	L2	L5	0	0
L3	L1	0.75	0.75	L3	L1	0.75	0.75	L3	L1	0.75	0.75	L3	L1	0.75	0.75
L3	L4	0.5	0	L3	L4	0.5	0.5	L3	L4	0.5	0.5	L3	L4	0.5	0.5
L3	L2	0.25	0	L3	L2	0.25	0	L3	L2	0.25	0.25	L3	L2	0.25	0.25
L3	L5	0	0	L3	L5	0	0	L3	L5	0	0	L3	L5	0	0
L4	L3	0.5	0.5	L4	L3	0.5	0.5	L4	L3	0.5	0.5	L4	L3	0.5	0.5
L4	L1	0.4	0	L4	L1	0.4	0.4	L4	L1	0.4	0.4	L4	L1	0.4	0.4
L4	L2	0.25	0	L4	L2	0.25	0	L4	L2	0.25	0.25	L4	L2	0.25	0.25
L4	L5	0	0	L4	L5	0	0	L4	L5	0	0	L4	L5	0	0
L5	L1	0.25	0.25	L5	L1	0.25	0.25	L5	L1	0.25	0.25	L5	L1	0.25	0.25
L5	L2	0	0	L5	L2	0	0	L5	L2	0	0	L5	L2	0	0
L5	L3	0	0	L5	L3	0	0	L5	L3	0	0	L5	L3	0	0
L5	L4	0	0	L5	L4	0	0	L5	L4	0	0	L5	L4	0	0

Tabela 23 – Tabelas de Similaridade para cada corte do threshold da abordagem HPLM.C

Threshold-1				**Threshold-2**				**Threshold-3**				**Threshold-4**			
origem	**destino**	**jaccard**	**peso**	**origem**	**destino**	**jaccard**	**peso**	**origem**	**destino**	**jaccard**	**peso**	**origem**	**destino**	**jaccard**	**peso**
L1	L4	0.4	0.4	L1	L4	0.4	0.4	L1	L5	0.25	0.25	L1	L2	0.2	0.2
L1	L5	0.25	0.25	L1	L5	0.25	0.25	L1	L2	0.2	0.2	L2	L1	0.2	0.2
L1	L2	0.2	0.2	L1	L2	0.2	0.2	L2	L3	0.25	0.25	L2	L5	0	0
L2	L3	0.25	0.25	L2	L3	0.25	0.25	L2	L4	0.25	0.25	L3	L5	0	0
L2	L4	0.25	0.25	L2	L4	0.25	0.25	L2	L1	0.2	0.2	L4	L5	0	0
L2	L1	0.2	0.2	L2	L1	0.2	0.2	L2	L5	0	0	L5	L2	0	0
L2	L5	0	0	L2	L5	0	0	L3	L2	0.25	0.25	L5	L3	0	0
L3	L4	0.5	0.5	L3	L4	0.5	0.5	L3	L5	0	0	L5	L4	0	0
L3	L2	0.25	0.25	L3	L2	0.25	0.25	L4	L2	0.25	0.25				
L3	L5	0	0	L3	L5	0	0	L4	L5	0	0				
L4	L3	0.5	0.5	L4	L3	0.5	0.5	L5	L1	0.25	0.25				
L4	L1	0.4	0.4	L4	L1	0.4	0.4	L5	L2	0	0				
L4	L2	0.25	0.25	L4	L2	0.25	0.25	L5	L3	0	0				
L4	L5	0	0	L4	L5	0	0	L5	L4	0	0				
L5	L1	0.25	0.25	L5	L1	0.25	0.25								
L5	L2	0	0	L5	L2	0	0								
L5	L3	0	0	L5	L3	0	0								
L5	L4	0	0	L5	L4	0	0								

Esparsificação por threshold significa retirar arestas que estejam acima ou abaixo de

um determinado valor. Uma opção aqui é remover todos os valores de similaridades de jaccard iguais a zero e menores que 0.01. Outra opção é calcular um percentual de corte considerando os valores máximos e mínimos existentes na Tabela de Similaridade, já que eles podem variar para cada dataset multirrótulo. Por exemplo, um dataset X pode ter o valor máximo de similaridade igual 0.8 enquanto um dataset Y pode ser 0.5. Usar o percentual então é o mais indicado para se manter a coerência do corte.

Outro detalhe que deve ser considerado para que as partições híbridas sejam geradas de acordo com o esperado, é que todos os rótulos devem permanecer como vértices no grafo, isto é, nenhum rótulo pode ser literalmente apagado do grafo. O threshold deve ser aplicado de forma a considerar que todos os rótulos existam no grafo e, para isto, um limite deve ser adicionado.

Portanto, para construir grafos usado a esparcificação do tipo Threshold para datasets multirrótulo, primeiro os self-loops devem ser retirados e, em seguida um cálculo de porcentagem que considera a existência de todos os rótulos no grafo como vértices deve ser efetuado. O threshold pode ser aplicado na Tabela de Similaridade iterativamente, considerando um valor decrescente de porcentagem a cada corte. No primeiro corte, $X\%$ das arestas são consideradas de acordo com o valor máximo de similaridade. A cada iteração, subtrai-se 10% desse valor de corte e verifica-se se todos os rótulos estão presentes. A partir do percentual em que os rótulos começam a ser cortados, a iteração para.

No exemplo que está se utilizando aqui, o valor máximo de Jaccard é 0.75, enquanto o valor mínimo é 0.0. Dessa forma calcula-se a porcentagem da seguitne forma: $maximo = (maximo * a)/100$, onde $a = 90$ e em cada iteração a é decrementado de 10. Os valores de corte (considerando 10% a cada iteração) serão 0.6750, 0.5400, 0.3780, 0.2268 e 0.1134. Na primeira iteração, arestas com pesos maiores que 0.6750 são retiradas, na segunda iteração são arestas com pesos maiores que 0.5400, e assim por diante. A Tabela 23 apresenta os cortes feitos com a esparcificação threshold para o dataset de exemplo.

4.3.2 Particionando o espaço de rótulos

Nesta etapa, métodos de detecção de comunidade podem ser aplicados em cada grafo para particionar o espaço de rótulos. Os métodos são divididos em duas categorias: a) Hierárquicos: Walktrap (PONS; LATAPY, 2005), Fast Greedy (CLAUSET; NEWMAN; MOORE, 2004) e Edge Betweenness (NEWMAN; GIRVAN, 2004); e b) Não Hierárquicos: Louvain (BLONDEL et al., 2008), InfoMap (M, 2009), SpinGlass (REICHARDT, 2006) e Label Propagation (RAGHAVAN; ALBERT; KUMARA, 2007).

Os métodos hierárquicos fornecem dendrogramas que podem ser usados para construir várias partições híbridas. O procedimento é semelhante ao do algoritmo de agrupamento hierárquico, mas utiliza diferentes técnicas para aglomerar os vértices. Ao contrário dos métodos Hierárquicos, os Não Hierárquicos fornecem apenas uma partição híbrida, cons-

truída usando diferentes conceitos. Cada um desses métodos são explicados resumidamente a seguir. A Figura 43 ilustra os Passos 2 e 3 do HPML.C

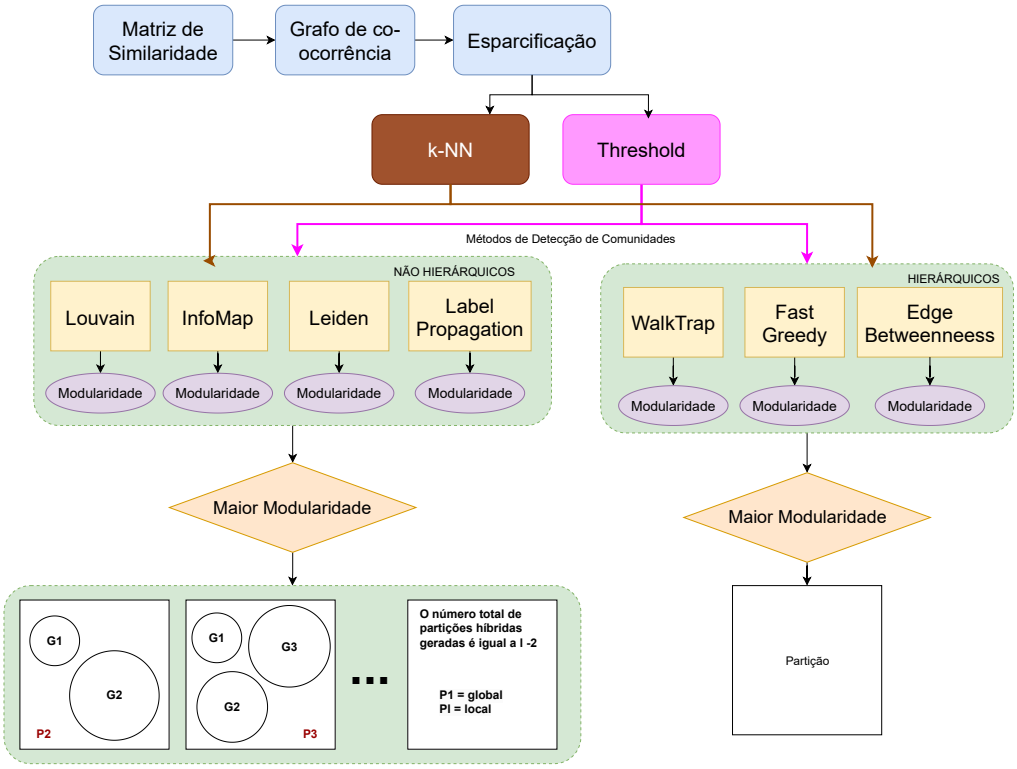


Figura 43 – Passos 2 e 3 Ilustrados da abordagem HPML.C: modelagem das correlações e particionamento do espaço de rótulos.

4.3.2.1 Edge Betweenness

Edge Betweenness significa aresta intermediária e, portanto, é um método de detecção de estrutura de comunidade baseada em arestas intermediárias. Muitas redes consistem em módulos (grupos/comunidades) densamente conectados, mas escassamente conectados a outros módulos. Edge Beteweenness mede o número de caminhos mais curtos através da aresta (NEWMAN; GIRVAN, 2004; JAVED et al., 2018; MITTAL; BHATIA, 2020). O método usa técnicas de algoritmos de agrupamento hierárquicos divisivos para descobrir divisões naturais baseadas em similaridades ou força de conexão entre os vértices. O algoritmo primeiro calcula a pontuação da aresta intermediária para todas as arestas da rede, depois encontra a aresta com o maior pontuação e remove-a da rede e, por fim, recalcula a pontuação da aresta intermediária para as arestas que ficaram. O processo é repetido até convergir.

É provável que as arestas que conectam módulos separados tenham alta intermediação de arestas, já que todos os caminhos mais curtos de um módulo a outro devem passar por elas. Portanto, se a aresta for removida gradualmente com a pontuação de intermediação

mais alta, um dendrograma é produzido. Cortando este dendrograma é possível encontrar diferentes partições dos dados e, consequementente as partições híbridas.

4.3.2.2 Fast Greedy

Fast Greedy é uma versão melhorada do algoritmo proposto por (NEWMAN, 2004). Os autores em (CLAUSET; NEWMAN; MOORE, 2004) propuseram uma mudança na estruturas de dados usadas na implementação do algoritmo de forma que ele é executado mais rapidamente e descarta informação não relevante no cálculo da modularidade. Comunidades que não têm arestas conectadas entre elas não contribuem para o aumento da modularidade, portanto, são desconsideradas no momento da fusão de comunidades nas iterações do algoritmo.

O algoritmo base encontra mudanças na modularidade que resultam na fusão de cada par de comunidades, escolhendo o maior entre elas e executando a fusão. No entanto, no algoritmo original de (NEWMAN, 2004) isto é feito em toda a Tabela de Similaridade, enquanto que no algoritmo de (CLAUSET; NEWMAN; MOORE, 2004) isto só é aplicado em pares de vértices que são fundidos por uma ou mais arestas. Como muitas matrizes são esparsas, então muito cálculo desnecessário é descartado e, tanto memória, quanto processamento são otimizados (JAVED et al., 2018).

O algoritmo funciona da mesma forma que o algoritmo de agrupamento hierárquico, começanco com n comunidades e convergindo ao final para uma única comunidade. Nesse processo, são calculados os valores iniciais de modularidade - que são armazenados em uma matriz - para cada par de vértice e uma pilha é populada com o maior elemento de cada linha da matriz de modularidade. Depois, o par com a maior modularidade da pilha é escolhido para se fundir com as comunidades correspondentes, as matrizes são atualizadas e a modularidade incrementada (CLAUSET; NEWMAN; MOORE, 2004; JAVED et al., 2018).

4.3.2.3 WalkTrap

O algoritmo Waltrap é baseado em random walks, isto é, em caminhadas aleatórias ou passeios aleatórios, o qual consiste de uma sucessão de passos aleatórios. No contexto de grafos, o processo do random walk indica que em cada passo o caminhante está em um vértice e aleatoriamente se move para outro vértice vizinho (JAVED et al., 2018; HUANG et al., 2021).

De acordo com Pons e Latapy (2005), passeios aleatórios tendem a ficar presos em partes densamente conectadas do grafo, dai o nome walktrap. Portanto, caminhos aleatórios de curta distância tendem a permanecer na mesma comunidade. O algoritmo também é baseado em algoritmos de agrupamento aglomerativo hierárquico, começando com n comunidades e em seguida calculando todas as distâncias entre todos os vértices adjacentes. O grupo evolui repetindo as operações a seguir em cada passo: a) Escolha 2 comunidades

(grupos/módulos) de acordo com o critério baseado na distância entre as comunidades; b) Junta as 2 comunidades em uma nova e criar uma nova partição; c) Atualize as distâncias entre as comunidades. O algoritmo é finalizado depois de $n - 1$ passos.

4.3.2.4 Louvain

O algoritmo é baseado no conceito de modularidade, encontra comunidades hierárquicas e é composto por duas fases que são repetidas iterativamente. É um algoritmo simples que consegue encontrar grupos rapidamente usando a alta modularidade da rede. A modularidade (Equação 48 e Equação 49) mede a qualidade da comunidade ou quão separados estão os diferentes tipos de vértices uns dos outros. Portanto, quantifica a densidade de links dentro das comunidades comparado com links entre comunidades (SILVA; ZHAO, 2016; MITTAL; BHATIA, 2020). Dentro de um grupo, os nós são altamente conectados em comparação com os outros grupos. Quanto maior o valor da modularidade melhor.

$$\mathcal{Q} = \sum q(e_{qp} - a^2) \tag{48}$$

$$a_q = \sum y e_{qp} \tag{49}$$

onde e_{qp} é a quantidade de ligações de um grafo que conecta os nós em uma comunidade q para os nós de uma comunidade p. Dessa forma, na fase 1 cada vértice começa em uma comunidade diferente, portanto, cada vértice é uma comunidade. Para cada vértice, são considerados os j vértices vizinhos do vértice i. O ganho da modularidade é avaliado quando o vértice i é removido de sua comunidade e colocado em outra comunidade.

O vértice i é então fixado na comunidade para a qual o ganho da modularidade é máximo e positivo. Se não há ganho positivo, i permance na comunidade original. Esse processo se repete e só pára quando um máximo local da modularidade é alcançado, ou seja, quando nenhum movimento de vértice é capaz de maximizar a modularidade. A fase 2 consiste em construir a rede cujos nós são as comunidades encontradas na primeira fase.

4.3.2.5 SpinGlass

Spinglass é baseado no Potts(WU, 1982) Spin Glass (DOMANY, 1999), um conceito da física e da mecânica estatística. Reichardt e Bornholdt (2004) e Reichardt (2006) mostraram que detecção de comunidades podem ser mapeadas de forma a encontrar o estado fundamental de um *Potts de Spin Glass* de alcance infinito a partir de um parâmetro simples e geral, que é válido para redes ponderadas e redes direcionadas. O método também foi motivado pela ideia de agrupamento superparamagnético (JAVED et al., 2018).

A energia do sistema de spin é equivalente à função de qualidade do agrupamento com os estados de spin sendo os índices de grupo, correspondendo a um método de particionamento de dados onde o número de agrupamentos é determinado automaticamente

pelo algoritmo como o número de estados de spin ocupados. Um único parâmetro relaciona o peso dado às ligações ausentes e existentes na função de qualidade e permite uma avaliação de estruturas de comunidade hierárquicas e sobrepostas.

Ao usar modelos de rotação para agrupamento, as medidas de similaridaes são traduzidas em forças e propriedades dinâmicas tais como correlações spin-spin que são medidas ou energias interpretadas como funções de qualidade. Assim, os métodos de detecção de comunidade podem ser mapeados para encontrar o estado verdadeiro de um spin glass de Potts de faixa infinita via o princípio de anatz e combinando informações ausentes e presentes (REICHARDT; BORNHOLDT, 2004; REICHARDT, 2006; JAVED et al., 2018).

4.3.2.6 Label Propagation

De acordo Raghavan, Albert e Kumara (2007), no algoritmo do Label Propagation cada nó é inicializado com um rótulo único e, a cada passo, cada nó adota o rótulo que a maioria de seus vizinhos possui. Neste processo iterativo, grupos de nós densamente conectados formam um consenso sobre um rótulo único para formar comunidades.

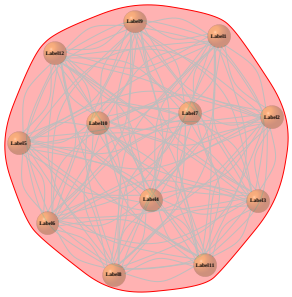
Este algoritmo simula a difusão do fluxo de rede através da difusão de rótulos. No grafo, cada vértice é associado a uma único rótulo. Depois o rótulo de cada vértice é atualizado iterativamente com o rótulo majoritário associado aos vizinhos, sendo a atualização aleatória. O algoritmo pára quando todos os rótulos dos vértices são consistentes com o rótulo mais frequente na vizinhança (RAGHAVAN; ALBERT; KUMARA, 2007; HUANG et al., 2021).

O algoritmo começa inicializando s rótulos em todos os nós da rede e em seguida organizando os nós da rede em uma ordem aleatória. Para cada nó escolhido nessa ordem específica, retorna o rótulo que ocorre com a frequência mais alta entre os vizinhos e quebra as ligações uniforme e aleatória. Se cada nó tiver um rótulo que indica o número máximo de seus vizinhos, o algoritmo para, caso contrário, continue até convergir.

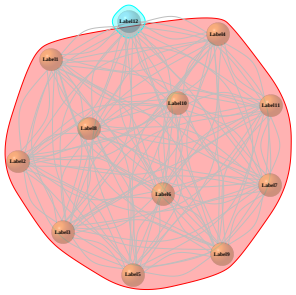
4.3.2.7 Info Map

O método InfoMap minimiza o tamanho da descrição de uma caminhada aleatória. Ao comprimir a descrição da rede, comunidades que refletem a dinâmica da rede podem ser encontradas. Rosvall, Axelsson e Bergstrom (2009) demonstraram que encontrar comunidades é equivalente a resolver um problema de código e, para isto, usaram o conceito de mapas para descrever a dinâmica entre as arestas e os nós das redes.

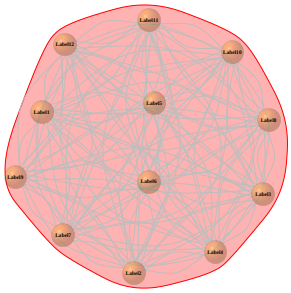
A trajetória de uma caminhada aleatória é descrita de tal forma que estruturas importantes têm nomes únicos e para isto usaram Huffman code. Dois níveis de descrição são usados para a caminhada aleatória: a grande maioria dos grupos recebe um nome único, mas esses nomes podem ser reutilizados nos nós dentro dos grupos. Esse processo resulta num mapa simplificado que destaca as regularidades da estrutura e seus relacionamentos



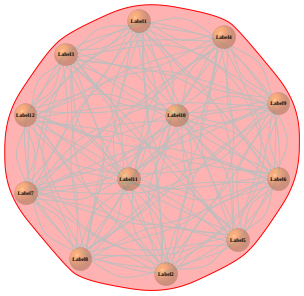
(a) Edge Betweenneess



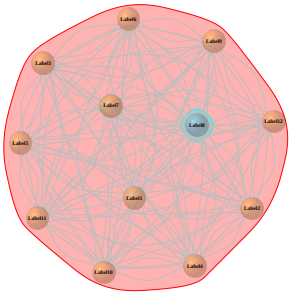
(b) Fast Greedy



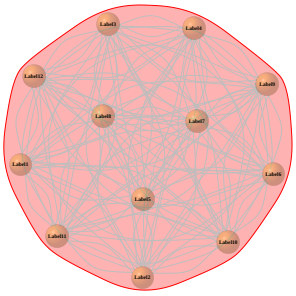
(c) WalkTrap



(d) Louvain



(e) SpinGlass



(f) Label Propagation

Figura 44 – Comunidades encontradas para o dataset PlantGO. Fonte: Elaborado pela autora.

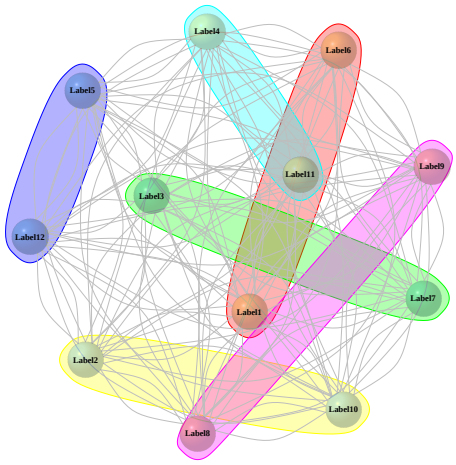


Figura 45 – Comunidades encontradas para o dataset PlantGO com InfoMap. Fonte: Elaborado pela autora.

(ROSVALL; AXELSSON; BERGSTROM, 2009; MITTAL; BHATIA, 2020; JAVED et al., 2018; HUANG et al., 2021).

4.3.2.8 Exemplos

O dataset PlantGO sem nenhum tipo de esparcificação foi usado para encontrar as comunidades apresentadas na Figura 44 e assim ilustrar cada um dos métodos apresentados nesta subseção. As partições encontradas para os métodos Edge Betweenness (Figura 44(a)), Fast Greedy (Figura 44(b)) e WalkTrap (Figura 44(c)) são apresentas nas Tabelas 24, 25 e 26 respectivamente. Como o dataset PlantGO possui 12 rótulos então 12 partições são encontradas, incluindo as global (P1) e local (P12). Os números dentro das Tabelas 24, 25 e 26 indicam o grupo ao qual o rótulo pertence. O número máximo de grupos possível é 12. Dessa forma P2 = 2 grupos, P3 = 3 grupos e assim por diante. Observa-se a diferença da distribuição dos rótulos nos grupos em cada partição. Edge Betweenness e Fast Greedy parecem ser complementares, enquanto WalkTrap apresenta uma distribuição diferente de ambos.

4.4 HPML versão D

O HPML.D é uma versão extendida do HPML.A que aplica conceitos de encadeamento do Classifier Chains (CC) e do Ensemble of Classifier Chains (ECC). A ideia base do CC é dividir o problema multirrótulo em subproblemas binários onde os n^3 classificadores são

[3] o número de classificadores é igual ao número de rótulos

Tabela 24 – Partições PlantGO com Edge Betweenneess

Rótulo	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
Label1	1	2	2	2	2	2	2	2	2	2	2	1
Label2	1	1	3	3	3	3	3	3	3	3	3	2
Label3	1	1	1	4	4	4	4	4	4	4	4	3
Label4	1	1	1	1	5	5	5	5	5	5	5	4
Label5	1	1	1	1	1	6	6	6	6	6	6	5
Label6	1	1	1	1	1	1	7	7	7	7	7	6
Label7	1	1	1	1	1	1	1	8	8	8	8	7
Label8	1	1	1	1	1	1	1	1	9	9	9	8
Label9	1	1	1	1	1	1	1	1	1	10	10	9
Label10	1	1	1	1	1	1	1	1	1	1	11	10
Label11	1	1	1	1	1	1	1	1	1	1	1	11
Label12	1	1	1	1	1	1	1	1	1	1	1	12

Tabela 25 – Partições PlantGO com Fast Greedy

Rótulo	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
Label1	1	1	1	1	1	1	1	1	1	1	1	1
Label2	1	1	1	1	1	1	1	1	1	1	1	2
Label3	1	1	1	1	1	1	1	1	1	1	2	3
Label4	1	1	1	1	1	1	1	1	1	2	3	4
Label5	1	1	1	1	1	1	1	1	2	3	4	5
Label6	1	1	1	1	1	1	1	2	3	4	5	6
Label7	1	1	1	1	1	1	2	3	4	5	6	7
Label8	1	1	1	1	1	2	3	4	5	6	7	8
Label9	1	1	1	1	2	3	4	5	6	7	8	9
Label10	1	1	1	2	3	4	5	6	7	8	9	10
Label11	1	1	2	3	4	5	6	7	8	9	10	11
Label12	1	2	3	4	5	6	7	8	9	10	11	12

Tabela 26 – Partições PlantGO com WalkTrap

Rótulo	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
Label1	1	1	1	3	2	1	5	4	4	3	2	1
Label2	1	1	1	1	5	5	6	5	5	4	3	2
Label3	1	1	3	4	4	3	3	2	1	5	4	3
Label4	1	2	2	2	1	6	7	6	6	6	5	4
Label5	1	1	1	3	2	1	2	1	7	7	6	5
Label6	1	1	3	4	4	3	3	2	1	8	7	6
Label7	1	1	1	1	3	2	1	7	8	9	8	7
Label8	1	1	1	1	3	2	1	8	9	10	9	8
Label9	1	1	1	3	2	1	2	1	2	1	10	9
Label10	1	2	2	2	1	4	4	3	3	2	1	10
Label11	1	2	2	2	1	4	4	3	3	2	1	11
Label12	1	1	1	3	2	1	2	1	2	1	11	12

conectados da seguinte forma: na fase de treinamento, os rótulos disponíveis do conjunto de dados são usados como novos atributos de entrada nos classificadores seguintes na cadeia, enquanto que na fase de teste, os rótulos preditos se tornam novos atributos. Dessa forma, os classificadores se conectam em cadeiras e possibilitam o aprendizado das correlações (READ et al., 2009; READ; PFAHRINGER, 2021; MISHRA; SINGH, 2022).

O Ensemble of Classifier Chains (ECC) é considerado um método estado-da-arte e considera correlações entre rótulos, sendo capaz de alcançar bom desempenho geral em vários datasets e medidas de avaliação (READ et al., 2009; READ; PFAHRINGER, 2021; MISHRA; SINGH, 2022). Um dos desafios ao trabalhar com ECC é a alta dimensionalidade do espaço de rótulos, que pode impor limitações para cadeias totalmente conectadas, à medida que a complexidade aumenta em termos de expansão do espaço de atributos. O HPML.D tenta melhorar as cadeias de classificadores ao encadear agrupamentos de rótulos correlacionados. Com base nessas premissas, as seguintes versões do HPLM.D

foram elaboradas:

- $HPML.D_{PADRAO}$: a versão original do método baseada no HPML.A;
- $HPML.D_{CI}$: uma versão do HPML onde o encadeamento é apenas interno, isto é, apenas os rótulos dentro de cada grupo são encadeados (c=chains, i=internal);
- $HPML.D_{CE}$: uma versão do HPML onde o encadeamento é apenas externo, isto é, apenas os grupos são encadeados (c=chains, e=external);
- $HPML.D_{CEI}$: uma versão do HPML onde é realizado tanto o encadeamento interno quanto o externo. Esta é uma versão conjunta do $HPML.D_{CI}$ e $HPML.D_{CE}$. Portanto, há o encadeamento dos rótulos e dos grupos.

Em todas as versões do HPML.D as correlações foram modeladas usando a medida de similaridade Jaccard. Para escolher a melhor métrica de ligação, foi executado um experimento com 70 datasets multirrótulo, onde foram gerados dendrogramas para as 6 métricas de ligação e calculado o coeficiente aglomerativo para cada uma delas. A métrica de ligação com o maior coeficiente foi escolhida para construir o dendrograma para cada dataset. A conclusão foi que, para todos os 70 datasets, Ward.D2 foi a métrica de ligação que obteve o melhor coeficiente aglomerativo, e portanto, os dendrogramas foram construídos usando esta métrica.

As partições híbridas são produzidas através do corte em níveis do dendrograma, exatamente como no HPML.A, o qual já foi explicado anteriormente. Para selecionar a melhor entre elas foi utilizado o coeficiente da silhueta onde a partição híbrida com a maior silhueta é escolhida para o teste.

Em todas as versões do HPML.D foram induzidas Florestas Aleatórias (Random Forests - RFs) e não as PCTs do CLUS como no HPML.A, HPML.B e HPML.C. As RFs são um método de ensemble que combinam várias árvores de decisão e usam o método de voto majoritário para decidir o rótulo final da predição. A mudança ocorreu devido aos seguintes fatores: 1) alto custo de execução do CLUS para datasets com mais de 100 rótulos nos servidores disponíveis. Apriori, não era esperado utilizar outro classificador, já que não fazia sentido induzir diferentes classificadores para realizar as comparações entre as diferentes versões do método HPML de forma justa, já que o que se queria era observar o comportamento do mesmo classificador em todas as versões definidas; 2) necessidade de testar o método com outro classificador, 3) consistência, 4) rapidez para grandes datasets, 5) considera as correlações entre rótulos, 6) é um método do estado da arte na literatura capaz de alcançar bons resultados preditivos sendo considerado um dos melhores métodos multirrótulo; 7) o HPML-D foi desenvolvido durante o doutorado sanduíche que tem duração de 6 meses, e por este motivo não poderia haver demora na execução dos experimentos e análise dos resultados (DENG et al., 2001; JOLY; GEURTS;

WEHENKEL, 2014; MOYANO et al., 2018; TYRALIS; PAPACHARALAMPOUS, 2019; ENDUT et al., 2022; BOGATINOVSKI et al., 2022)

Um dos desafios do ECC é como tratar a longa cadeia de rótulos que, a depender da dimensionalidade do espaço de rótulos, pode se tornar impraticável. De acordo com (MISHRA; SINGH, 2022) uma das formas de se resolver isto pode ser usando cadeias de tamanho limitado. Portanto, o HPML pode ser uma forma de se resolver este problema já que o espaço de rótulos é quebrado em grupos disjuntos de rótulos correlacionados, diminuindo assim o número de rótulos total que o ECC deve lidar.

No HPML.D, RFs são induzidas em cada grupo e nenhum tipo de encadeamento é utilizado. A hipótese é que se datasets grandes (mais de 100 rótulos) podem obter um resultado melhor no HPML.D$_{PADRAO}$ que o ECC, então, pode-se afirmar que aprender grupos disjuntos de rótulos correlacionados sem encadeamento nenhum é melhor que aprender todos o rótulos juntos em um conjunto de classificadores em cadeia. A Figura 46 ilustra o HPML.D$_{PADRAO}$.

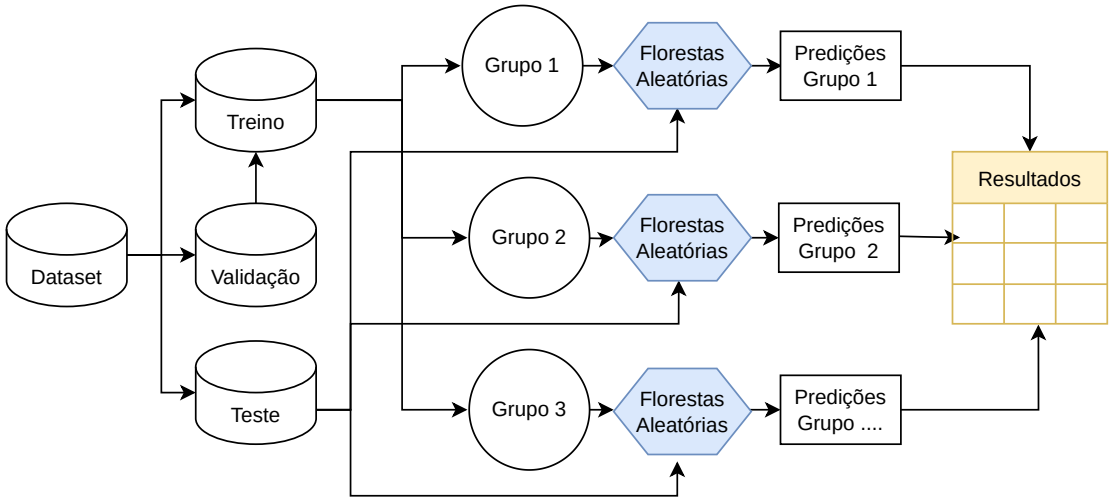


Figura 46 – Abstração da metodologia do HPML.D$_{PADRAO}$

No HPML.D$_{CI}$, o espaço de rótulos continua a ser quebrado em espaços menores, mas o ECC é utilizado em cada grupo e não um classificador multirrótulo tradicional. A pergunta que motiva este método é: o ECC induzido em cada grupo é capaz de aprender as relações entre os rótulos desses grupos e ser melhor que o ECC para todos os rótulos? É a mesma premissa da comparação com as partições globais e locais quando se induzia o CLUS no HPML.A. Como o ECC transmite as correlações aprendidas na cadeia, então a possibilidade delas realmente serem aprendidas provavelmente seriam maiores. A literatura mostra que o ECC é capaz de superar métodos da abordagem local (GUEHRIA, 2023; MOYANO et al., 2018), portanto, supõe-se que aprendendo grupos de rótulos correlacionados isto também poderia ocorrer. A Figura 47 ilustra o HPML.D$_{CI}$.

No HPML.D$_{CE}$ os grupos são encadeados na ordem em que foram criados no dendrograma. Apesar de trabalhos na literatura apontarem que o ideal é usar uma or-

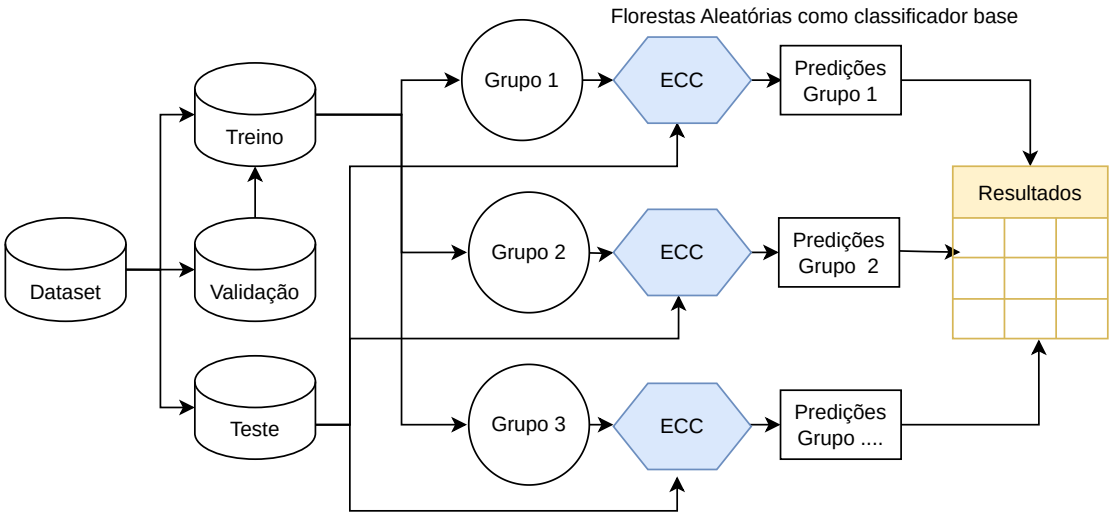


Figura 47 – Abstração da metodologia do HPML.D$_{CI}$

dem aleatória na cadeia, outros afirmam que é possível utilizar uma ordem diferente da aleatória, no caso uma ordem baseada nas correlações (READ; PFAHRINGER, 2021; MORAL-GARCíA et al., 2022; MISHRA; SINGH, 2022) . As hipóteses levantadas para este método são: ao agrupar os rótulos correlacionados, a longa cadeia de conjuntos de dados com alta dimensionalidade do espaço de rótulos é quebrada e isso pode levar a um bom aprendizado; ii) é possível aumentar o poder de predição do classificador aprendendo os grupos disjuntos de rótulos correlacionados em uma ordem de encadeamento com base nas correlações, superando o ECC.

O encadeamento dos grupos no HPML.D$_{CE}$ ocorre da seguinte forma: i) na fase de treinamento, os rótulos disponíveis são usados como novos atributos nos grupos seguintes; ii) na fase de teste, os rótulos preditos são usados como novos atributos nos grupos seguintes. HPML.D$_{CE}$ é uma cadeia de uma única direção - o último grupo não é encadeado com o primeiro. Em cada grupo, RFs foram induzidas e as predições combinadas para fazer a avaliação final. A Figura 48 ilustra a fase de treinamento do HPML.D$_{CE}$, enquanto que a Figura 49 ilustra a fase de teste para uma partição híbrida ilustrativa.

Na última versão do HPML.D, o encadeamento ocorre tanto externa quanto internamente, e é denominado HPML.D$_{CEI}$. Neste método os grupos disjuntos de rótulos correlacionados são aprendidos com o ECC e os rótulos/predições dos grupos anteriores são passados para os grupos seguintes. Dessa forma, as informações de correlações entre rótulos são transferidas mais de uma vez, começando em um nível local de correlações (encadeamento interno) e terminando em um nível global (encadeamento externo).

Este método começa com o primeiro grupo gerado, passa os rótulos/predições para todos os outros grupos que fazem parte da partição, e termina com o último grupo contendo todos os rótulos dos grupos anteriores no espaço de atributos de entrada - de alguma forma é similar ao agrupamento hierárquico. Novamente a cadeia é de uma única direção

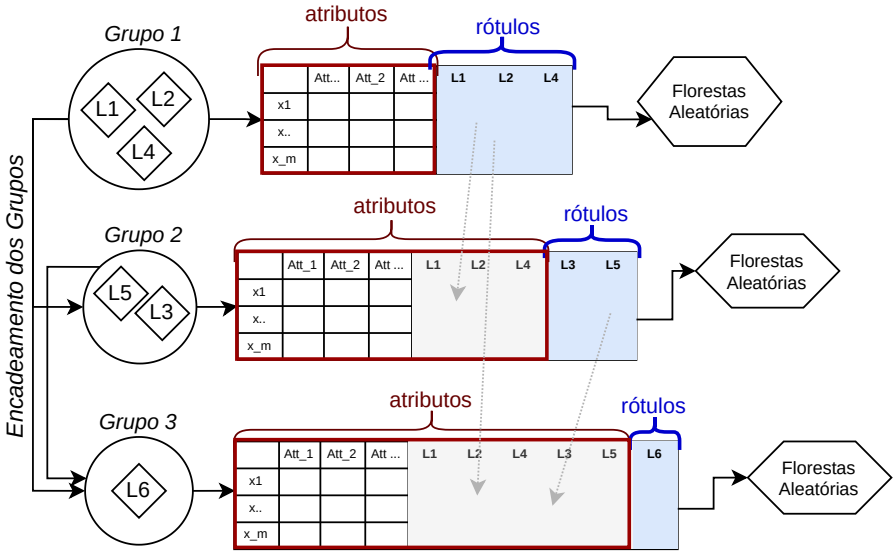


Figura 48 – Fase de treinamento do HPML.D$_{CE}$ com uma partição ilustrativa

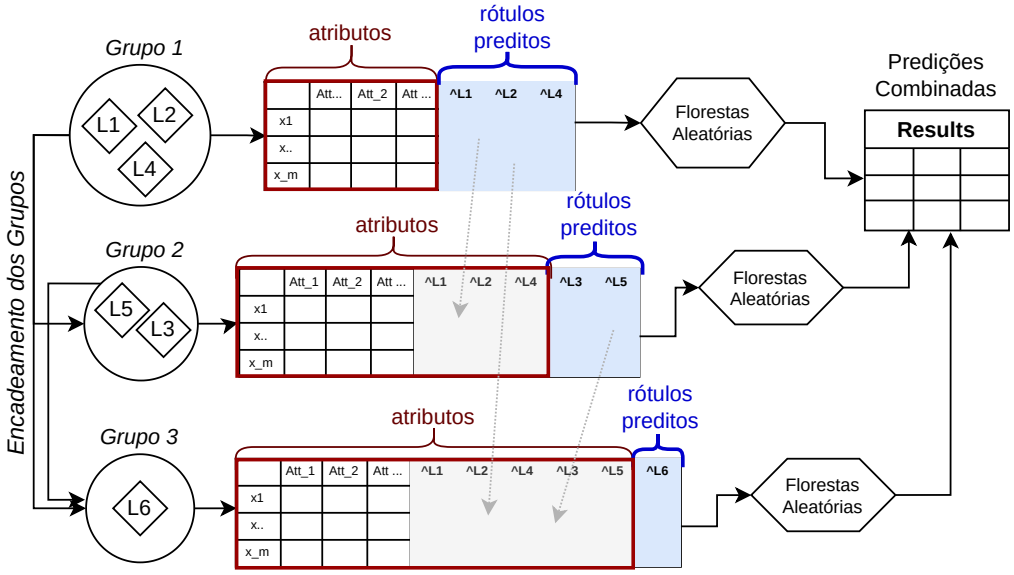


Figura 49 – Fase de teste do HPML.D$_{CE}$ com uma partição ilustrativa

(top-down). As hipóteses levantadas para o HPML.D$_{CI}$ e HPML.D$_{CE}$ são herdadas por este método e uma outra questão é levantada: aprender vários níveis de correlações é melhor do que um ou nenhum?

As Figuras 50 e 51 ilustram o processo de treinamento e teste do HPML.D$_{CEI}$ em uma partição híbrida ilustrativa. Concluindo o capítulo, supõe-se que as três versões do HPML tenham desempenho superior que o ECC pois quebram a cadeia de classificadores em cadeias menores e exploram as correlações entre rótulos em níveis diferentes do ECC.

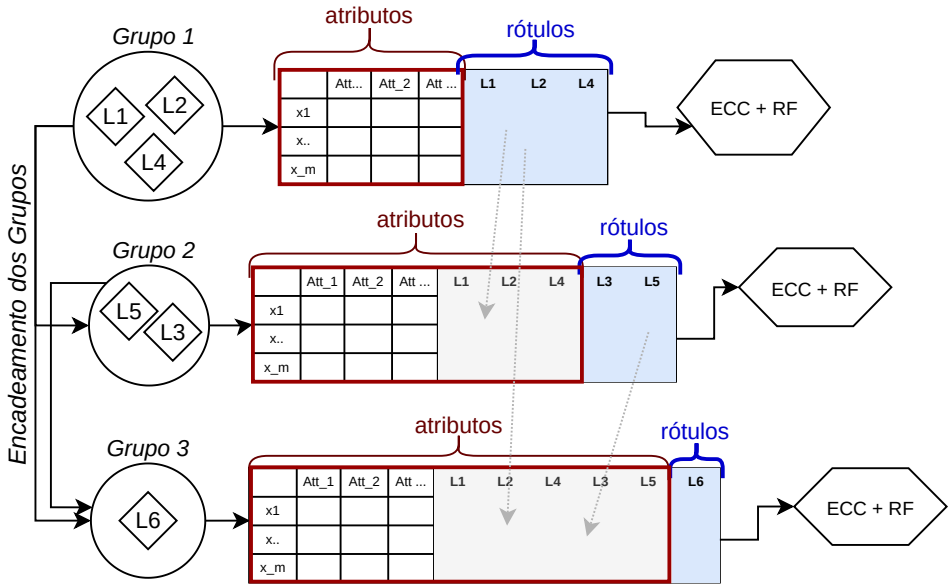


Figura 50 – Fase de treinamento do HPML.D$_{CEI}$ com uma partição ilustrativa

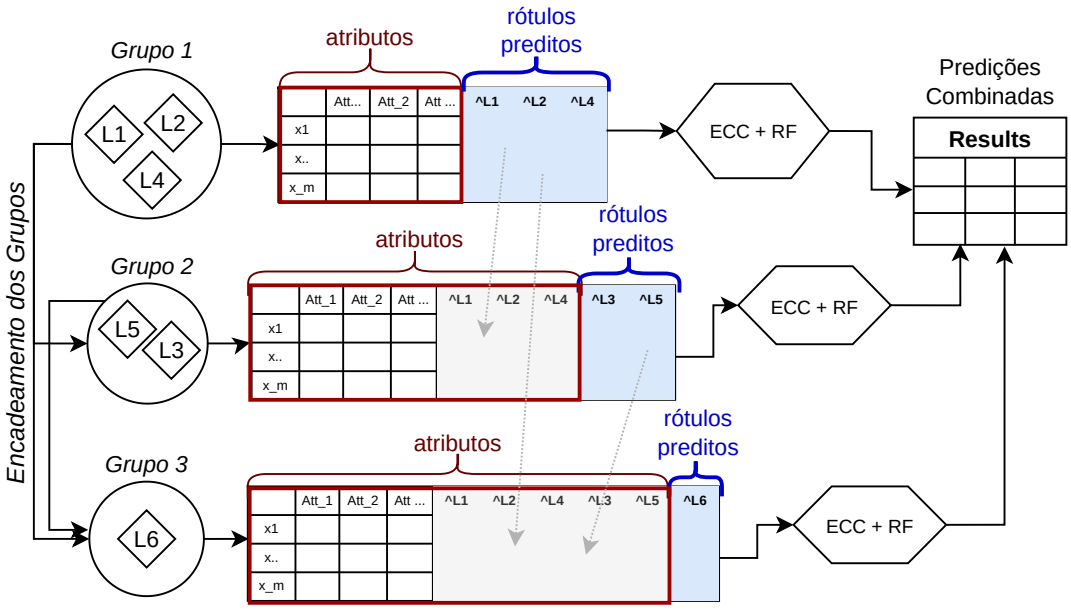


Figura 51 – Fase de teste do HPML.D$_{CEI}$ com uma partição ilustrativa

Capítulo 5

Configuração dos Experimentos

Neste capítulo, são apresentados os conjuntos de dados multirrótulo utilizados nos experimentos (Seção 5.1), os métodos base usados nas comparações experimentais (Seção 5.3) e (Seção 5.4), e uma visão geral de todos os métodos propostos e investigados nesta pesquisa (Seção 5.7, Seção 5.5 e Seção 5.6). Foram utilizadas as linguagem R e Python, a IDE RStudio e outros algoritmos desenvolvidos em Java. Os ambientes para execução dos experimentos foram montados usando Conda e Singularity com suporte para Ubuntu, R, Java, Python e Rclone[1] para transferência de arquivos. O código-fonte de todos os métodos aqui desenvolvidos, todos os materiais necessários para replicar os experimentos e os resultados completos estão disponíveis publica e gratuitamente no GitHub[2].

5.1 Conjuntos de Dados Multirrótulo

Foram utilizados 70 conjuntos de dados multirrótulo de cinco domínios de aplicação diferentes, os quais estão disponíveis gratuitamente para download nos seguintes repositórios: <https://cometa.ujaen.es/> e <https://www.uco.es/kdis/mllresources/#3sourcesDesc>. Os datasets cellcycle, church, derisi, eisen, gasch1, expr, pheno e seq podem ser baixados no repositório desta tese. Os conjuntos de dados são apresentados de forma separada para cada um dos experimentos conduzidos.

Para o experimento HPML.A$_C$, por ter sido o primeiro, apenas os conjuntos de dados multirrótulo mais conhecidos de bencharmk foram escolhidos. No segundo experimento, Exaustivo-Oráculo, apenas conjuntos com menos de 8 rótulos no espaço de rótulos foram

[1] <https://ubuntu.com/>, <https://www.r-project.org/>, <https://posit.co/>, <https://openjdk.org/>, <http://scikit.ml/>, <https://www.python.org/>, <https://rclone.org/>

[2] <https://github.com/cissagatto/HPML>

selecionados, pois é impraticável validar e testar todas as partições possíveis para um conjunto de dados com 8 ou mais rótulos. No terceiro experimento, o de Comunidades, optou-se por testar a hipótese com conjuntos mais variados, incluindo alguns com muito mais rótulos dos ques até então haviam sido utilizados. Por fim, no experimento de encadeamento, como a ideia é quebrar a longa cadeia de classificadores, então faria mais sentido experimentar apenas com conjuntos com muitos rótulos, no caso, mais de 100.

Portanto, as Tabelas 27, 28 e 29 resumem as características de cada um dos conjuntos de dados conforme descrição a seguir:

- Domain: domínio de aplicação de cada conjunto de dados;
- m: número total de instâncias (linhas);
- d: número total de atributos da instância;
- X: número total de atributos de entrada;
- L: número total de rótulos no espaço de rótulos (atributos de saída);
- Labelsets: conjuntos de rótulos (ou combinações de rótulos);
- Single: rótulos únicos;
- Max.Freq.: proporção de instâncias associadas aos conjuntos de rótulos que ocorrem com mais frequência
- Card: média de rótulos por instância;
- Dens: frequência média de rótulos;
- Mean-IR: nível de desbalanceamento médio;
- Scumble: nível de concordância entre rótulos frequentes e pouco frequentes;
- ULD: nível de dependência entre rótulos[3]
- TCS: nível de dificuldade de aprendizado para um modelo preditivo;
- GridN: total de neurônios utilizados para gerar o mapa bidimensional de kohonen;
- Max.Nei: número máximo de vizinhos que pode ser usado no k-NN.

[3] O cálculo desta medida foi feito para cada um dos 10 folds da validação cruzada. Os valores reportados na tabela são a média dos 10 folds

Tabela 27 – Conjuntos de Dados Multirrótulo - Parte 1

EXPERIMENTO 1: HPML.A$_C$

Nome	Dom.	m	d	X	L	LabelSets	Single	Max.Freq.	Card.	Dens.	Mean-IR	Scumble	TCS	ULD	GridN	Max.Nei
birds	Áudio	645	279	260	19	133	73	294	1,0140	0,0534	5,4070	0,0330	13,3955	0,1122	25	18
emotions	Música	593	78	72	6	27	4	81	1,8685	0,3114	1,4781	0,0110	9,3643	0,2784	9	5
EukaryotePseAAC	Biologia	7766	462	440	22	112	37	1580	1,1456	0,0521	45,0117	0,0174	13,8963	0,0426	9	21
flags	Imagem	194	26	19	7	54	24	27	3,3918	0,4845	2,2547	0,0606	8,8793	0,1572	16	6
GpositiveGO	Biologia	519	916	912	4	7	2	206	1,0077	0,2519	3,8605	0,0010	10,1478	0,3392	4	3
PlantGO	Biologia	978	3103	3091	12	32	8	277	1,0787	0,0899	6,6904	0,0058	13,9869	0,0672	9	11
scene	Imagem	2407	300	294	6	15	3	405	1,0740	0,1790	1,2538	0,0003	10,1834	0,1139	4	5
VirusGO	Biologia	207	755	749	6	17	6	56	1,2174	0,2029	4,0412	0,0079	11,2437	0,1429	4	5
yeast	Biologia	2417	117	103	14	198	77	237	4,2371	0,3026	7,1968	0,1044	12,5621	0,2541	25	13
Yelp	Texto	10806	676	671	5	32	0	2120	1,6383	0,3277	2,8756	0,0332	11,5839	0,1033	9	4

EXPERIMENTO 2: EXAUSTIVO-ORÁCULO

Nome	Dom.	m	d	X	L	LabelSets	Single	Max.Freq.	Card.	Dens.	Mean-IR	Scumble	TCS	ULD	GridN	Max.Nei
emotions	Música	593	78	72	6	27	4	81	1,8685	0,3114	1,4781	0,0110	9,3643	0,2784	9	5
flags	Imagem	194	26	19	7	54	24	27	3,3918	0,4845	2,2547	0,0606	8,8793	0,1572	16	6
GpositiveGO	Biologia	519	916	912	4	7	2	206	1,0077	0,2519	3,8605	0,0010	10,1478	0,3392	4	3
GpositivePseAAC	Biologia	519	444	440	4	7	2	206	1,0077	0,2519	3,8605	0,0010	9,4190	0,3392	4	3
scene	Imagem	2407	300	294	6	15	3	405	1,0740	0,1790	1,2538	0,0003	10,1834	0,1139	4	5
VirusGO	Biologia	207	755	749	6	17	6	56	1,2174	0,2029	4,0412	0,0079	11,2437	0,1429	4	5
VirusPseAAC	Biologia	207	446	440	6	17	6	56	1,2174	0,2029	4,0412	0,0079	10,7117	0,1429	4	5
Yelp	Texto	10806	676	671	5	32	0	2120	1,6383	0,3277	2,8756	0,0332	11,5839	0,1033	9	4

EXPERIMENTO 3: COMUNIDADES

Nome	Dom.	m	d	X	L	LabelSets	Single	Max.Freq.	Card.	Dens.	Mean-IR	Scumble	TCS	ULD	GridN	Max.Nei
birds	Áudio	645	279	260	19	133	73	294	1,014	0,0534	5,407	0,0330	13,3955	0,1122	25	18
cal500	Música	502	242	68	174	502	502	1	26,044	0,1497	20,5778	0,3372	15,5972	0,1387	100	173
cellcycle	Biologia	3757	255	77	178	1423	1010	366	2,191	0,0123	99,0612	0,1270	16,7861			177
derisi	Biologia	3725	241	63	178	1416	1003	365	2,199	0,0124	99,1253	0,1276	16,5805			177
eisen	Biologia	2424	244	79	165	1014	728	150	2,341	0,0142	75,5898	0,1359	16,3971			164
emotions	Música	593	78	72	6	27	4	81	1,869	0,3114	1,4781	0,0110	9,3643	0,2784	9	5
EukaryotePseAAC	Biologia	7766	462	440	22	112	37	1580	1,146	0,0521	45,0117	0,0174	13,8963	0,0426	9	21
flags	Imagem	194	26	19	7	54	24	27	3,392	0,4845	2,2547	0,0606	8,8793	0,1572	16	6
gasch1	Biologia	3764	351	173	178	1421	1010	366	2,187	0,0123	98,8751	0,1267	17,5942			177
GnegativeGO	Biologia	1392	1725	1717	8	19	5	533	1,046	0,1307	18,4476	0,0096	12,4722	0,2770	9	7
GnegativePseAAC	Biologia	1392	448	440	8	19	5	533	1,046	0,1307	18,4476	0,0096	11,1107	0,2770	9	7
langlog	Texto	1460	1079	1004	75	304	189	207	1,180	0,0157	39,2669	0,0510	16,9463	0,2018	289	74
medical	Texto	978	1494	1449	45	94	33	155	1,245	0,0277	89,5014	0,0471	15,6286		9	44
pheno	Biologia	1591	234	69	165	784	599	158	2,243	0,0136	48,6674	0,1263	16,0045			164
PlantGO	Biologia	978	3103	3091	12	32	8	277	1,079	0,0899	6,6904	0,0058	13,9869	0,0672	9	11
scene	Imagem	2407	300	294	6	15	3	405	1,074	0,1790	1,2538	0,0003	10,1834	0,1139	4	5
seq	Biologia	3919	656	478	178	1434	1017	470	2,135	0,012	99,5157	0,1230	18,6196			177
VirusPseAAC	Biologia	207	446	440	6	17	6	56	1,217	0,2029	4,0412	0,0079	10,7117	0,1429	4	5
yeast	Biologia	2417	117	103	14	198	77	237	4,237	0,3026	7,1968	0,1044	12,5621	0,2541	25	13
Yelp	Texto	10806	676	671	5	32	0	2120	1,638	0,3277	2,8756	0,0332	11,5839	0,1033	9	4

Tabela 28 – Conjuntos de Dados Multirrótulo - Parte 2

EXPERIMENTO 4: ENCADEAMENTO																
Nome	**Dom.**	m	d	X	L	**LabelSets**	**Single**	**Max.Freq.**	**Card.**	**Dens.**	**Mean-IR**	**Scumble**	**TCS**	**ULD**	**GridN**	**Max.Nei**
bibtex	Texto	7395	1995	1836	159	2856	2199	471	2,4019	0,0151	12,4983	0,0938	20,5414	0,1473	196	158
cal500	Música	502	242	68	174	502	502	1	26,0438	0,1497	20,5778	0,3372	15,5972	0,1387	100	173
corel16k001	Imagem	13766	653	500	153	4803	3120	253	2,8587	0,0187	34,1552	0,2731	19,722	0,1355		152
corel16k003	Imagem	13760	654	500	154	4812	3069	258	2,8286	0,0184	37,058	0,285	19,7304	0,1261		153
corel16k004	Imagem	13837	662	500	162	4860	3112	250	2,842	0,0175	35,8989	0,2772	19,791	0,1376		161
corel16k005	Imagem	13847	660	500	160	5034	3293	252	2,8577	0,0179	34,9364	0,285	19,8138	0,13		159
corel16k006	Imagem	13859	662	500	162	5009	3239	176	2,8849	0,0178	33,3979	0,2905	19,8212	0,1377		161
corel16k007	Imagem	13915	674	500	174	5158	3389	254	2,8859	0,0166	37,7146	0,2821	19,922	0,1361		173
corel16k008	Imagem	13864	668	500	168	4956	3169	253	2,883	0,0172	36,2	0,2894	19,8469	0,1408	X	167
corel16k009	Imagem	13884	673	500	173	5175	3346	177	2,9301	0,0169	36,4456	0,2978	19,9195	0,1342		172
corel16k010	Imagem	13618	644	500	144	4692	2999	350	2,8153	0,0196	32,9985	0,2786	19,638	0,1281	196	143
rcv1sub1	Texto	6000	47337	47236	101	1028	657	246	2,8797	0,0285	54,4923	0,2237	22,3134	0,191	36	100
rcv1sub2	Texto	6000	47337	47236	101	954	589	549	2,6342	0,0261	45,5138	0,2092	22,2387	0,2236	49	100
sta_che	Texto	6961	715	540	175	3032	2331	318	2,1093	0,0121	56,8779	0,1867	19,4733	0,0526		174

5.2 Validação Cruzada de 10 Folds

Todos os experimentos foram conduzidos na forma de validação cruzada com 10-folds. Para cada conjunto de dados multirrótulo foram criados e salvos antecipadamente os 10-folds e exatamente esses arquivos foram usados em todos os experimentos. Dessa forma garante-se a reprodutibilidade dos experimentos e estes folds estão disponíveis para download no repositório oficial da tese (<https://github.com/cissagatto/HPML>).

No repositório há links que levam para os códigos específicos de cada uma das versões HPML implementadas. Portanto, os resultados reportados no Capítulo 6 são a média dos 10 folds. Além disso, também no repositório é possível encontrar links que direcionam para um sumário dos resultados onde constam também o desvio padrão e outras informações que, por motivos de espaço não foram colocados na tese.

Para realizar a divisão foi utilizada uma estratificação específica para multirrótulo disponível no pacote UTIML (<https://cran.r-project.org/web/packages/utiml/index.html>) da linguagem R. Aproximadamente 80% dos dados foram utilizados para os conjuntos de treino, 10% para teste e outros 10% para validação.

O código em R/Python foi desenvolvido para executar em paralelo nos servidores do BioMaL (<http://www.biomal.ufscar.br/>) e no cluster da UFSCar (<https://www.sin.ufscar.br/servicos/computacao-sob-demanda/cluster-ufscar>), dessa forma foi possível otimizar o tempo de execução e utilizar corretamente cada um dos folds gerados.

5.3 Partições Globais

As partições globais são geradas com base na abordagem multirrótulo global já explicada no Capítulo 2. A Figura 52 ilustra o processo de obtenção de partições globais. Como já mencionado, qualquer classificador multirrótulo pode ser utilizado. Neste trabalho optou-se por usar os classificadores CLUS e Florestas Aleatórias nas estratégias de particionamento.

Tabela 29 – Conjuntos de Dados Multirrótulo - Parte 3

EXPERIMENTO: MELHOR MÉTRICA DE LIGAÇÃO

Nome	Dom.	m	d	X	L	LabelSets	Single	Max.Freq.	Distinct	Card.	Dens.	Mean-IR	Scumble	TCS	ULD	GridN	Max.Nei
bibtex	Text	7395	1995	159	1836	2856	2199	471	2856	2,4019	0,0151	12,50	0,0938	20,54	0,1473	196	158
birds	Audio	645	279	19	260	133	73	294	133	0,1014	0,0534	5,41	0,0330	13,40	0,1122	25	18
cal500	Music	502	242	174	68	502	502	1	502	26,0438	0,1497	20,58	0,3372	15,60	0,1387	100	173
cellcycle	biology	3757	255	178	77	1423	1010	366		2,1908	0,0123	99,06	0,1270	16,79	0,0000		177
church	biology	3755	205	178	27	1421	1011	365		2,1883	0,0123	98,88	0,1271	15,74	0,0000		177
corel16k001	Image	13766	653	153	500	4803	3120	253		2,8587	0,0187	34,16	0,2731	19,72	0,1355		152
corel16k002	Image	13761	664	164	500	4868	3103	251		2,8824	0,0176	37,68	0,2883	19,80	0,1409		163
corel16k003	Image	13760	654	154	500	4812	3069	258		2,8286	0,0184	37,06	0,2850	19,73	0,1261		153
corel16k004	Image	13837	662	162	500	4860	3112	250		0,2842	0,0175	35,90	0,2772	19,79	0,1376		161
corel16k005	Image	13847	660	160	500	5034	3293	252		2,8577	0,0179	34,94	0,2850	19,81	0,1300		159
corel16k006	Image	13859	662	162	500	5009	3239	176		2,8849	0,0178	33,40	0,2905	19,82	0,1377		161
corel16k007	Image	13915	674	174	500	5158	3389	254		2,8859	0,0166	37,71	0,2821	19,92	0,1361		173
corel16k008	Image	13864	668	168	500	4956	3169	253		0,2883	0,0172	36,20	0,2894	19,85	0,1408		167
corel16k009	Image	13884	673	173	500	5175	3346	177		2,9301	0,0169	36,45	0,2978	19,92	0,1342		172
corel16k010	Image	13618	644	144	500	4692	2999	350		2,8153	0,0196	33,00	0,2786	19,64	0,1281	196	143
corel5k	Image	5000	873	374	499	3175	2523	55	3175	0,3522	0,0094	189,57	0,3941	20,20		121	373
derisi	biology	3725	241	178	63	1416	1003	365		2,1989	0,0124	99,13	0,1276	16,58			177
eisen	biology	2424	244	165	79	1014	728	150		2,3412	0,0142	75,59	0,1359	16,40			164
emotions	Music	593	78	6	72	27	4	81	58	1,8685	0,3114	1,48	0,1100	9,36	0,2784	9	5
enron	Text	1702	1054	53	1001	753	573	163		3,3784	0,0637	73,95	0,3028	17,50			52
EukaryoteGO	Biology	7766	462	22	440	112	37	1580		1,1456	0,0521	45,01	0,0174	13,90		9	21
EukaryotePseAAC	Biology	7766	462	22	440	112	37	1580		1,1456	0,0521	45,01	0,0174	13,90	0,0426	9	21
expr	biology	3779	729	178	551	1424	1010	368		2,1863	0,0123	99,43	0,1266	18,75	0,0000		177
flags	Image	194	26	7	19	54	24	27	54	3,3918	0,4845	2,25	0,0606	8,88	0,1572	16	6
foodtruck	Recommend,	407	33	12	21	116	74	115		2,2899	0,1908	7,09	0,1035	10,28			11
gasch1	biology	3764	351	178	173	1421	1010	366		2,1865	0,0123	98,88	0,1267	17,59			177
gasch2	biology	3779	230	178	52	1424	1010	368		2,1863	0,0123	99,43	0,1266	16,39			177
genbase	Biology	662	1213	27	1186	32	10	170	32	1,2523	0,0464	37,31	0,0288	13,84		9	26
GnegativeGO	Biology	1392	1725	8	1717	19	5	533		0,1046	0,1307	18,45	0,0096	12,47	0,2770	9	7
GnegativePseAAC	Biology	1392	448	8	440	19	5	533		0,1046	0,1307	18,45	0,0096	11,11	0,2770	9	7
GpositiveGO	Biology	519	916	4	912	7	2	206		1,0077	0,2519	3,86	1,0000	10,15	0,3392	4	3
GpositivePseAAC	Biology	519	444	4	440	7	2	206		1,0077	0,2519	3,86	1,0000	9,42	0,3392	4	3
HumanGO	Biology	3106	9858	14	9844	85	28	718		1,1851	0,0847	15,29	0,0203	16,28	0,0596		13
HumanPseAAC	Biology	3106	454	14	440	85	28	718		1,1851	0,0847	15,29	0,0203	13,17	0,0596		13
imdb	Image	120919	1029	28	1001	4503	2263	13144		1,9997	0,0714	25,12	0,1082	18,65	0,0323		27
langlog	Text	1460	1079	75	1004	304	189	207	1214	1,1801	0,0157	39,27	0,5100	16,95	0,2018	289	74
mediamill	Text	43907	221	101	120	6555	4104	2363	6555	4,3756	0,0433	256,40	0,3547	18,19	0,2175	144	100
medical	Text	978	1494	45	1449	94	33	155	94	1,2454	0,0277	89,50	0,0471	15,63		9	44
ng20	Text	19300	1026	20	1006	55	17	997		1,0289	0,0514	1,01	0,0000	13,92	0,0633		19
nuswide_BoW	Image	269648	582	81	501	18430	11378	60301	18430	1,8685	0,0231	95,12	0,1710	20,43		225	80
nuswide_VLAD	Image	269648	210	81	129	18430	11378	60301	18430	1,8685	0,0231	95,12	0,1710	19,08		289	80
ohsumed	Biology	13929	1025	23	1002	1147	578	1175		1,6631	0,0723	7,87	0,0688	17,09	0,0734		22
pheno	biology	1591	234	165	69	784	599	158		2,2432	0,0136	48,67	0,1263	16,00	0,0000		164
PlantGO	Biology	978	3103	12	3091	32	8	277		1,0787	0,0899	6,69	0,0058	13,99	0,0672	9	11
PlantPseAAC	Biology	978	452	12	440	32	8	277		1,0787	0,0899	6,69	0,0058	12,04	0,0672	9	11
rcv1sub1	Text	6000	47337	101	47236	1028	657	246	1028	2,8797	0,0285	54,49	0,2237	22,31	0,1910	36	100
rcv1sub2	Text	6000	47337	101	47236	954	589	549	954	2,6342	0,0261	45,51	0,2092	22,24	0,2236	49	100
rcv1sub3	Text	6000	47337	101	47236	939	591	635	939	2,6142	0,0259	68,33	0,2075	22,22		49	100
rcv1sub4	Text	6000	47330	101	47229	816	491	950	816	2,4837	0,0246	89,37	0,2165	22,08		64	100
rcv1sub5	Text	6000	47336	101	47235	946	586	526	946	2,6415	0,0262	69,68	0,2381	22,23		49	100
scene	Image	2407	300	6	294	15	3	405	15	0,1074	0,1790	1,25	0,0003	10,18	0,1139	4	5
slashdot	Text	3782	1101	22	1079	156	56	525		1,1809	0,0537	17,69	0,0131	15,12	0,0210		21
stackex_chess	Text	1675	812	227	585	1078	890	48		2,4113	0,0106	85,79	0,2625	18,78			226
stackex_cs	Text	9270	909	274	635	4749	3679	119		2,5562	0,0093	85,00	0,2723	20,53	0,1071		273
tmc-2007-500	Text	28596	522	22	500	1172	408	2484		2,2196	0,1009	17,13	0,1927	16,37	0,1124		21
VirusGO	Biology	207	755	6	749	17	6	56		1,2174	0,2029	4,04	0,0079	11,24	0,1429	4	5
VirusPseAAC	Biology	207	446	6	440	17	6	56		1,2174	0,2029	4,04	0,0079	10,71	0,1429	4	5
yahoo_arts	Text	7484	23172	26	23146	599	358	1240	599	1,6539	0,0636	94,74	0,0595	19,70		81	25
yahoo_business	Text	11214	21954	30	21924	233	115	5992	233	0,1599	0,0533	880,18	0,1252	18,85		49	29
yahoo_computers	Text	12444	34129	33	34096	428	239	4122	428	1,5072	0,0457	176,70	0,0965	19,99	0,1910	64	32
yahoo_education	Text	12030	27567	33	27534	511	283	2611	511	1,4632	0,0443	168,11	0,0418	19,96	0,1919	49	32
yahoo_entertainment	Text	12730	32022	21	32001	337	151	2769	337	1,4137	0,0673	64,42	0,0387	19,24	0,1603	36	20
yahoo_health	Text	9205	30637	32	30605	335	169	2832	335	1,6441	0,0514	653,53	0,9200	19,61		25	31
yahoo_recreation	Text	12828	30346	22	30324	530	284	1652	530	1,4289	0,6500	12,20	0,0301	19,68		49	21
yahoo_reference	Text	8027	39712	33	39679	275	144	3038	275	1,1744	0,0356	461,86	0,0486	19,70		49	32
yahoo_science	Text	6428	37227	40	37187	457	252	1200	457	1,4498	0,0362	52,63	0,0575	20,34	0,2090	49	39
yahoo_social	Text	12111	52389	39	52350	361	179	4062	361	1,2793	0,0328	257,70	0,4900	20,42	0,2171	64	38
yahoo_society	Text	14512	31829	27	31802	1054	624	4092	1054	1,6704	0,0619	302,07	0,0957	20,62		64	26
yeast	Biology	2417	117	14	103	198	77	237	198	4,2371	0,3026	7,20	0,1044	12,56	0,2541	25	13
Yelp	Text	10806	676	5	671	32	0	2120	0	1,6383	0,3277	2,88	0,0332	11,58	0,1033	9	4

5.4 Partições Locais

As partições locais são geradas com base na abordagem multirrótulo local já explicada no Capítulo 2. A Figura 53 ilustra o processo de obtenção de partições locais. Em cada rótulo foi induzido ou o CLUS ou o RF, a depender do experimento conduzido.

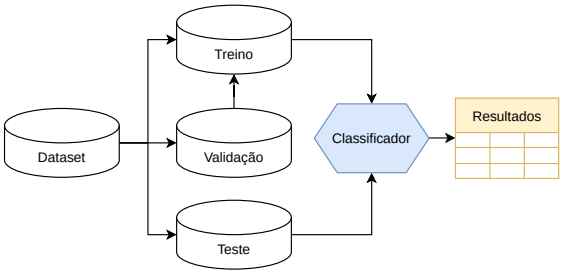


Figura 52 – Estratégia Global.

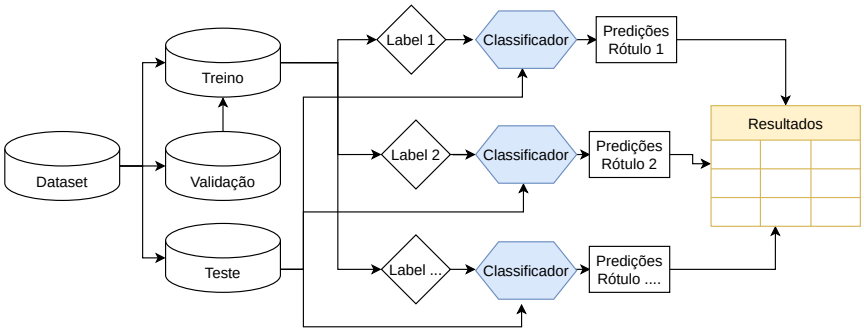


Figura 53 – Estratégia Local.

5.5 Partições Exaustivas

A Figura 54 apresenta uma visão geral da estratégia de busca exaustiva. Nesta estratégia, todas as partições possíveis são criadas e validadas usando os conjuntos de dados de validação e treino. A melhor partição é então escolhida e avaliada no conjunto de dados de teste. Os conjuntos de treinamento e validação são unidos em um único conjunto de dados de treinamento na avaliação final. Dois experimentos exaustivos foram conduzidos: i) $E\text{-}Ma$, onde a medida de avaliação multirrótulo Macro-F1 foi usada para escolher a melhor partição, e ii) $E\text{-}Mi$, onde a Micro-F1 foi usada como critério. Todas as partições possíveis para um dataset podem ser obtidas através do cálculo do número de Bell como já mencionado no Capítulo 1. As partições de número 1 correspondem às partições globais, enquanto partições de número $L - 1$[4] correspondem à partições locais, portanto, essas partições não precisam ser validadas ou testadas nos experimentos exaustivo e oráculo.

[4] Por exemplo, se o dataset tem 10 rótulos então são geradas 10 partições. O nome da última partição neste caso é "Partição 9" pois $L - 1 = 10 - 1 = 9$

5.6 Partições Oráculo

A Figura 55 apresenta uma visão geral da estratégia oráculo. A partição obtida pelo oráculo é a melhor partição possível no conjunto de dados de teste. O oráculo testa todas as partições possíveis usando os conjuntos de dados de treinamento[5] e teste e, em seguida, escolhe a melhor com base nas medidas de desempenho multirrótulo. A melhor partição encontrada usando o conjunto de dados de teste é chamada *Partição Oráculo*. Essa melhor partição é usada como referência de comparação com as partições encontradas pelos outros métodos. O objetivo na comparação entre as partições oráculos e as outras partições, é verificar o quão próximas (ou não) elas estão da partição oráculo.

Reforçando que todas as partições possíveis para um dataset são construídas, isto é, se um conjunto de dados possui 4 rótulos, então de acordo com o número de Bell 15 diferentes partições são geradas. Dentre essas 15 partições, uma é a global e a outra é a local, as quais não são testadas aqui, então as 13 partições restantes são construídas e testadas. A partição dentre essas 13 que obtiver ou o melhor desempenho preditivo, ou o melhor coeficiente de silhueta, é então escolhida como a Partição Oráculo, isto é, a partição mais adequada para o conjunto de dados em questão.

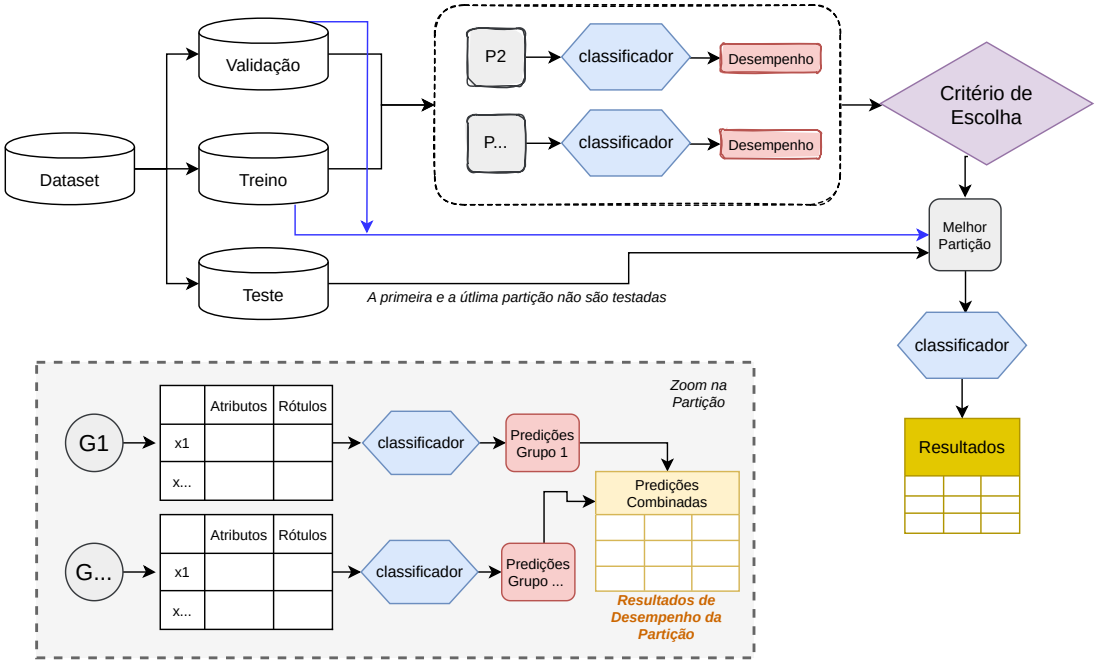


Figura 54 – Estratégia Exaustiva.

[5] neste caso juntam-se o conjunto de validação e de treino para formar o conjunto de treinamento

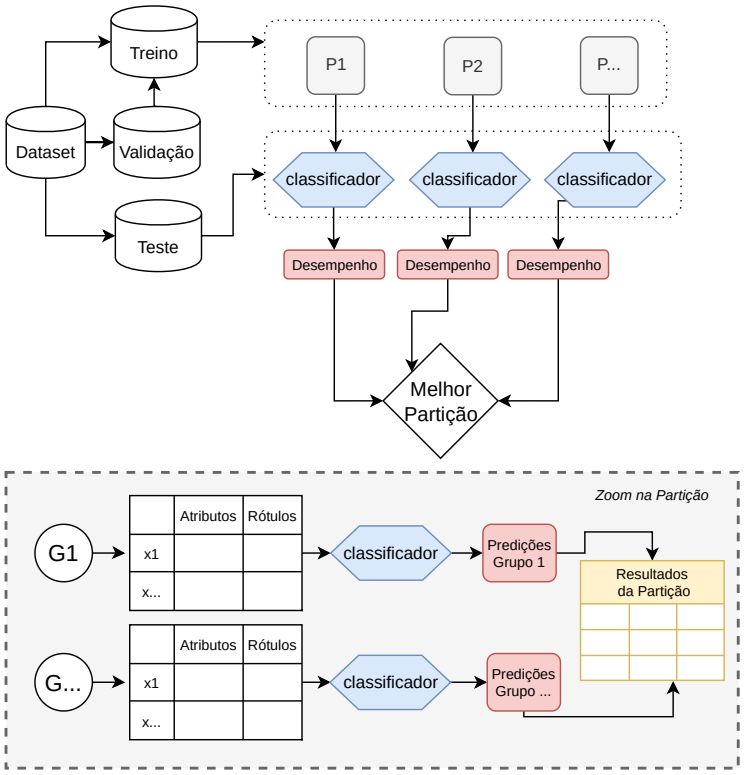


Figura 55 – Estratégia Oráculo.

5.7 Partições Aleatórias

5.7.1 Agrupamento Hierárquico Aglomerativo

Para comparar as partições híbridas geradas com o algoritmo hierárquico aglomerativo foram elaboradas três estratégias para gerar partições aleatórias, as quais serão explicadas a seguir. Para as versões 1 e 2 todas as partições aleatórias geradas são validadas e então uma entre elas é escolhida para ser testada. Em ambos os casos, são usadas a Macro-F1 e o coeficiente da silhueta como meios de validação e a partição com o maior valor é escolhida para o teste. A versão 3 gera uma única partição e portanto não há necessidade de validação.

5.7.1.1 Partição Aleatória Versão 1

A Figura 56 mostra a visão geral da primeira estratégia denominada aqui como $R1$. O número de partições geradas aqui varia de 1 até L, sendo que a primeira e a última partição correspondem às partições global e local. O número de grupos dentro de cada partição cresce juntamente com o número da partição e os rótulos são aleatoriamente distribuídos. Por exemplo, para um conjunto de dados com 6 rótulos no espaço de rótulos, 6 partições são geradas para a estratégia híbrida, sendo 1 global, 1 local e 4 híbridas. Para a estratégia aleatória versão 1 o processo é o mesmo do HPML, mas em vez de partições híbridas, obtemos partições aleatórias com agrupamentos de rótulos aleatórios.

5.7.1.2 Partição Aleatória Versão 2

A segunda versão também distribui os rótulos aleatoriamente em cada grupo como ocorre na versão 1, mas na versão 2 o número de clusters em cada partição é aleatório. A estratégia é denominada $R2$ e é representada pela Figura 57. Por exemplo, com seis rótulos no espaço de rótulos, são geradas seis partições e é escolhido um número aleatório de grupos para cada uma dessas partições. A partição 2 pode ser formada por 4 grupos, enquanto a partição 3 pode ter 2 grupos, enquanto na versão 1 a partição 2 possui 2 grupos, a partição 3 possui 3 grupos e assim por diante.

5.7.1.3 Partição Aleatória Versão 3

A terceira e última estratégia gera uma única partição aleatória, é denominada $R3$ e representada pela Figura 58. O número total de grupos é gerado aleatoriamente e os rótulos também são distribuídos aleatoriamente entre os grupos. Enquanto $R1$ e $R2$ geram várias partições aleatórias, $R3$ gera apenas uma, mas em todas as três estratégias os rótulos são distribuídos aleatoriamente em cada grupo.

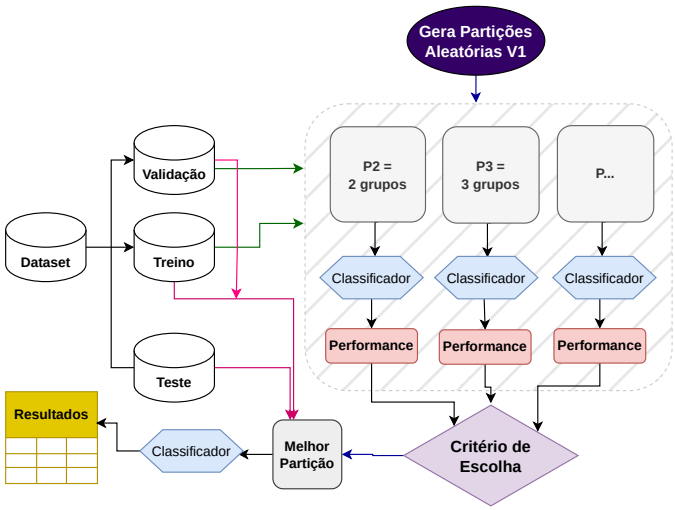


Figura 56 – Partições Aleatórias Versão 1.

5.7.2 Métodos de Detecção Comunidades

Para comparar as partições híbridas geradas pelos métodos de detecção de comunidade também foi desenvolvida uma estratégia para gerar partições aleatórias por meio desses métodos. A Figura 59 ilustra o processo da estratégia. Como já mencionado no Capítulo 2, os métodos de detecção de comunidade podem ser hierárquicos ou não hierárquicos, portanto, foi desenvolvido uma estratégia aleatória para ambos os tipos de métodos.

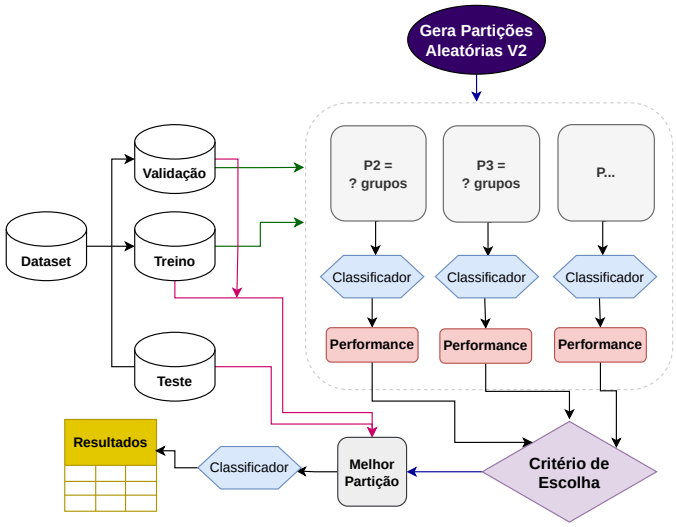


Figura 57 – Partições Aleatórias Versão 2.

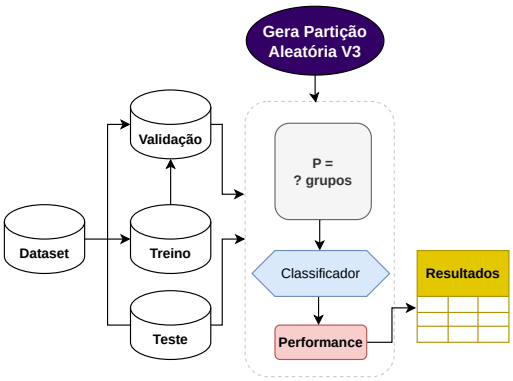


Figura 58 – Partições Aleatórias Versão 3.

O grafo de co-ocorrência aleatório de rótulos é construído a partir de uma matriz de adjacência do espaço de rótulos. Essa matriz de adjacência é o resultado do cálculo do número de instâncias classificadas em todos os pares de rótulos do conjunto de dados. Depois de obter a matriz de adjacência, a esparsificação k-NN é executada com um valor aleatório de k, gerando o grafo final.

Depois disso, aplicam-se os métodos que detectarão as comunidades. No caso dos métodos hierárquicos, novamente a modularidade é usada como critério de escolha do melhor método de comunidade que gerará o dendrograma e que será cortado para gerar as partições aleatórias. Nesse caso, para escolher a melhor partição aleatória, o coeficiente da silhueta é calculado e a partição com maior valor é escolhida para ser testada.

5.8 Ensemble of Classifier Chains

No caso do HPML.D, além de ter sido comparado com as partições local e global de Florestas Aleatórias nos experimentos, ele também foi comparado com o próprio ECC. A

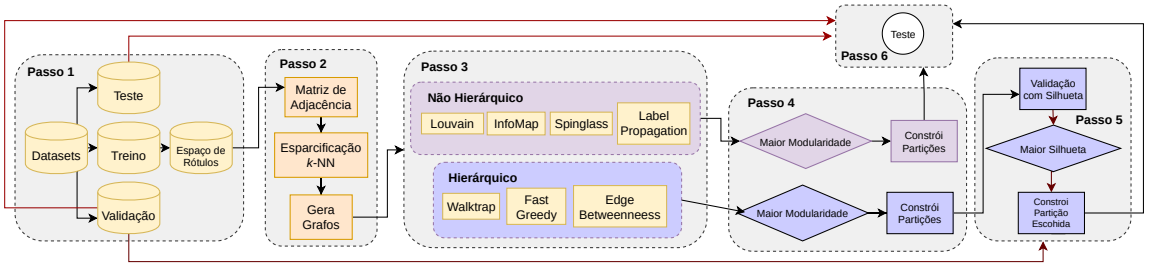


Figura 59 – Partições Aleatórias Comunidades. k é aleatório.

Figura 60 ilustra a metodologia do ECC.

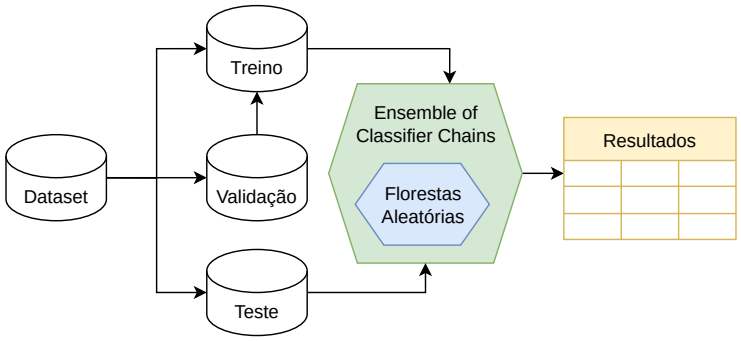


Figura 60 – ECC.

5.9 Montagem dos Experimentos

A Tabela 30 apresenta as características de cada um dos experimentos. Ao todo, quatro diferentes experimentos foram definidos com base nas quatro versões do HPML. As diferenças entre eles estão na técnica usada para modelar as correlações, na técnica escolhida para particionar o espaço de rótulos, nos critérios de escolha de melhor partição e na forma do teste. Os passos referentes à construção dos datasets para cada grupo de cada partição e validação ocorre de maneira semelhante para todos os experimentos. Nos experimentos HPML.AC, Exaustivo-Oráculo e Comunidades foi utilizado o classificador Clus, e no experimento Encadeamento foram usadas Florestas Aleatórias. Após a condução dos três primeiros experimentos, foi conduzido um experimento para verificar qual a melhor métrica de ligação para o algoritmo de agrupamento hierárquico aglomerativo, o qual é detalhado na Subseção 5.9.5.

Tabela 30 – Configuração dos Experimentos

Passo	**HPML.A*C***	**Exaustivo-Oráculo**	**Comunidades**	**Encadeamento**
2	*Matriz de Similaridade:* - Índice Jaccard	*Matriz de Similaridade:* - Índice Jaccard *Redes Neurais:* - Mapas Auto Organizáveis	*Grafos de Co-Ocorrência de Rótulos:* - Índice Jaccard - Rogers Tanimoto	*Matriz de Similaridade:* - Índice Jaccard
3	*Algoritmo de Agrupamento Hierárquico Aglomerativo:* - Melhor dendrograma escolhido através do maior coeficiente aglomerativo entre as métricas de ligação simples, completa e média	*Algoritmo de Agrupamento Hierárquico Aglomerativo:* - Dendrograma construído com a métrica de ligação simples	*Métodos de Detecção de Comunidades:* - Hierárquicos: + Maior modularidade entre WalkTrap, Edge Betweenneess e Fast Greedy - Não Hierárquico: + Maior modularidade entre Louvain, SpinGlass, InfoMap e Label Propagation	*Algoritmo de Agrupamento Hierárquico Aglomerativo:* - Dendrograma construído com a métrica de ligação Ward.D2
4	Corte nos Dendrogramas	Corte do Mapa	Corte nos Dendrogramas	Corte nos Dendrogramas
5	Classificador	Classificador e Silhueta	Silhueta	Silhueta
6	Maior Macro-F1	Maior Macro-F1, Micro-F1 e Silhueta	Maior Silhueta	Maior Silhueta
7	Clus	Clus	Clus	ECC e Random Forests

5.9.1 HPML.A.c

Este foi o primeiro experimento conduzido e parte dos seus resultados [6] foram publicados no IJCNN 2021[7]. Nesta montagem usou-se o Índice Jaccard, o algoritmo de agrupamento hierárquico aglomerativo com a seleção de uma entre três métricas de ligação, e validação com classificador tendo a medida Macro-F1 como critério de seleção da melhor partição. A Tabela 31 apresenta os métodos gerados neste experimento. Na tabela a sigla A.A.H.A. indica algoritmo de agrupamento hierárquico aglomerativo e o subscrito $_c$ indica CLUS. Foram comparadas as partições local, global, aleatórias e híbridas. O valor $PARCIAL$ é colocado em correlações no método global pois este método usa o dataset inteiro e, portanto, algumas das correlações naturalmente existentes entre os rótulos podem vir a ser aprendidas. Por fim, a Figura 61 ilustra a metodologia deste experimento.

5.9.2 Exaustivo-Oráculo

O mapa de Kohonen e o algoritmo de agrupamento hierárquico aglomerativo com ligação simples foram usados para gerar as partições híbridas neste experimento, aqui referenciado como Exaustivo-Oráculo. Novamente foi utilizado o Índice Jaccard e a validação foi feita com um classificador e também com o coeficiente da silhueta. A seleção de melhor partição foi feita com três diferentes critérios: macro-F1, micro-F1 e coeficiente da silhueta. Com isto, obteve-se como resultado diversos tipos diferentes de partições, as quais estão sumarizadas na Tabela 32. Além disso, também foram conduzido experimentos com as partições exaustivas e oráculo. Os resultados deste experimento foram submetidos à Knowledge and Information Systems[8]. Na Tabela 32 *A.A.H.A.* indica algoritmo de agrupamento hierárquico aglomerativo, *O* indica Oracle (oráculo), *E* indica Exhaustive (exaustivo), *R* indica Random, *S* indica Silhouete, *Ma* indica Macro-F1 e *Mi* indica Micro-F1. Foram comparadas as partições local, global, exaustivas, oráculo, aleatórias e híbridas. A Figura 62 representa uma abstração do processo deste experimento que representa os seis tipos de partições híbridas geradas aqui. Cada partição gerada ou pelo A.A.H.A, ou pelo SOM, é validada com um classificador e com o coeficiente da silhueta. Depois, as partições com o maior valor de Macro-F1, Micro-f1 e silhueta são escolhidas para teste.

[6] Por padrão, a biblioteca UTIML gera resultados para 22 medidas de avaliação multirrótulo, entre elas medidas de classificação e ranqueamento. Como são muitas medidas, optou-se neste projeto por utilizar-se apenas das CLP, MLP, WLP, Macro e Micro Precisão, Revocação e F1 para o HPML.A, HPML.B e HPML.C. No artigo do IJCNN foram publicados apenas a Macro e Micro Precisão e Revocação.

[7] <https://www.ijcnn.org/>

[8] <https://www.researchsquare.com/article/rs-3133411/v1>

5.9.3 Comunidades

Neste experimento grafos de co-ocorrência de rótulos usando as medidas de similaridade Jaccard e Rogers-Tanimoto foram gerados e então particionados usando métodos de detecção de comunidades para encontrar as partições híbridas. A validação foi feita usando apenas o coeficiente da Silhueta e apenas o classificador Clus foi utilizado no teste. A Tabela 33 apresenta os métodos gerados por este método. Parte dos resultados deste experimento foram publicados no BRACIS 2023 e foi premiado como o segundo melhor paper da conferência[9]. Na Tabela 33 J indica jaccard, R indica rogers-tanimoto, RA indica random, H indica método de comunidade hierárquico, NH indica método de comunidade não-hierárquico, K indica esparcificação com k-NN e T indica esparsificação com threshold (limiar). Foram comparadas as partições local, global, aleatórias e híbridas.

5.9.4 Encadeamento

Neste experimenhto as correlações foram modeladas usando o índice Jaccard e as partições geradas usando o algoritmo de agrupamento hierárquico aglomerativo com a métrica de ligação Ward.D2. A validação foi feita usando apenas o coeficiente da Silhueta e as Florestas Aleatórias foram utilizadas no teste. A Tabela 34 apresenta os métodos gerados por este método. As diferentes versões do HPML.D foram comparadas com as partições local, global e também com o ECC. As Figuras 46, 47, 48, 49, 50 e 51, apresentadas na Seção 4.4 são referência para este experimento.

Tabela 31 – Particionamentos gerados no HPML.AC

Acrônimo	Correlações	Parti.	Critério Escolha	Validação	Teste
1 G_C	Parcial	Único	Nenhum	Nenhum	Clus
2 Lo_C	Nenhuma	Binário	Nenhum	Nenhum	Clus
3 $HPML.A_C$	Jaccard	A.A.H.A.	Macro-F1	Clus	Clus
4 *R1*	Aleatório	Aleatório	Macro-F1	Clus	Clus
5 *R2*	Aleatório	Aleatório	Macro-F1	Clus	Clus

5.9.5 Melhor Métrica de Ligação

Após a execução dos três primeiros experimentos, um experimento considerando 70 conjuntos de dados multirrótulo foi conduzido para verificar qual é a melhor métrica de ligação para ser usada no algoritmo de agrupamento hierárquico aglomerativo. Para cada um dos conjuntos de dados da Tabela 29 foi calculado o coeficiente aglomerativo e o resultado foi que a métrica Ward.D2 obteve o maior coeficiente aglomerativo para esses conjuntos de dados avaliados. Portanto, após a condução deste experimento, optou-se por utilizar a métrica de ligação Ward.D2 no experimento de encadeamento.

[9] <https://www.bracis.dcc.ufmg.br/program/best-papers>

Tabela 32 – Particionamentos gerados no Exaustivo-Oráculo[10].

Acrônimo	Correlações	Parti.	Critério de Escolha	Validação	Test
1 G_C	Parcial	Único	Nenhuma	Nenhuma	Clus
2 Lo_C	Nenhum	Binário	Nenhuma	Nenhuma	Clus
3 E_{Ma}	Dataset	Todas possíveis	Macro-F1	Clus	Clus
4 E_{Mi}	Dataset	Todas possíveis	Micro-F1	Clus	Clus
5 O_{Ma}	Dataset	Todas possíveis	Macro-F1	Nenhuma	Clus
6 O_{Mi}	Dataset	Todas possíveis	Micro-F1	Nenhuma	Clus
7 $HPML.A_{Ma}$	Jaccard	A.A.H.A.	Macro-F1	Clus	Clus
8 $HPML.A_{Mi}$	Jaccard	A.A.H.A.	Micro-F1	Clus	Clus
9 $HPML.A_S$	Jaccard	A.A.H.A.	Silhouete	Silhouete	Clus
10 $HPML.B_{Ma}$	Kohonen	Corte	Macro-F1	Clus	Clus
11 $HPML.B_{Mi}$	Kohonen	Corte	Micro-F1	Clus	Clus
12 $HPML.B_S$	Kohonen	Corte	Silhouete	Silhouete	Clus
13 $R1_{Ma}$	Aleatório	Aleatório	Macro-F1	Clus	Clus
14 $R1_{Mi}$	Aleatório	Aleatório	Micro-F1	Clus	Clus
15 $R1_S$	Aleatório	Aleatório	Silhouete	Silhouete	Clus
16 $R2_{Ma}$	Aleatório	Aleatório	Macro-F1	Clus	Clus
17 $R2_{Mi}$	Aleatório	Aleatório	Micro-F1	Clus	Clus
18 $R1_S$	Aleatório	Aleatório	Silhouete	Silhouete	Clus
19 $R3$	Aleatório	Aleatório	Nenhuma	Nenhuma	Clus

Tabela 33 – Particionamentos gerados no experimento Comunidades.

Acrônimo	Correlações	Parti.	Critério Escolha	Validação	Teste
1 G_C	Parcial	Único	Nenhuma	Nenhuma	Clus
2 Lo_C	Nenhum	Binário	Nenhuma	Nenhuma	Clus
3 RA_H	Aleatório	Comunidades	Silhueta	Silhueta	Clus
4 RA_{NH}	Aleatório	Comunidades	Silhueta	Silhueta	Clus
5 $HPML.C_{JHK1}$	Jaccard	Comunidades	Silueta	Silhueta	Clus
6 $HPML.C_{JHK2}$	Jaccard	Comunidades	Silueta	Silhueta	Clus
7 $HPML.C_{JHK3}$	Jaccard	Comunidades	Silueta	Silhueta	Clus
8 $HPML.C_{JNHK1}$	Jaccard	Comunidades	Silhueta	Silhueta	Clus
9 $HPML.C_{JNHK2}$	Jaccard	Comunidades	Silhueta	Silhueta	Clus
10 $HPML.C_{JNHK3}$	Jaccard	Comunidades	Silhueta	Silhueta	Clus
11 $HPML.C_{JHT0}$	Jaccard	Comunidades	Silhueta	Silhueta	Clus
12 $HPML.C_{JHT1}$	Jaccard	Comunidades	Silhueta	Silhueta	Clus
13 $HPML.C_{JNHT0}$	Jaccard	Comunidades	Silhueta	Silhueta	Clus
14 $HPML.C_{JNHT1}$	Jaccard	Comunidades	Silhueta	Silhueta	Clus
15 $HPML.C_{RHK1}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
16 $HPML.C_{RHK2}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
17 $HPML.C_{RHK3}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
18 $HPML.C_{RNHK1}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
19 $HPML.C_{RNHK2}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
20 $HPML.C_{RNHK3}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
21 $HPML.C_{RHT0}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
22 $HPML.C_{RHT1}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
23 $HPML.C_{RNHT0}$	Rogers	Comunidades	Silhueta	Silhueta	Clus
24 $HPML.C_{RNHT1}$	Rogers	Comunidades	Silhueta	Silhueta	Clus

Tabela 34 – Particionamentos gerados no Encadeamento

Acrônimo	Correlações	Parti.	Critério Escolha	Validação	Teste
1 G_{RF}	Parcial	Único	Nenhuma	Nenhuma	Random Forests
2 Lo_{RF}	Nenhum	Binário	Nenhuma	Nenhuma	Random Forests
3 ECC_{RF}	Parcial	Binário	Nenhuma	Nenhuma	Random Forests
4 $HPML.D_{PADRAO}$	Jaccard	H.A	Silhueta	Silhueta	Random Forests
5 $HPML.D_{CI}$	Jaccard	H.A.	Silhueta	Silhueta	ECC + Random Forests
6 $HPML.D_{CE}$	Jaccard	H.A.	Silhueta	Silhueta	Random Forests
7 $HPML.D_{CEI}$	Jaccard	H.A.	Silhueta	Silhueta	ECC + Random Forests

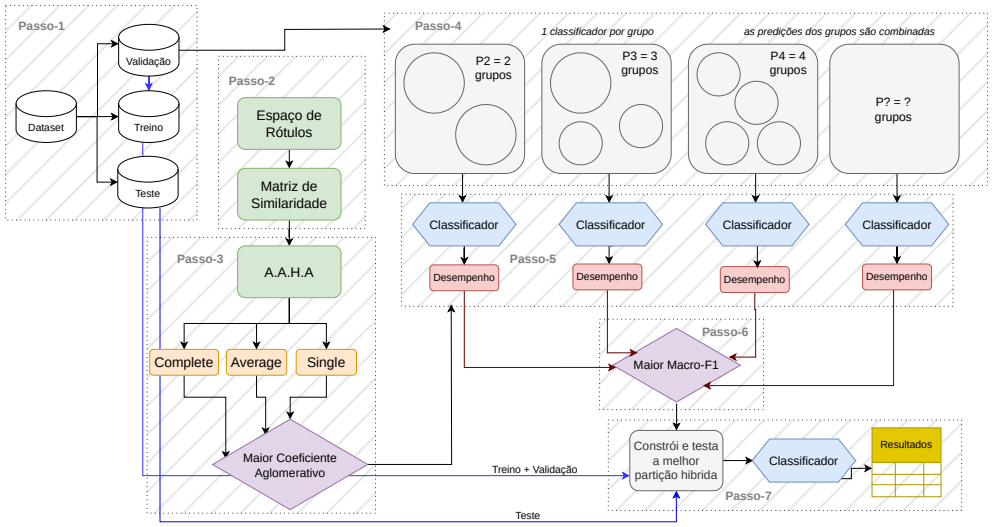


Figura 61 – Metodologia no HPML.A$_C$

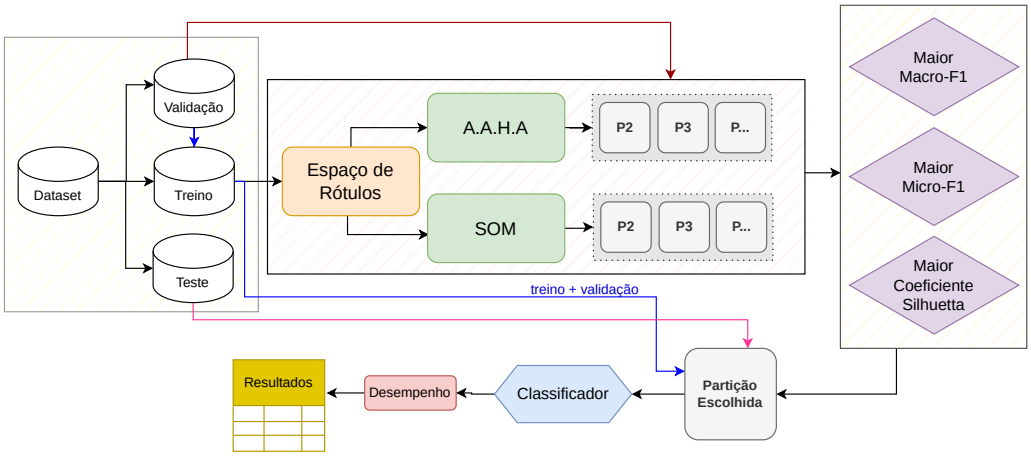


Figura 62 – Abstração das versões do HPML.A e HPML.B para o Exaustivo-Oráculo

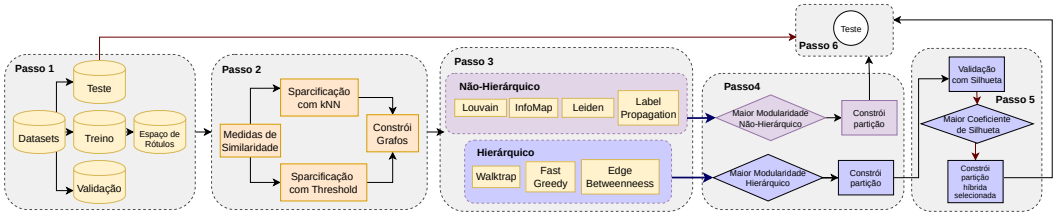


Figura 63 – Abstração do HPML.C

5.10 Considerações Finais

Esta seção apresentou os datasets utilizados em todos os experimentos conduzidos, com suas respectivas características, assim como todos os esquemas (ou fluxogramas) para cada experimento definido. No total foram gerados 49 diferentes métodos para particionamentos, incluindo as diferentes partições aleatórias, globais e locais. As linguagens de programação utilizadas para a implementação foram R e Python. Também foram utilizados recursos disponíveis pela própria universidade como Rclone, Google Drive e Servidores remotos. O próximo capítulo apresenta e discute os resultados obtidos em cada um dos set-ups.

Capítulo 6

Resultados e Discussão

Neste capítulo serão apresentados os resultados dos experimentos conduzidos. Todos os resultados estão disponíveis publicamente no repositório oficial da Tese[1]. As medidas de avaliação multirrótulo Micro e Macro Médias, assim como as medidas CLP, MLP e WLP foram as escolhidas para análise dos resultados dos três primeiros experimentos , e para o último experimento foram usadas as médias baseadas em curvas ROC e PR. Considerando que o HPML tenta melhorar o aprendizado dos classificadores por meio do aprendizado de grupos disjuntos de rótulos correlacionados, essas medidas são as mais adequadas pois são baseadas em rótulos.

6.1 HPML.A.c

A Tabela 35 apresenta os resultados de desempenho preditivo para as medidas de avaliação multirrótulo CLP, MLP, WLP, Macro e Micro F1, Precisão e Revocação. Os melhores valores estão marcados na cor azul, enquanto que os piores na cor vermelha. As últimas quatro linhas da tabela apresentam a média dos 10 folds, o valor máximo, o valor mínimo de desempenho e também o desvio padrão. A Tabela 36 sumariza as métricas de ligação escolhidas em cada fold para construir o dendrograma que deu origem às partições híbridas. É possível notar que para a grande maioria dos conjuntos de dados e folds, a métrica *single* obteve o melhor coeficiente aglomerativo.

Os resultados mostram que HPML.A$_C$ foi capaz de encontrar partições entre Lo$_C$ (local) e G$_C$ (global) onde os resultados foram competitivos com os obtidos usando as partições convencionais Lo$_C$, G$_C$, e as aleatórias $R1$ (random-1) $R2$ (random-2). Um resultado interessante que pode ser observado é que o uso de G$_C$ não levou aos melhores

[1] <https://github.com/cissagatto/HPML>

resultados como muitas vezes apontado na literatura. Isto pode ser um indicativo de que nesses conjuntos de dados, uma combinação de PCTs binárias é melhor do que o uso de apenas uma PCT multirrótulo.

Tabela 35 – Desempenho Preditivo

Dataset	CLP ↓					MLP ↓					WLP ↓				
	HPML.A.c	**R1**	**R2**	**Lo.c**	**G.c**	**HPML.A.c**	**R1**	**R2**	**Lo.c**	**G.c**	**HPML.A.c**	**R1**	**R2**	**Lo.c**	**G.c**
birds	**0,000**	0,005	**0,000**	**0,000**	**0,053**	**0,216**	0,432	0,337	0,216	**0,947**	0,400	0,553	0,468	**0,384**	**0,947**
emotions	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**
eukaPseAac	**0,000**	**0,000**	**0,000**	**0,000**	**0,041**	**0,264**	0,495	0,382	**0,264**	**0,945**	**0,545**	0,632	0,614	0,550	**0,945**
flags	**0,257**	0,143	0,086	**0,057**	0,229	**0,029**	**0,129**	**0,029**	0,043	0,100	0,100	**0,143**	**0,100**	0,114	0,100
gPositiveGo	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	0,025	**0,050**	**0,000**	0,025	**0,000**	0,050	**0,075**	**0,025**	0,050	**0,025**
plantGo	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,033**	0,125	0,042	**0,033**	**0,383**	**0,058**	0,142	0,067	0,058	**0,392**
scene	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**
virusGo	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	0,050	0,033	0,033	**0,017**	**0,083**	**0,000**	**0,067**	0,067	**0,000**	**0,000**
yeast	**0,000**	0,021	**0,000**	**0,000**	**0,071**	**0,071**	0,150	0,086	**0,071**	**0,236**	**0,071**	0,186	0,100	**0,071**	**0,279**
yelp	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**	**0,000**
Average	0,026	0,017	0,009	0,006	0,039	0,069	0,141	0,091	0,067	0,270	0,123	0,180	0,144	0,123	0,269
Max.	0,257	0,143	0,086	0,057	0,229	0,264	0,495	0,382	0,264	0,947	0,545	0,632	0,614	0,550	0,947
Min.	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
S.D.	0,081	0,045	0,027	0,018	0,072	0,094	0,180	0,144	0,094	0,378	0,191	0,228	0,216	0,190	0,382

Dataset	Macro-F1 ↑					Macro-Precision ↑					Macro-Recall ↑				
	HPML.A.c	**R1**	**R2**	**Lo.c**	**G.c**	**HPML.A.c**	**R1**	**R2**	**Lo.c**	**G.c**	**HPML.A.c**	**R1**	**R2**	**Lo.c**	**G.c**
birds	0,3076	0,2254	0,2845	**0,3169**	**0,0022**	0,3513	0,2768	0,3396	**0,3618**	**0,0011**	0,3272	0,2521	0,2965	**0,3378**	**0,0526**
emotions	**0,6107**	0,5876	0,5892	**0,5702**	0,5872	**0,6232**	0,6184	0,6062	**0,5857**	0,6033	**0,6142**	0,5805	0,5917	**0,5668**	0,5974
eukaPseAac	0,1039	0,0845	0,0938	**0,1042**	**0,0021**	**0,1116**	0,1096	0,1062	0,1109	**0,0065**	**0,1110**	0,1009	0,0927	0,1077	**0,0468**
flags	0,6268	0,5960	**0,6307**	0,6144	**0,5954**	0,6376	**0,5801**	**0,6459**	0,6277	0,6437	**0,6556**	0,6312	0,6427	0,6223	**0,6066**
gPositiveGo	0,8632	**0,8310**	**0,8823**	0,8749	0,8817	0,8546	**0,8190**	0,8807	0,8550	**0,8863**	0,8909	**0,8688**	0,9063	**0,9091**	0,9065
plantGo	0,6753	0,6246	0,6839	**0,6911**	**0,4199**	**0,6836**	0,6124	0,6723	0,6834	**0,4339**	0,7232	0,6797	0,7400	**0,7456**	**0,4448**
scene	**0,6187**	0,6326	0,6281	**0,6326**	0,6222	0,6215	0,6316	0,6229	**0,6173**	**0,6500**	0,6369	0,6567	0,6554	**0,6757**	**0,6080**
virusGo	0,8155	**0,8663**	0,8358	0,8659	**0,7868**	**0,8110**	**0,8619**	0,8295	0,8400	0,8349	0,8532	0,8926	0,8824	**0,9227**	**0,7819**
yeast	0,3980	0,3736	0,3927	**0,3985**	**0,3174**	0,4049	**0,3906**	0,4188	0,4046	**0,4546**	0,3937	0,3684	0,3831	**0,3952**	**0,2947**
yelp	0,6907	0,6794	0,6822	**0,6968**	**0,6163**	0,7146	**0,7209**	0,7174	0,7184	**0,6926**	0,6745	0,6517	0,6581	**0,6819**	**0,5705**
Average	0,571	0,550	0,570	0,577	0,483	0,581	0,562	0,584	0,580	0,521	0,588	0,568	0,585	0,596	0,491
Max.	0,863	0,866	0,882	0,875	0,882	0,855	0,862	0,881	0,855	0,886	0,891	0,893	0,906	0,923	0,906
Min.	0,104	0,084	0,094	0,104	0,002	0,112	0,110	0,106	0,111	0,001	0,111	0,101	0,093	0,108	0,047
S.D.	0,234	0,251	0,245	0,242	0,299	0,228	0,237	0,235	0,230	0,307	0,242	0,255	0,257	0,256	0,285

Dataset	Micro-F1 ↑					Micro-Precision ↑					Micro-Recall ↑				
	HPML.A.c	**R1**	**R2**	**Lo.c**	**G.c**	**HPML.A.c**	**R1**	**R2**	**Lo.c**	**G.c**	**HPML.A.c**	**R1**	**R2**	**Lo.c**	**G.c**
birds	0,3452	0,2892	0,3480	**0,3572**	**0,0215**	0,3007	0,2619	**0,3130**	0,3113	**0,0217**	0,4065	0,3253	0,3943	**0,4203**	**0,0214**
emotions	**0,6221**	0,5992	0,5969	**0,5805**	0,6123	**0,6210**	0,6066	0,5955	**0,5841**	0,6083	**0,6238**	0,5938	0,6002	**0,5776**	0,6182
eukaPseAac	0,2651	0,2363	0,2609	**0,2695**	**0,0056**	0,2482	0,2303	0,2481	**0,2506**	**0,0060**	0,2845	0,2431	0,2753	**0,2917**	**0,0053**
flags	**0,7360**	0,7270	0,7273	**0,7179**	0,7211	0,7053	**0,7049**	0,7165	0,7108	**0,7259**	**0,7714**	0,7521	0,7389	0,7262	**0,7222**
gPositiveGo	0,9422	0,9379	0,9462	**0,9463**	**0,9367**	0,9325	**0,9243**	0,9349	0,9333	**0,9403**	0,9522	0,9522	0,9580	**0,9599**	**0,9331**
plantGo	0,7624	0,7617	**0,7718**	0,7705	**0,7200**	0,7311	0,7286	**0,7383**	0,7365	**0,7150**	0,7972	0,7993	**0,8097**	0,8087	**0,7254**
scene	**0,6065**	0,6167	0,6131	**0,6170**	0,6163	0,5914	0,5931	0,5858	**0,5765**	**0,6384**	0,6239	0,6437	0,6441	**0,6638**	**0,5957**
virusGo	0,9007	**0,9174**	0,9012	0,9004	**0,8842**	0,8700	0,8855	0,8633	**0,8587**	**0,8866**	0,9381	**0,9530**	0,9453	0,9489	**0,8827**
yeast	**0,5527**	0,5713	0,5675	0,5551	**0,5929**	**0,5605**	0,5940	0,5837	0,5609	**0,6881**	0,5458	0,5516	**0,5530**	0,5498	**0,5218**
yelp	0,7409	0,7298	0,7350	**0,7485**	**0,6716**	0,7420	0,7405	0,7434	**0,7493**	**0,7036**	0,7398	0,7197	0,7270	**0,7478**	**0,6423**
Average	0,647	0,639	0,647	0,646	0,578	0,630	0,627	0,632	0,627	0,593	0,668	0,653	0,665	0,669	0,567
Max.	0,942	0,938	0,946	0,946	0,937	0,933	0,924	0,935	0,933	0,940	0,952	0,953	0,958	0,960	0,933
Min.	0,265	0,236	0,261	0,270	0,006	0,248	0,230	0,248	0,251	0,006	0,284	0,243	0,275	0,292	0,005
S.D.	0,218	0,233	0,219	0,217	0,318	0,221	0,230	0,219	0,219	0,322	0,216	0,237	0,220	0,216	0,317

Tabela 36 – Métricas de ligação escolhidas

Fold	Birds	EukaryotePseAAC	Emotions	Flags	GpositiveGO	PlantGO	Scene	VirusGO	Yeast	Yelp
1	complete	single	complete	complete	complete	single	single	single	single	single
2	complete	single	complete	complete	average	single	single	single	single	single
3	complete	single	complete	complete	average	single	single	single	single	single
4	average	single	complete	complete	complete	single	single	single	single	single
5	single	single	complete	complete	complete	single	single	single	single	single
6	complete	single	complete	complete	single	single	single	single	single	single
7	complete	single	complete	complete	single	single	single	single	single	single
8	complete	single	complete	complete	single	single	single	single	single	single
9	average	single	complete	complete	complete	single	single	single	single	single
10	complete	single	complete	complete	complete	single	single	single	single	single

Analisando os resultados de CLP, MLP e WLP nota-se que os conjuntos de dados emotions, scene e Yelp obtiveram o melhor desempenho em todos os particionamentos em todas essas medidas. Esses conjuntos de dados possuem poucos rótulos e a escolha das melhores partições híbridas e aleatórias para eles foi diversa nos folds, o que pode ser um indicativo de que, para conjuntos de dados com poucos rótulos, a dificuldade de

aprendizado é menor. Relembrando: Constant Label Problem (CLP) mede quando o mesmo rótulo é predito para todas as instâncias; Wrong Label Prediction (WLP) mede quando o rótulo pode ser predito para algumas instâncias, mas essas predições estão sempre erradas; Missing Label Prediction (MLP) calcula a proporção de rótulos que nunca são preditos. Isso pode ser confirmado olhando-se para os resultados das macro e micro médias desses conjuntos de dados: os resultados foram satisfatórios ainda que haja espaço para melhora. Esses conjuntos de dados também tem valores de TCS($\sim 0.9, \sim 10.0, \sim 11.0$), Scumble ($\sim 0,01, \sim 1.2, \sim 0.03$) e MeanIR[2] ($\sim 1.4, \sim 1.2, \sim 2.8$) baixos, além de que possuem poucas combinações de rótulos únicas.

Chama a atenção os valores de MLP e WLP em G_C para os conjuntos de dados birds (0.9470) e EukaryotePseAAC (0.9450) e a consequência disso é claramente observável nos resultados das macro-micro médias: valores abaixo de 0.1 indicando que o classificador global não foi capaz de aprender os rótulos. Nos outros particionamentos os valores foram próximos e em alguns casos idênticos, sendo EukaryotePseAAC e birds os casos mais graves. Mais uma vez isso pode ser constatado no desempenho das macro-micro médias, onde os valores para esses dois conjuntos de dados foram muito baixos (~ 0.3 e ~ 0.1) para todos os tipos de particionamentos. Além disso, esses dois conjuntos de dados possuem um valor alto de TCS ($\sim 13.0, \sim 14.0$), MeanIR ($\sim 6.0, \sim 45.0$) e combinações únicas de rótulos $(73, 37)$, e baixo valor de Scumble ($\sim 0.03, \sim 0.01$), características que podem ter afetado o aprendizado.

O dataset PlantGO obteve baixos valores para CLP, MLP e WLP. Com exceção de G_C nas macros, os outros particionamentos mantiveram-se na média: entre 0.6 e 0.7 na macro e entre 0.7 e 0.8 na micro. PlantGO possui 12 rótulos no espaço de rótulos, 8 combinações únicas de rótulos, com TCS = 13.9869, Scumble = 0.0058 e MeanIR = 6.6904. Comparando com os piores casos, o valor de TCS não é tão diferente, mas há diferença no valor de Scumble e, no caso do EukaryotePseAAC o valor de MeanIR é muito alto. Já nos melhores casos, os valores de TCS e Scumble são próximos, mas os valores de MeanIR são muito mais baixos. Diante disso, é possível concluir para os conjuntos de dados analisados que um alto valor da média do nível de desbalanceamento pode levar a um baixo desempenho do classificador e que o nível de concorrência entre os rótulos não impacta tanto no processo de aprendizagem.

Também foi contabilizado em quantos conjuntos de dados um particionamento obteve melhor desempenho preditivo comparado aos outros particionamentos, isto é, uma comparação pareada que resulta em uma tabela método X método e com o total de conjuntos de dados como resultado. Os resultados estão sumarizados na Tabela 37 para as medidas de avaliação multirrótulo analisadas. A interpretação desta tabela é feita da seguinte forma: o particionamento da linha obteve melhor desempenho que o particionamento da

[2] TCS: Theoretical Complexity Score, Scumble: concorrência de rótulos, MeanIR: média de desbalanceamento

coluna em D conjuntos de dados.

Tabela 37 – Comparação Pareada

CLP

	HPML.A$_C$	R1	R2	Lo$_C$	G$_C$	Média
HPML.A$_C$	0	2	0	0	3	**1,25**
R1	1	0	0	0	4	**1,25**
R2	1	3	0	0	4	**2,00**
Lo$_C$	1	3	1	0	4	**2,25**
G$_C$	1	0	0	0	0	**0,25**
Média	**1,00**	**2,00**	**0,25**	**0,00**	**3,75**	

MLP

	HPML.A$_C$	R1	R2	Lo$_C$	G$_C$	Média
HPML.A$_C$	0	5	4	2	6	**4,25**
R1	2	0	0	1	5	**2,00**
R2	2	7	0	2	6	**4,25**
Lo$_C$	1	6	5	0	6	**4,50**
G$_C$	1	2	0	1	0	**1,00**
Média	**1,50**	**5,00**	**2,25**	**1,50**	**5,75**	

WLP

	HPML.A$_C$	R1	R2	Lo$_C$	G.c	Média
HPML.A$_C$	0	7	6	3	5	**5,25**
R1	0	0	0	0	4	**1,00**
R2	1	7	0	1	5	**3,50**
Lo.c	1	7	6	0	5	**4,75**
G.c	0	3	1	1	0	**1,25**
Média	**0,50**	**6,00**	**3,25**	**1,25**	**4,75**	

Macro-Precisão

	HPML.A$_C$	R1	R2	Lo$_C$	G$_C$	Média
HPML.A$_C$	0	7	4	6	5	**5,50**
R1	3	0	5	4	6	**4,50**
R2	6	5	0	5	6	**5,50**
Lo$_C$	4	6	5	0	5	**5,00**
G$_C$	5	4	4	5	0	**4,50**
Média	**4,50**	**5,50**	**4,50**	**5,00**	**5,50**	

Macro-Revocação

	HPML.A$_C$	R1	R2	Lo$_C$	G$_C$	Média
HPML.A$_C$	0	8	6	3	9	**6,50**
R1	2	0	3	2	8	**3,75**
R2	4	7	0	2	8	**5,25**
Lo$_C$	7	8	8	0	9	**8,00**
G$_C$	1	2	2	1	0	**1,50**
Média	**3,50**	**6,25**	**4,75**	**2,00**	**8,50**	

Macro-F1

	HPML.A$_C$	R1	R2	Lo$_C$	G.c	Média
HPML.A$_C$	0	8	5	2	8	**5,75**
R1	2	0	2	2	9	**3,75**
R2	5	8	0	3	10	**6,50**
Lo$_C$	8	8	7	0	8	**7,75**
G$_C$	2	1	0	2	0	**1,25**
Média	**4,25**	**6,25**	**3,50**	**2,25**	**8,75**	

Micro-Precisão

	HPML.A$_C$	R1	R2	Lo.c	G$_C$	Média
HPML.A.c	0	7	4	3	5	**4,75**
R1	3	0	4	4	4	**3,75**
R2	6	6	0	8	4	**6,00**
Lo.c	7	6	2	0	4	**4,75**
G.c	5	6	6	6	0	**5,75**
Média	**5,25**	**6,25**	**4,00**	**5,25**	**4,25**	

Micro-Revocação

	HPML.A$_C$	R1	R2	Lo$_C$	G$_C$	Média
HPML.A.c	0	6	5	2	10	**5,75**
R1	4	0	2	4	9	**4,75**
R2	5	8	0	4	9	**6,50**
Lo.c	8	6	6	0	9	**7,25**
G.c	0	1	1	1	0	**0,75**
Média	**4,25**	**5,25**	**3,50**	**2,75**	**9,25**	

Micro-F1

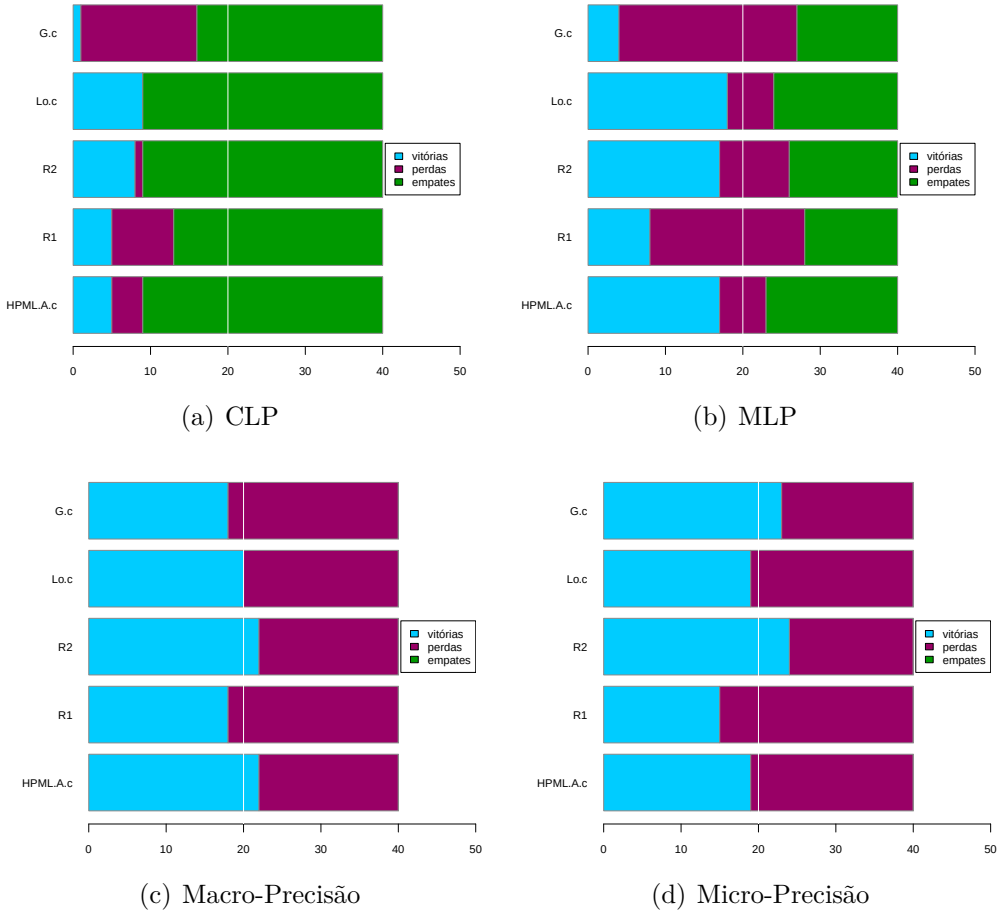
	HPML.A$_C$	R1	R2	Lo$_C$	G$_C$	Média
HPML.A.c	0	7	4	3	8	**5,50**
R1	3	0	4	4	8	**4,75**
R2	6	6	0	5	7	**6,00**
Lo.c	7	6	5	0	7	**6,25**
G.c	2	2	3	3	0	**2,50**
Média	**4,50**	**5,25**	**4,00**	**3,75**	**7,50**	

Observando a linha do HPML.A$_C$, nota-se que este particionamento teve desempenho melhor que o $R1$ em 7 dos 10 conjuntos de dados na Macro-Precision, enquanto que o $R1$ obteve melhor desempenho que o HPML.A$_C$ em apenas 3 dos 10 conjuntos de dados. Somando-se 3 com 7 temos 10 que é a quantidade total de conjuntos de dados avaliado neste experimento. Nas medidas CLP, MLP e WLP, há muitas situações em que os conjuntos de dados e métodos obtiveram o melhor resultado possível que é 0.0 e por isso o valor de um método não é maior que o outro, mas sim igual e o maior igual não foi computado nesta comparação pareada. Na Macro-Precisão, as partições híbridas foram melhores que as locais em 6 dos 10 conjuntos de dados, indicando que o classificador conseguiu melhor desempenho nesses conjuntos de dados ao aprender os grupos disjuntos de rótulos correlacionados.

O número de vitórias, derrotas e empates entre os métodos também foi contabilizado e pode ser visualizado nos gráficos de vitórias-derrotas-empates nas Figuras 64 e 65. O gráfico de vitórias-derrotas-empates é diferente da comparação entre pares apresentada na Tabela 37. O gráfico de vitórias-derrotas-empates apresenta as vitórias de um método contra todos os outros (o número de vezes em que o método foi melhor que todos os outros), enquanto que a comparação pareada mostra o total de conjuntos de dados em que um método M$_{linha}$ foi melhor que outro M$_{coluna}$.

Nas medidas de avaliação CLP, MLP, Macro-F1, Micro-F1, Macro-Revocação e Micro-Revocação, as partições locais têm o maior número de vitórias, enquanto que HPML.A.$_C$ tem o maior número de vitórias nas medidas WLP e Macro-Precisão. As partições R2 têm o maior número de vitórias em Macro e Micro-Precisão. Estes gráficos confirmam algumas das conclusões já relatadas: as partições locais tendem a obter os melhores resultados, as globais os piores, e por fim, as híbridas e aleatórias competitivas entre si. Por exemplo, na MLP, HPML.A.$_C$ e R2 têm exatamente o mesmo número de vitórias, enquanto que

na CLP o mesmo ocorre com HPML.A.$_C$ e R1, e por fim, também na Micro-Precisão os métodos HPML.A.$_C$ e Lo$_C$.



(a) CLP (b) MLP

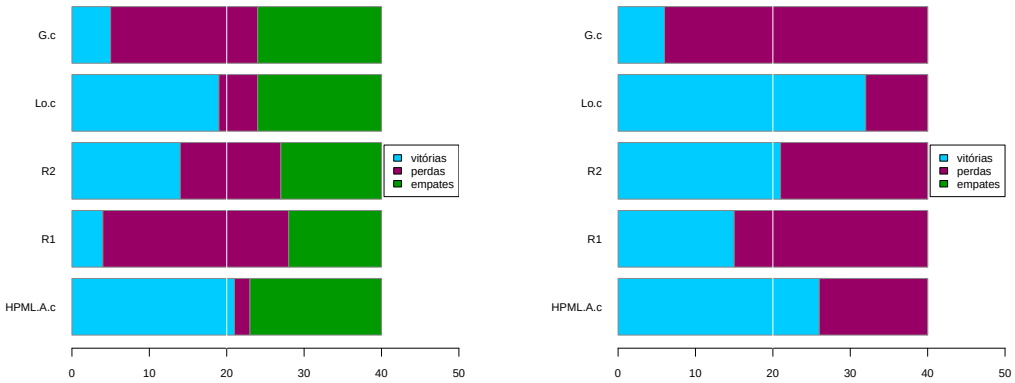
(c) Macro-Precisão (d) Micro-Precisão

Figura 64 – Gráficos de Vitórias, Derrotas e Empates para CLP, MLP, Macro-Precisão e Micro-Precisão. Comparação 1 X todos. Há vários empates em CLP e MLP, o que é normal devido à característica da medida: muitos casos iguais a 0. Há alta competitividade entre os métodos.

Tabela 38 – Partições Escolhidas. H = HPML.A$_C$

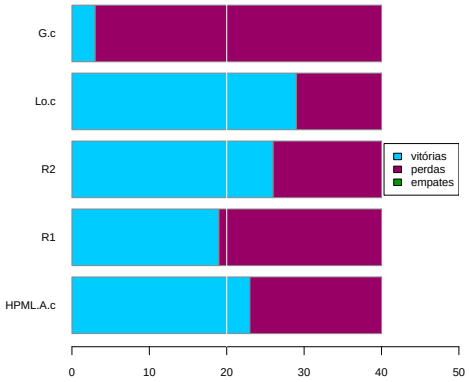
	Birds			**Emotions**			**EukaryotePseAAC**			**Flags**			**GpositiveGO**			**PlantGO**			**Scene**			**VirusGO**			**yeast**			**Yelp**		
Fold	**H**	**R1**	**R2**	**H**	**R1**	**R2**	**H**	**R1**	**R2**	**H**	**R1**	**R2**	**H**	**R1**	**R2**	**H**	**R1**	**R2**	**H**	**R1**	**R2**	**H**	**R1**	**R2**	**H**	**R1**	**R2**	**H**	**R1**	**R2**
1	18	18	6	2	3	4	21	3	2	6	2	3	2	3	2	10	11	2	5	4	2	3	2	3	13	13	13	4	4	2
2	18	18	2	2	3	4	21	3	11	2	4	6	2	2	2	4	10	10	3	5	5	5	5	3	13	13	2	4	4	2
3	18	18	12	4	5	4	21	5	3	6	4	6	2	2	2	11	10	10	2	4	2	2	4	2	13	13	2	4	2	2
4	18	18	2	2	3	2	21	3	2	2	3	5	3	3	3	11	11	2	3	5	5	5	3	3	13	13	13	4	2	2
5	18	18	3	4	5	5	21	5	3	6	6	2	2	3	2	11	11	4	5	3	2	3	2	2	13	13	13	4	3	2
6	18	18	2	3	2	2	21	2	21	2	5	4	2	2	2	11	11	4	3	4	2	5	5	2	13	10	13	4	4	4
7	18	18	18	4	5	2	21	5	2	2	2	2	3	2	2	11	11	4	3	3	3	2	2	2	13	13	2	4	4	2
8	18	18	18	4	4	2	21	4	3	6	2	2	3	3	2	11	11	3	5	5	2	2	4	2	13	13	2	4	4	4
9	18	18	3	4	5	4	21	5	2	6	6	4	3	2	3	11	11	10	2	5	2	3	4	2	13	13	4	4	3	2
10	18	18	2	2	3	2	21	3	5	6	6	2	2	2	2	11	11	11	2	5	5	5	5	3	13	12	13	4	3	2

Analisando a partição híbrida escolhida (Tabela 38) do conjunto de dados de Flags, em seis das dez execuções no experimento de validação cruzada, a melhor partição (maior valor macro-F1) selecionada por HPML.A$_C$ foi uma partição híbrida onde apenas um grupo é composto por dois rótulos correlacionados, enquanto todos os outros grupos são

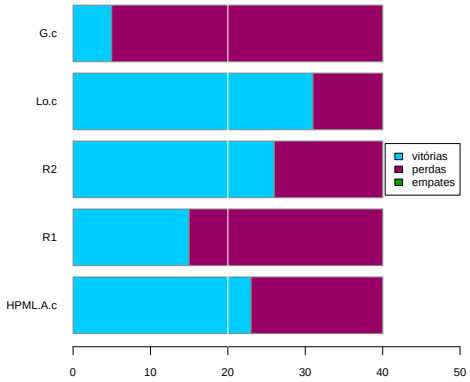


(a) WLP

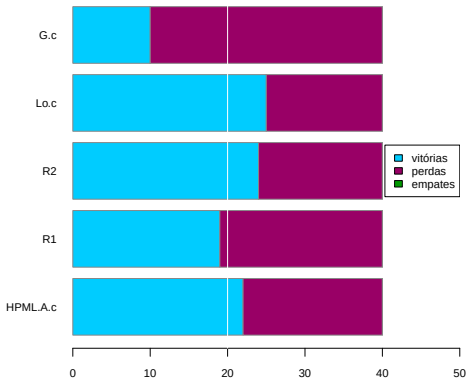
(b) Macro-Revocação



(c) Micro-Revocação



(d) Macro-F1



(e) Micro-F1

Figura 65 – Gráficos de Vitórias, Derrotas e Empates para WLP, Macro-Revocação, Micro-Revocação, Macro-F1 e Micro-F1. Há vários empates na WLP, mas nas outras medidas há alta competitividade entre as partições híbridas, aleatórias e locais. As partições geradas pelo método global tem o menor número de vitórias.

compostos por rótulos individuais (P6). Assim, como o dataset Flags tem sete rótulos (P7=local, P1=global, P6 a P2 são híbridas), o modelo resultante foi uma PCT multirrótulo e cinco PCTs binárias, o que é muito semelhante a induzir apenas classificadores binários na partição local. Isso também foi observado nos conjuntos de dados birds, EukaryotePseAAC, yeast e yelp, e essa configuração de partição foi escolhida em todas as 10 vezes. Também nas partições $R1$ isso foi observado em birds, PlantGO e yeast.

As melhores partições aleatórias escolhidas em $R1$ no dataset birds foram as mesmas que a híbrida (18 - mas tem distribuição de rótulos aleatória) e obtiveram resultados piores quando comparadas. Isso também foi observado em outros conjuntos de dados, como PlantGO e yeast. Isso mostra que as partições híbridas foram capazes de melhorar as predições para esses conjuntos de dados. No caso de $R1$, é possível que a aleatoriedade não tenha ajudado o suficiente para melhorar as predições.

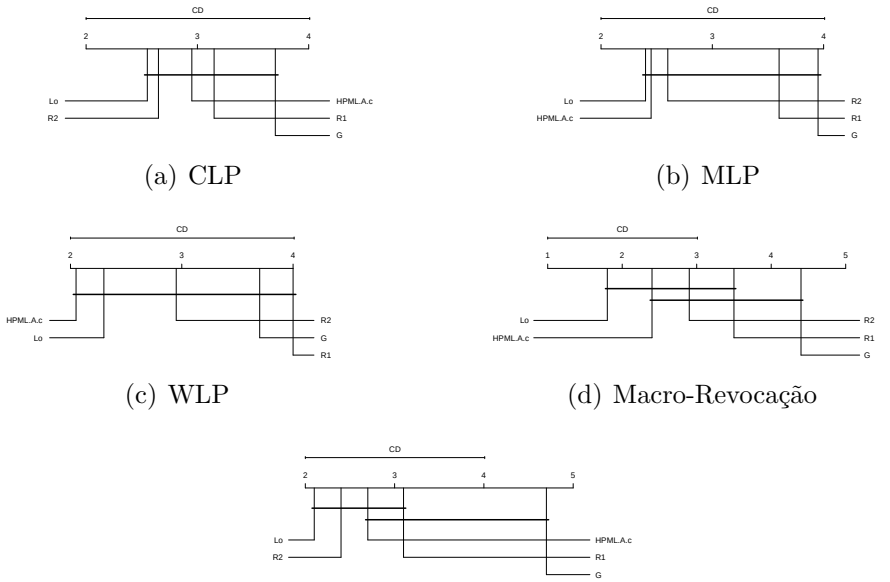
É possível também que dois rótulos altamente correlacionados possam ser agrupados em uma partição aleatória em um conjunto de dados com poucos rótulos, como os conjuntos de dados GpositiveGO e Yelp. Em um conjunto de dados com um alto número de rótulos, isso pode não ser verdade. Também é provável que em $R1$, para conjuntos de dados com poucos rótulos, a aleatoriedade não esteja alterando muito a distribuição dos rótulos.

Para verificar a significância estatística dos resultados, foi executado o teste estatístico de Friedman seguido do teste post-hoc de Nemenyi. Eles são recomendados, pois não são paramétricos e não fazem suposições sobre a distribuição dos dados (DEMSAR, 2006). Os p-values e as hipóteses para cada medida estão listadas na Tabela 39. Com $\alpha = 0,05$ conclui-se que existem diferenças estatisticamente significativas para as medidas Macro-Revocação, Macro-F1, e Micro-Revocação

Em seguida foi executado o teste post-hoc de Nemenyi para verificar onde estão presentes as diferenças estatísticas. A Figura 66 mostra os diagramas de distância crítica obtidos comparando os resultados obtidos pelas PCTs quando executada nas partições HPML.A$_C$, Lo$_C$, G$_C$, $R1$ e $R2$ sendo a distância crítica de $\sim 2,01$. As linhas conectadas mostram onde não foram detectadas diferenças estatisticamente significativas. Para CLP, MLP, WLP, Macro-Precisão, Micro-Precisão e Micro-F1 não foram detectadas diferenças estatísticas significativas.

Tabela 39 – Experimento 1: Resultados do Teste de Friedman

Medida	**ChiSquare**	$p-$**Value**	**Hipótese**
CLP	3,36	0,499482256	H0:Identical
MLP	8,34	0,079888185	H0:Identical
WLP	11,54	0,021120758	Ha:Different
Macro-Precisão	0,64	0,958516729	H0:Identical
Micro-Precisão	2,08	0,721047551	H0:Identical
Macro-Revocação	16,08	0,002913673	Ha:Different
Micro-Revocação	16,64	0,002270313	Ha:Different
Macro-F1	16,64	0,002270313	Ha:Different
Micro-F1	5,84	0,211420054	H0:Identical



(a) CLP (b) MLP

(c) WLP (d) Macro-Revocação

(e) Micro-Revocação

(f) Macro-Precisão (g) Micro-Precisão

(h) Macro-F1 (i) Micro-F1

Figura 66 – Gráficos de Distância Crítica de Nemenyi

6.2 Exaustivo-Oráculo

O principal objetivo deste experimento foi o de estudar o quanto uma partição híbrida é próxima da melhor partição possível para um dataset. Em problemas de classificação multirrótulo, existe um grande número de partições possíveis, e encontrar a ótima que maximize o poder preditivo de um classificador é uma tarefa desafiadora. O número de todas as partições possíveis de um conjunto de n rótulos, com cada partição consistindo em k subconjuntos não vazios distintos (COMTET, 1974; SPIVEY, 2008; MEZO, 2011), pode ser calculado pelo número de Bell (B_n) de acordo com a Equação 1 e como já explicado no Capítulo 1. Nesta seção são apresentadas as seguintes análises: 1) entre os particionamentos escolhidos com a Macro-F1, 2) entre os particionamentos escolhidos com a Micro-F1.

6.2.1 Análise da Macro-F1

A Tabela 40 apresenta os resultados do desempenho preditivo para a medida de avaliação Macro-F1 com métodos que utilizaram o critério de seleção de silhueta e Macro-F1. Os melhores resultados possíveis são os obtidos pelo oráculo (O_{Ma}) e estão destacados na cor verde. Os resultados mais próximos do oráculo são considerados os melhores entre todos os outros métodos, e são destacados em azul, enquanto que os piores resultados são destacados em vermelho. Para simplificar a apresentação de tabelas e figuras, os acrônimos $HPML.A$ e $HPML.B$ foram substituídos apenas por H, assim os nomes das partições híbridas ficam no formato $H_{correlacao-criterio}$.

Tabela 40 – Resultados Macro-F1

MACRO-F1													
Dataset	G_C	Lo_C	$H.A_{Ma}$	$H.A_S$	$H.B_{Ma}$	$H.B_S$	E_{Ma}	O_{Ma}	$R1_{Ma}$	$R1_S$	$R2_{Ma}$	$R2_S$	R3
emotions	**0,5553**	0,5633	**0,5931**	0,5666	**0,5931**	0,5721	0,5840	**0,6095**	0,5859	0,5600	0,5712	0,5635	0,5824
flags	0,6122	0,5842	0,5997	0,5771	0,5787	0,5975	0,5690	**0,6534**	0,5867	0,5920	0,5890	**0,5651**	**0,6123**
GpositiveGO	**0,8859**	0,8847	0,8706	0,8726	**0,8568**	0,8726	0,8764	**0,8994**	0,8802	0,8798	0,8679	0,8603	0,8605
GpositivePseAAC	0,4104	0,4146	0,3928	**0,3918**	0,4024	0,4019	0,4113	**0,4699**	**0,4515**	0,4391	0,4073	0,4095	0,4331
scene	**0,6006**	**0,6299**	0,6081	0,6193	0,6025	0,6052	0,6171	**0,6395**	0,6113	0,6027	0,6172	0,6294	0,6052
VirusGO	0,7870	0,8786	0,8361	0,8777	**0,7599**	0,7662	0,8365	**0,8911**	0,8434	0,7805	**0,8790**	0,8533	0,8634
VirusPseAAC	**0,1197**	0,2741	0,2289	0,2390	0,2058	0,1981	0,2744	**0,3560**	0,2888	0,1704	0,2550	0,1859	**0,3367**
Yelp	**0,6372**	**0,7016**	0,6976	0,6856	0,6840	0,6724	0,7006	**0,7016**	0,6838	0,6644	0,6922	0,6822	0,7004
Média	**0,5760**	**0,6164**	**0,6034**	**0,6037**	**0,5854**	**0,5858**	**0,6087**	**0,6525**	**0,6164**	**0,5861**	**0,6099**	**0,5937**	**0,6242**

Tabela 41 – Comparação pareada Macro-F1

	G_C	Lo_C	$H.A_{Ma}$	$H.A_S$	$H.B_{Ma}$	$H.B_S$	E_{Ma}	O_{Ma}	$R1_{Ma}$	$R1_S$	$R2_{Ma}$	$R2_S$	R3	**Média**
G_C	0	2	3	3	4	4	2	0	2	3	3	3	1	**2,7**
Lo_C	6	0	6	7	7	6	6	0	4	6	5	7	4	**5,8**
$H.A_{Ma}$	5	2	0	4	6	6	2	0	3	6	4	5	3	**4,2**
$H.A_S$	5	1	4	0	5	4	3	0	3	5	2	6	3	**3,7**
$H.B_{Ma}$	4	1	1	3	0	4	2	0	2	3	1	4	1	**2,4**
$H.B_S$	4	2	2	3	4	0	1	0	1	5	3	4	2	**2,8**
E_{Ma}	6	2	6	5	6	7	0	0	2	5	5	6	4	**4,9**
O_{Ma}	8	8	8	8	8	8	8	0	8	8	8	8	8	**8,0**
$R1_{Ma}$	6	4	5	5	6	7	6	0	0	7	4	6	4	**5,5**
$R1_S$	5	2	2	3	5	3	3	0	1	0	3	3	2	**2,9**
$R2_{Ma}$	5	3	4	6	7	5	3	0	4	5	0	6	3	**4,6**
$R2_S$	5	1	3	2	4	4	2	0	2	5	2	0	1	**2,8**
R3	7	4	5	5	7	6	4	0	4	6	5	7	0	**5,5**
Média	**5,5**	**2,7**	**4,1**	**4,5**	**5,8**	**5,3**	**3,5**	**0,0**	**3,0**	**5,3**	**3,8**	**5,4**	**3,0**	

Como esperado, o oráculo sempre gera as partições que levam aos melhores resultados de Macro-F1. Considerando os demais métodos, a tabela também mostra que, na média dos 8 conjuntos de dados, a partição aleatória R3 obteve melhores resultados que os outros métodos. As partições híbridas levaram a resultados que podem ser considerados competitivos, pois, como os testes estatísticos irão mostrar, as diferenças para os outros métodos não são estatisticamente significativas.

A Tabela 41 apresenta uma comparação pareada de todos os métodos, mostrando o número de conjuntos de dados em que um método na linha teve melhor desempenho do que

um método na coluna. Ao comparar os métodos de particionamento híbrido com a busca exaustiva, nota-se que H.A$_{Ma}$ e H.A$_S$ obtiveram resultados competitivos, especialmente considerando que H.A$_{Ma}$ obteve o melhor resultado em um dos conjuntos de dados.

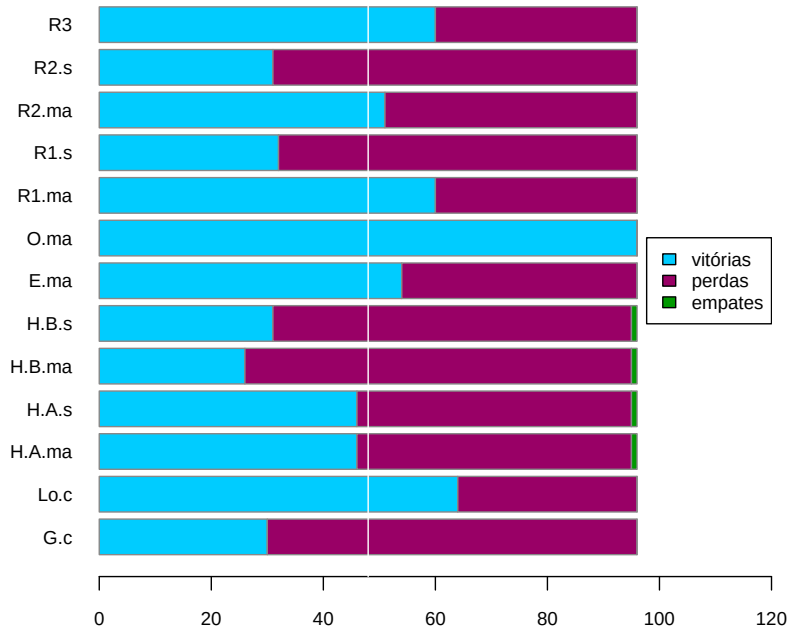


Figura 67 – Gráfico de Vitórias, Derrotas e Empates Macro-F1. Oráculo tem o maior número de vitórias como é o esperado, enquanto que o HPM.B$_{Ma}$ tem o menor. As partições híbridas, aleatórias e locais se mostram competitivas.

Também foram feitos os gráficos de vitórias, derrotas e empates que são apresentados na Figura 67. O total que pode ser alcançado aqui é de 96 vitórias (ou derrotas ou empates), pois tem-se 13 métodos e 8 conjuntos de dados, o que totaliza 104, mas é necessário retirar repetições, portanto 104-8 conjuntos de dados = 96. Como esperado, a partição oráculo é a vencedora (96) e também pode-se observar que os empates são mínimos, ocorrendo apenas entre as partições híbridas. A partição local tem um número de vitórias (64) maior que todos os outros particionamentos, mas está bem próxima das partições aleatórias versão 1 (60) e 3 (60).

O particionamento com o menor número de vitórias é o H.B$_{Ma}$ (26), mas R1$_S$, H.B$_S$, R2$_S$ e G estão em uma faixa similar, entre 32 e 30 vitórias. As partições híbridas H.A$_{Ma}$ e H.A$_S$ ficaram na faixa média, com 46 e 41 vitórias respectivamente. Por fim, a partição exaustiva e aleatória versão 2 tem mais vitórias quando comparadas com outros métodos, 54 e 51. Neste cenário, é compreensível que as partições exaustivas tenham mais vitórias que as partições híbridas, já que na estratégia exaustiva todas as possíveis partições são validadas.

No geral, as variantes dos métodos aleatórios e híbridos obtiveram resultados competitivos. É interessante observar também que as partições aleatórias obtiveram resultados melhores ou competitivos quando comparadas às abordagens global e local. Isso é um

indicativo de que as abordagens convencionais podem não estar considerando adequadamente as correlações entre rótulos. Olhando para a abordagem global, fica claro que, todos os outros métodos que de alguma forma tentam aprender grupos disjuntos de rótulos correlacionados têm melhor desempenho. Portanto, deste experimento pode-se concluir que utilizar grupos disjuntos de rótulos correlacionados é superior a usar um único grupo com todos os rótulos.

Quando as partições híbridas são comparadas com as partições obtidas pela abordagem local, observa-se que na maioria dos conjuntos de dados, as partições locais produziram os melhores resultados. Isso pode ser porque os conjuntos de dados utilizados tem um pequeno número de rótulos, mas também pode haver influencia da medida de avaliação. A Macro-Precision e Macro-Recall são medidas que consideram os desempenhos dos métodos em cada rótulo e, em seguida, calculam a média desses resultados sobre o número de rótulos. Nesse caso, induzir um classificador binário por rótulo pode ser vantajoso.

Além disso, as partições híbridas levaram a resultados melhores ou competitivos em alguns casos, indicando que os grupos disjuntos de rótulos correlacionados podem levar a melhores resultados, o que é particularmente importante para conjuntos de dados com um número muito grande de rótulos. Os resultados dos particionamentos aleatórios suportam esta afirmação, uma vez que as partições geradas aleatoriamente levaram, em alguns casos, a melhores resultados do que a abordagem local.

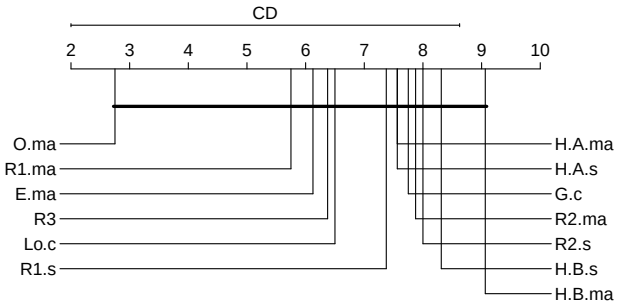


Figura 68 – Gráfico de Distância Crítica de Nemenyi para Macro-F1.

Para avaliar a significância estatística dos resultados, foi executado o teste de Friedman com nível de significância de 95% ($\alpha = 0,05$), obtendo um p-valor de $0,050175$ e ChiSquare de $37,4258241758242$. Aplicou-se então um teste post-hoc de Nemenyi, obtendo-se um valor de distância crítica de $6,060886776$, que foi utilizado para construir o diagrama mostrado na Figura 68. Como todos os métodos estão conectados no diagrama, concluiu-se que não foram encontradas diferenças estatisticamente significativas.

Pelos resultados dos testes estatísticos, percebe-se que as abordagens local e global produziram resultados semelhantes aos obtidos usando partições geradas aleatoriamente. Assim, os testes estatísticos apóiam a afirmação de que as abordagens globais e locais atuais ainda falham em lidar corretamente com as correlações entre rótulos. Os expe-

rimentos também mostram que partições aleatórias de rótulos podem levar a resultados competitivos.

6.2.2 Análise da Micro-F1

A Tabela 42 apresenta os resultados do desempenho preditivo para a medida de avaliação Micro-F1 com métodos que utilizaram o critério de seleção de silhueta e também Micro-F1. Os melhores resultados possíveis são os obtidos pelo oráculo (O_{Mi}) e estão destacados na cor verde. Os resultados mais próximos do oráculo são considerados os melhores entre todos os outros métodos, e são destacados em azul, enquanto que os piores resultados são destacados em vermelho. Para simplificar a apresentação da tabela de resultados, os acrônimos $HPML.A$ e $HPML.B$ foram substituídos apenas por H, assim os nomes das partições híbridas ficam no formato $H_{correlacao-criterio}$.

Tabela 42 – Resultados Micro-F1

MICRO-F1													
Dataset	G_C	Lo_C	$H.A_{Mi}$	$H.A_S$	$H.B_{Mi}$	$H.B_S$	E_{Mi}	O_{Mi}	$R1_{Mi}$	$R1_S$	$R2_{Mi}$	$R2_S$	R3
emotions	0,6012	0,5775	**0,6131**	0,6047	0,6034	0,6066	0,6023	**0,6343**	0,6038	**0,5754**	0,5949	0,5878	0,5906
flags	0,7276	0,7179	0,7327	0,7236	0,7276	**0,7385**	0,7384	**0,7461**	0,7256	0,7292	0,7158	0,7102	**0,6908**
GpositiveGO	0,9405	**0,9473**	0,9402	0,9428	0,9421	0,9428	0,9456	**0,9511**	0,9449	0,9449	0,9415	0,9434	**0,9375**
GpositivePseAAC	**0,5741**	0,5648	0,5471	**0,5470**	0,5677	0,5659	0,5588	**0,5881**	0,5662	0,5616	0,5643	0,5652	0,5504
scene	0,5945	**0,6113**	0,5884	0,6022	**0,5879**	0,5958	0,5989	**0,6237**	0,5970	0,5913	0,6005	0,6112	0,5897
VirusGO	**0,8804**	0,9152	0,9155	**0,9244**	0,8954	0,8913	0,9128	**0,9310**	0,9165	0,9052	0,9208	0,9064	0,9171
VirusPseAAC	**0,2153**	0,3849	0,3312	0,3627	**0,4147**	0,3960	0,4105	**0,4385**	0,4047	0,2909	0,3671	0,2795	0,4097
Yelp	**0,6913**	**0,7521**	0,7472	0,7302	0,7385	0,7274	0,7505	**0,7521**	0,7359	0,7212	0,7436	0,7350	0,7459
Média	**0,6531**	**0,6839**	**0,6769**	**0,6797**	**0,6847**	**0,6830**	**0,6897**	**0,7081**	**0,6868**	**0,6650**	**0,6810**	**0,6673**	**0,6790**

Tabela 43 – Comparação pareada Micro-F1

	G_C	Lo_C	$H.A_{Mi}$	$H.A_S$	$H.B_{Mi}$	$H.B_S$	E_{Mi}	O_{Mi}	$R1_{Mi}$	$R1_S$	$R2_{Mi}$	$R2_S$	R3	Média
G_C	0	3	3	2	2	1	1	0	2	3	2	3	5	**2,5**
Lo_C	5	0	5	5	4	4	5	0	3	7	5	6	5	**4,9**
$H.A_{Mi}$	5	3	0	4	5	3	2	0	2	5	3	5	4	**3,7**
$H.A_S$	6	3	4	0	4	3	3	0	3	5	5	4	5	**4,1**
$H.B_{Mi}$	6	4	3	4	0	4	3	0	3	4	4	5	5	**4,1**
$H.B_S$	7	4	5	4	4	0	3	0	1	6	4	4	5	**4,3**
E_{Mi}	7	3	6	5	5	5	0	0	5	7	5	6	7	**5,5**
O_{Mi}	8	8	8	8	8	8	8	0	8	8	8	8	8	**8,0**
$R1_{Mi}$	6	5	6	5	5	7	3	0	0	8	4	7	6	**5,6**
$R1_S$	5	1	3	3	4	2	1	0	0	0	2	3	4	**2,5**
$R2_{Mi}$	6	3	5	3	4	4	3	0	4	6	0	6	6	**4,5**
$R2_S$	5	2	3	4	3	4	2	0	1	5	2	0	4	**3,2**
R3	3	3	4	3	3	3	1	0	2	4	2	4	0	**2,9**
Média	**5,8**	**3,5**	**4,6**	**4,2**	**4,3**	**4,0**	**2,9**	**0,0**	**2,8**	**5,7**	**3,8**	**5,1**	**5,3**	

Como esperado, o oráculo sempre gera as partições que levam à melhor Micro-F1. Considerando os demais métodos, a tabela também mostra que, em média, as partições exaustivas obtiveram os melhores resultados, seguida de perto pelas partições $R1_{Mi}$. As partições híbridas novamente levaram a resultados competitivos, obtendo os melhores

valores de Micro-F1 em três conjuntos de dados. Porém, como os testes estatísticos mostraram, as diferenças entre todos os resultados não são estatisticamente significativas.

A Tabela 43 apresenta uma comparação pareada entre todos os métodos, mostrando o número total de conjuntos de dados em que um método na linha teve melhor desempenho do que um método na coluna. Analisando os resultados obtidos ao usar partições híbridas e partições globais, observa-se novamente que partições de com grupos disjuntos de rótulos correlacionados podem melhorar o desempenho da classificação. As partições híbridas obtiveram melhores resultados na Micro-F1 em comparação com a abordagem global na grande maioria dos conjuntos de dados.

Ao comparar as partições híbridas com as partições locais, os resultados agora são muito mais competitivos do que os obtidos com a Macro-F1. Como a medida Micro-F1 não considera desempenhos em rótulos individuais, as partições híbridas levaram a melhores resultados que a abordagem local em cinco dos oito conjuntos de dados. Ainda que os conjuntos de dados tenham um pequeno número de rótulos, três dos métodos híbridos obtiveram valores médios de Micro-F1 muito próximos ou superiores à abordagem local. Isso sugere novamente os benefícios de agrupar rótulos correlacionados.

Os resultados dos métodos que geram partições aleatórias novamente suportam a afirmação de que os métodos tradicionais baseados em global e local ainda falham em lidar corretamente com as correlações entre rótulos. Os experimentos mostram novamente que partições aleatórias de rótulos podem levar a resultados competitivos ou até melhores, melhorando o desempenho dos classificadores multirrótulo. Neste caso, as partições híbridas obtiveram resultados altamente competitivos com as partições aleatórias, mostrando ser uma alternativa viável para gerar partições com grupos de rótulos correlacionados.

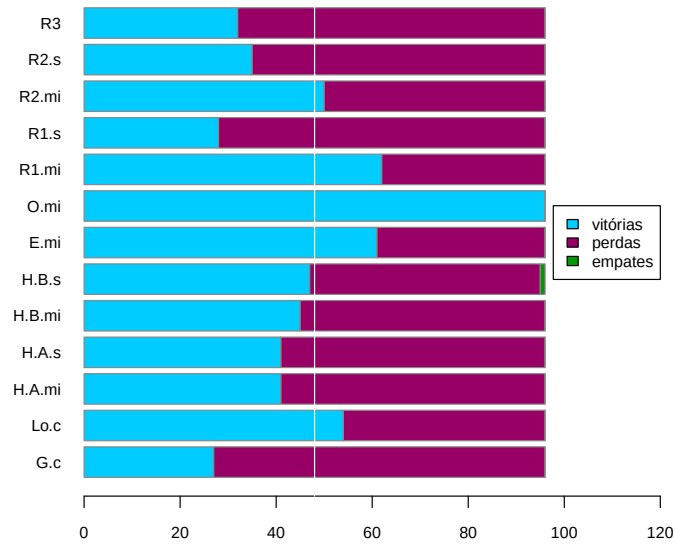


Figura 69 – Gráfico de Vitórias, Derrotas e Empates para Micro-F1. Oráculo tem o maior número de vitórias como é o esperado, enquanto que o $R1_S$ e G_C tem o menor. As partições híbridas, aleatórias e locais se mostram competitivas.

A Figura 69 apresenta o gráfico de vitórias, derrotas e empates para a medida Micro-F1. O cenário deste gráfico afirma novamente que partições oráculo sempre levam ao melhor resultado possível (96 vitórias), no entanto existem diferenças com relação à Macro-F1. Na faixa entre 65 e 5 vitórias estão os métodos $R1_{Mi}$, E_{Mi} e Lo_C, e neste contexto as partições locais não configuram a segunda posição de maior vencedor. Mais uma vez as partições globais ficam em último lugar (27), e próximas a elas as partições aleatórias, $R2_S$ (35), R3 (32) e $R1_S$(28). Na faixa média, entre 50 e 41 vitórias, encontram-se as partições híbridas e a $R2_{Mi}$. É possível notar, portanto, que nesta medida, todas as partições híbridas foram capazes de superar as partições globais e três dos cinco tipos de partições aleatórias. Tudo isto reforça o que já foi mencionado na análise da Macro-F1.

Da mesma forma que foi feita para a medida Macro-F1, a significância estatística dos resultados foi avaliada executando um teste de Friedman com nível de 95% ($\alpha = 0,05$), obtendo um p_Value de $0,189876014$ e ChiSquere igual a $33,2925824175824$. O teste post-hoc de Nemenyi foi então aplicado, obtendo-se um valor de diferença crítica de 6.060886776, que foi usado para construir o diagrama mostrado na Figura 69.

No caso da Micro-F1, existem algumas diferenças estatísticas entre alguns métodos. Nota-se dois grupos de métodos onde não há diferença estatística entre eles. O grupo à direita é composto pelas partições híbridas, global e dois métodos aleatórios, enquanto o grupo à esquerda é composto pela partição oráculo, exaustiva, local e três aleatórias. Apesar da comparação pareada ter mostrado que as partições locais ganharam menos vezes com relação às outras partições, a análise estatística confirma que elas ainda são as melhores que as outras.

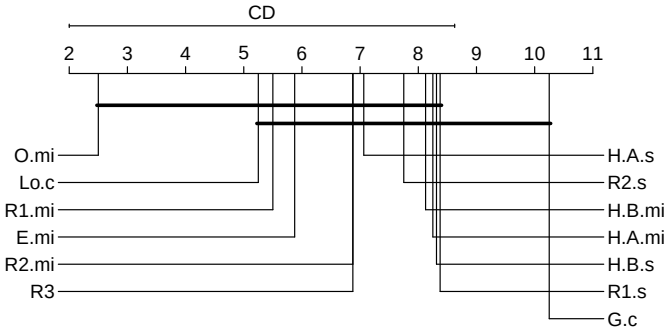


Figura 70 – Gráfico de Distância Crítica de Nemenyi para Micro-F1.

As partições aleatórias e híbridas obtiveram resultados competitivos, mas no caso da Micro-F1, estatisticamente três versões de partições aleatórias foram melhores que todas as versões de partições híbridas. Isto confirma novamente a suposição de que aprender grupos de rótulos aleatórios, ou aprender grupos de rótulos correlacionados, é melhor do que aprender um único grupo de rótulos (global). No entanto, o gráfico de distância crítica também mostra que as partições locais levam a melhores resultados.

6.2.3 Análise das Partições

As melhores partições oráculo, escolhidas de acordo com o critério de seleção Macro-F1 e Micro-F1, assim como o total de rótulos dentro de cada grupo, são apresentadas na Tabela 44. **Partição (P)** indica o número da partição[3], **Grupos (G)** indica a quantidade de grupos dentro da partição, ou seja, a partição P é composta por G grupos de rótulos, e **L** indica a quantidade de rótulos (labels) dentro de cada grupo. Por exemplo, a partição de número 37 foi escolhida como a melhor para o dataset emotions na Macro-F1, e essa partição é composta por três grupos de rótulos. A Tabela 45 apresenta a distribuição dos rótulos, dentro de cada grupo, em cada uma das melhores partições oráculo. Os números de 1 a 4 indicam o número do grupo o qual o rótulo pertence, e Ma-F1 e Mi-F1 referem-se à O_{Ma} e O_{Mi}.

Tabela 44 – Melhores Partições Oráculo

	$\mathbf{O}_{Ma}$			$\mathbf{O}_{Mi}$		
Dataset	**P**	**G**	**L**	**P**	**G**	**L**
emotions	37	3	G1 = 4 G2 = 1 G3 = 1	37	3	G1 = 4 G2 = 1 G3 = 1
flags	69	2	G1 = 4 G2 = 3	466	4	G1 = 3 G2 = 2 G3 = 1 G4 = 1
GpositiveGO	3	2	G1 = 3 G2 = 1	12	3	G1 = 2 G2 = 1 G3 = 1
GpositivePseAAC	8	2	G1 = 2 G2 = 2	8	2	G1 = 2 G2 = 2
scene	27	3	G1 = 4 G2 = 1 G3 = 1	27	3	G1 = 4 G2 = 1 G3 = 1
VirusGO	148	4	G1 = 2 G2 = 2 G3 = 1 G4 = 1	148	4	G1 = 2 G2 = 2 G3 = 1 G4 = 1
VirusPseAAC	199	5	G1 = 2 G2 = 1 G3 = 1 G4 = 1 G5 = 1	107	3	G1 = 3 G2 = 2 G3 = 1
Yelp	52	5	G1 = 1 G2 = 1 G3 = 1 G4 = 1 G5 = 1	52	5	G1 = 1 G2 = 1 G3 = 1 G4 = 1 G5 = 1

Para os conjuntos de dados emotions, GpositivePseAAC, scene, VirusGO e Yelp, a quantidade de grupos de rótulos e a quantidade total de rótulos em cada grupo é igual na Macro-F1 e Micro-F1, ou seja, exatamente a mesma partição. Já nos conjuntos de dados flags, GpositiveGO e VirusPseAAC as partições são diferentes em cada critério. O fato de cinco dos oito conjuntos de dados terem uma configuração idêntica, pode indicar que o classificador terá um melhor aprendizado com uma partição que tenha esta configuração.

[3] Isto foi explicado no capítulo de metodologia e montagem de experimentos.

Tabela 45 – Distribuição dos rótulos nas melhores partições oráculo

Emotions			GpositiveGO			Scene			VirusGO		
Label	**Ma-F1**	**Mi-F1**	**Label**	**Ma-F1**	**Mi-F1**	**Label**	**Ma-F1**	**Mi-F1**	**Label**	**Ma-F1**	**Mi-F1**
relaxing.calm	1	1	*Label1*	1	2	*Beach*	1	1	*Label1*	1	1
quiet.still	1	1	*Label2*	1	1	*FallFoliage*	2	2	*Label2*	3	3
sad.lonely	1	1	*Label3*	1	3	*Field*	1	1	*Label3*	4	4
angry.aggresive	1	1	*Label4*	2	1	*Mountain*	3	3	*Label4*	2	2
amazed.suprised	2	2				*Sunset*	1	1	*Label5*	2	2
happy.pleased	3	3				*Urban*	1	1	*Label6*	1	1
Yelp			**GpositivePseAAC**			**FLAGS**			**VirusPseAAC**		
Label	**Ma-F1**	**Mi-F1**	**Label**	**Ma-F1**	**Mi-F1**	**Label**	**Ma-F1**	**Mi-F1**	**Label**	**Ma-F1**	**Mi-F1**
IsAmbianceGood	3	3	*Label1*	1	1	*black*	1	1	*Label1*	2	3
IsDealsGood	4	4	*Label2*	2	2	*blue*	2	4	*Label2*	3	2
IsFoodGood	1	1	*Label3*	1	1	*green*	2	3	*Label3*	1	2
IsPriceGood	5	5	*Label4*	2	2	*orange*	2	2	*Label4*	4	1
IsServiceGood	2	2				*red*	1	1	*Label5*	1	1
						white	1	2	*Label6*	5	1
						yellow	1	1			

Relembre que todas as partições possíveis (número de bell) foram testadas para o oráculo e dentre essas partições, várias têm a mesma quantidade de grupos, mas a distribuição dos rótulos dentro de cada uma é diferente. A Tabela 46 apresenta um resumo de todas as partições geradas e que foram utilizadas nos experimentos exaustivo e oráculo. Alguns nomes de conjuntos de dados foram encurtados para que a Tabela pudesse ser devidamente apresentada. A coluna GRUPOS indica o número de grupos de rótulos da partição, portanto, 1 = 1 grupo, 2 = 2 grupos, 3 = 3 grupos, 5 = 5 grupos, 6 = 6 grupos e 7 = 7 grupos. Em seguida, na primeira coluna em cada dataset, consta o total de partições que foram geradas com esse número de grupos. Exemplificando, para o dataset emotions foi gerada uma partição com um único grupo, 31 partições com 2 grupos de rótulos, 90 partições com 3 grupos, 65 partições com quatro grupos, 15 partições com 5 grupos e 1 partição com 6 grupos.

Tabela 46 – Sumário de todas as partições possíveis.

Grupos	**emotions**		**flags**		**GposGO**		**GposPAC**		**scene**		**VirusGO**		**VirusPAC**		**Yelp**	
1	1	0%	1	0%	1	7%	1	7%	1	0%	1	0%	1	0%	1	2%
2	31	15%	63	7%	**7**	**47%**	**7**	**47%**	31	15%	31	15%	31	15%	15	29%
3	**90**	**44%**	301	34%	6	40%	6	40%	**90**	**44%**	**90**	**44%**	**90**	**44%**	**25**	**48%**
4	65	32%	**350**	**40%**	1	7%	1	7%	65	32%	65	32%	65	32%	10	19%
5	15	7%	140	16%					15	7%	15	7%	15	7%	1	2%
6	1	0%	21	2%					1	0%	1	0%	1	0%		
7			1	0%												
Total	**203**		**877**		**15**		**15**		**203**		**203**		**203**		**52**	

A segunda coluna em cada dataset, corresponde à porcentagem (por questões de espaço o valor foi arrendondado), enquanto que a última linha da tabela apresenta o número total de partições geradas. Portanto, deve-se interpretar a segunda coluna da seguinte forma: foram geradas 203 partições para o dataset emotions e destas, 15% é composta de 2 grupos, enquanto que 44% delas é composta por 3 grupos de rótulos. Os números marcados de azul na tabela correspondem ao maior número de partições geradas com a mesma quantidade de grupos. Assim, no dataset Yelp, 48% de todas as partições geradas são compostas de 3 grupos de rótulos.

As Tabelas de 47 até 49 apresentam as partições escolhidas em cada um dos métodos testados neste experimento para cada um dos 10 folds e em cada dataset. Considerando que temos 15 métodos, 8 conjuntos de dados e 10 folds, existe um total de 1200 escolhas. Computando a frequência de escolha das partições, chega-se aos seguintes valores: 40% das partições escolhidas são compostas de 2 grupos, 27% de partições com 3 grupos, 18% de partições com 4 grupos, 13% de partições com 5 grupos, 2% de partições com 6 grupos. Isso mostra coerência entre as partições escolhidas pelo oráculo, híbridas e aleatórias.

Estas informações podem ser melhor visualizadas na Tabela 50, que apresenta o número de grupos da partição mais selecionada para cada dataset, cada método e fold. Nesta tabela, os números de 1 a 5 indicam o número de grupos da partição. Exemplificando, no dataset flags, no método E_{Ma}, partições com 3 e 4 grupos foram escolhidas 50% das vezes cada uma, enquanto que no HB_{Ma}, uma partição com 3 grupos foi escolhida mais vezes do que todas as outras partições.

É interessante notar que, em alguns métodos aleatórios o número de grupos da melhor partição escolhida muda com relação aos métodos determinísticos. É o caso do dataset emotions em que a melhor partição escolhida pelo O_{Mi} e O_{Ma} foi uma composta por três grupos de rótulos. Nos métodos HA_{Ma}, HA_{Mi}, HA_S, HB_{Ma}, HB_{Mi}, HB_S, E_{Ma}, e E_{Mi} as melhores partições escolhidas com maior frequência são compostas ou por 2, ou por 3 grupos, semelhante ao oráculo. Mas este não é o caso de $R1_{Ma}$, $R1_{Mi}$ e R3, que escolheram por mais vezes uma partição composta por 5 grupos.

Uma partição com 5 grupos é mais próxima de uma partição local do que uma global, já que apenas dois rótulos estão agrupados juntos e o restante estão em grupos separados. É possível que, nos folds destes conjuntos de dados, essa distribuição aleatória dos rótulos nas partições tenha influenciado o aprendizado dos grupos, levando a um melhor desempenho preditivo do que uma partição com 2 ou 3 grupos.

Ainda vale ressaltar que este é o número de grupos, dentro da partição, escolhido com maior frequência, não significa que todos os folds escolheram exatamente a mesma partição. Isto pode ser verificado nas tabelas de partições escolhidas. Dessa forma, o fato de em diferentes folds terem sido escolhidas diferentes partições, mostra que para aquela porção dos dados, aquela partição em particular é a que melhor ajuda o classificador a aprender os rótulos e também obter um melhor desempenho.

Há também casos em que exatamente o mesmo número de grupos escolhido dos métodos híbridos e aleatórios bateram com o oráculo. Estes matches estão marcados de verde na tabela. Isso indica que, para alguns conjuntos de dados, os métodos HPML.A, HPML.B e também alguns aleatórios foram capazes de obter a melhor partição (de acordo com o oráculo) na maior parte dos folds. Portanto, as partições híbridas resultantes dos métodos aqui apresentados não estão tão distantes da partição ótima.

Os métodos que utilizaram a silhueta como critério de escolha da melhor partição também tiveram selecionados mais vezes partições próximas das melhores partições oráculo

Tabela 47 – Partições Escolhidas Parte 1

E_{Ma}

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**
1	154	4	246	3	3	2	8	2	55	3	2	2	201	5	35	3
2	21	2	597	4	3	2	14	3	140	3	5	2	138	3	52	5
3	7	2	409	4	14	3	8	2	67	3	8	2	186	4	45	4
4	99	3	158	3	5	2	8	2	61	3	31	3	103	3	38	3
5	7	2	17	2	14	3	7	2	161	4	52	3	151	4	45	4
6	118	4	47	3	6	2	13	3	44	2	9	2	197	5	52	5
7	7	2	120	3	2	2	7	2	44	2	12	2	103	3	52	5
8	158	4	583	4	3	2	4	2	153	4	6	2	150	4	45	4
9	37	3	79	2	3	2	12	3	59	3	10	2	81	3	45	4
10	82	3	523	4	8	2	12	3	160	4	13	2	123	4	42	4

E_{Mi}

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**
1	76	3	246	3	3	2	4	2	55	3	2	2	202	5	52	5
2	21	2	393	3	3	2	14	3	128	3	5	2	39	2	52	5
3	18	2	402	4	14	3	5	2	67	3	8	2	202	5	52	5
4	99	3	346	3	5	2	4	2	61	3	31	3	201	5	29	3
5	7	2	52	2	14	3	9	3	5	2	52	3	42	2	52	5
6	118	4	152	3	5	2	7	2	137	3	9	2	200	5	52	5
7	7	2	96	3	2	2	14	3	44	2	12	2	106	3	52	5
8	158	4	475	4	12	3	4	2	154	4	6	2	201	5	45	4
9	187	4	79	2	3	2	2	2	129	3	10	2	67	3	43	4
10	82	3	410	4	8	2	10	3	160	4	13	2	123	4	45	4

HPML.A_{Ma}

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**
1	5	5	5	5	2	2	2	2	5	5	2	2	5	5	4	4
2	3	3	3	3	2	2	3	3	2	2	4	4	3	3	4	4
3	2	2	6	6	2	2	3	3	5	5	2	2	3	3	4	4
4	3	3	5	5	3	3	2	2	5	5	4	4	5	5	4	4
5	2	2	2	2	2	2	2	2	4	4	3	3	4	4	4	4
6	2	2	3	3	2	2	3	3	5	5	3	3	5	5	4	4
7	5	5	3	3	3	3	2	2	4	4	5	5	2	2	4	4
8	4	4	6	6	2	2	2	2	5	5	2	2	5	5	4	4
9	5	5	2	2	2	2	2	2	2	2	4	4	5	5	4	4
10	2	2	6	6	2	2	2	2	5	5	5	5	5	5	4	4

HPML.A_{Mi}

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**
1	2	2	5	5	2	2	2	2	5	5	2	2	5	5	4	4
2	3	3	3	3	2	2	3	3	2	2	4	4	3	3	4	4
3	2	2	6	6	2	2	2	2	5	5	2	2	3	3	4	4
4	4	4	6	6	3	3	2	2	4	4	4	4	5	5	4	4
5	2	2	2	2	2	2	2	2	4	4	3	3	4	4	4	4
6	5	5	3	3	2	2	3	3	5	5	3	3	5	5	4	4
7	3	3	3	3	3	3	2	2	4	4	5	5	2	2	4	4
8	4	4	6	6	2	2	2	2	4	4	4	4	5	5	4	4
9	5	5	2	2	3	3	2	2	2	2	4	4	2	2	4	4
10	2	2	5	5	2	2	2	2	5	5	5	5	2	2	4	4

HPML.A_{S}

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**
1	2	2	5	2	2	2	2	2	4	4	4	4	4	4	3	3
2	2	2	5	2	2	2	2	2	4	4	4	4	4	4	3	2
3	2	2	5	2	2	2	2	2	4	4	4	4	4	4	3	3
4	2	2	5	4	2	2	2	2	4	4	4	4	4	4	3	3
5	2	2	5	3	2	2	2	2	4	4	4	4	4	4	3	3
6	2	2	5	3	2	2	2	2	4	4	4	4	4	4	3	3
7	2	2	5	3	2	2	2	2	4	4	4	4	4	4	3	3
8	2	2	5	4	2	2	2	2	4	4	4	4	4	4	3	3
9	2	2	5	3	2	2	2	2	4	4	4	4	4	4	3	3
10	2	2	5	4	2	2	2	2	4	4	4	4	4	4	3	3

Tabela 48 – Partições Escolhidas Parte 2

HPML.B$_{Ma}$

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	3	3	5	2	2	2	2	2	3	3	2	2	2	2	3	3
2	2	2	10	3	2	2	2	2	3	3	3	3	3	3	2	2
3	2	2	8	5	2	2	2	2	3	3	2	2	3	3	5	5
4	3	3	4	3	2	2	3	3	2	2	2	2	2	2	3	3
5	4	4	2	2	2	2	2	2	3	3	2	2	3	3	4	4
6	2	2	5	3	2	2	3	3	3	3	2	2	2	2	6	5
7	6	3	3	3	3	3	2	2	3	3	2	2	2	2	3	3
8	6	3	6	5	3	3	3	3	3	3	2	2	3	3	8	5
9	5	3	2	2	3	3	3	3	2	2	2	2	3	3	8	4
10	3	3	4	4	2	2	3	3	3	3	3	3	2	2	8	5

HPML.B$_{Mi}$

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	2	2	5	2	2	2	2	2	3	3	2	2	2	2	3	3
2	2	2	10	3	2	2	2	2	3	3	3	3	3	3	5	4
3	2	2	8	5	2	2	2	2	3	3	2	2	2	2	5	5
4	3	3	4	3	2	2	2	2	2	2	2	2	2	2	3	3
5	4	4	2	2	2	2	2	2	3	3	2	2	3	3	4	4
6	2	2	5	3	2	2	3	3	2	2	2	2	2	2	6	5
7	3	3	2	2	3	3	2	2	3	3	3	3	3	3	3	3
8	6	3	15	5	3	3	3	3	3	3	2	2	3	3	8	5
9	5	3	4	3	3	3	3	3	2	2	2	2	3	3	8	4
10	3	3	4	4	2	2	3	3	3	3	3	3	2	2	8	5

HPML.B$_{S}$

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	3	3
2	3	3	4	2	2	2	2	2	2	2	2	2	2	2	4	3
3	3	3	12	3	2	2	2	2	2	2	2	2	2	2	2	2
4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	7	4
5	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
6	2	2	2	2	2	2	2	2	2	2	2	2	2	2	3	3
7	7	3	11	4	2	2	2	2	2	2	2	2	2	2	2	2
8	3	3	12	3	2	2	2	2	2	2	2	2	2	2	7	4
9	2	2	11	5	2	2	2	2	2	2	2	2	2	2	2	2
10	3	3	2	2	2	2	2	2	2	2	2	2	2	2	5	4

R1$_{Ma}$

	Emotions		Flags		GpositiveGO		GposPAC		Scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	3	3	6	6	2	2	2	2	2	2	3	3	4	4	4	4
2	2	2	2	2	3	3	3	3	2	2	3	3	4	4	4	4
3	5	5	4	4	2	2	2	2	4	4	2	2	5	5	4	4
4	4	4	5	5	3	3	2	2	4	4	3	3	4	4	4	4
5	2	2	3	3	2	2	2	2	4	4	3	3	2	2	3	3
6	4	4	5	5	2	2	2	2	2	2	5	5	5	5	4	4
7	5	5	5	5	2	2	2	2	4	4	5	5	3	3	4	4
8	5	5	6	6	2	2	2	2	4	4	3	3	5	5	4	4
9	5	5	2	2	3	3	2	2	5	5	3	3	4	4	3	3
10	2	2	5	5	2	2	3	3	3	3	5	5	3	3	4	4

R1$_{Mi}$

	emotions		flags		GposGO		GposPAC		scene		VirusGO		VirusPAC		Yelp	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	2	2	6	6	2	2	2	2	2	2	3	3	3	3	4	4
2	2	2	5	5	3	3	3	3	2	2	3	3	4	4	4	4
3	2	2	4	4	2	2	2	2	4	4	2	2	3	3	4	4
4	4	4	5	5	3	3	2	2	4	4	3	3	5	5	4	4
5	2	2	3	3	2	2	2	2	4	4	2	2	5	5	3	3
6	4	4	2	2	2	2	2	2	2	2	5	5	4	4	4	4
7	5	5	5	5	2	2	2	2	4	4	5	5	2	2	4	4
8	2	2	6	6	2	2	2	2	4	4	3	3	4	4	4	4
9	5	5	5	5	2	2	2	2	5	5	3	3	2	2	3	3
10	2	2	5	5	2	2	2	2	3	3	5	5	4	4	4	4

Tabela 49 – Partições Escolhidas Parte 3

R1$_S$																
	Emotions		**Flags**		**GposGO**		**GposPAC**		**Scene**		**VirusGO**		**VirusPAC**		**Yelp**	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	2	2	2	2	2	2	2	2	4	4	2	2	3	3	3	3
2	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	2	1	2	2	2	2	3	3	2	2	2	2	3	3
4	3	3	2	2	2	2	2	2	2	2	2	2	2	2	2	2
5	2	2	4	3	2	2	2	2	2	2	2	2	4	4	3	3
6	3	3	2	2	2	2	2	2	2	2	2	2	2	2	2	2
7	4	3	2	2	2	2	2	2	4	4	4	4	2	2	2	2
8	3	3	2	2	2	2	2	2	2	2	4	4	2	2	3	3
9	2	2	3	2	2	2	2	2	4	4	2	2	2	2	2	2
10	2	2	5	4	2	2	2	2	3	3	4	4	2	2	3	2

R2$_{Ma}$																
	Emotions		**Flags**		**GposGO**		**GposPAC**		**Scene**		**VirusGO**		**VirusPAC**		**Yelp**	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	3	4	4	4	2	2	2	2	3	4	2	4	2	5	2	4
2	5	2	3	5	2	3	2	3	5	5	2	5	3	5	3	3
3	4	3	2	6	2	2	2	3	5	4	2	3	4	5	2	4
4	3	2	2	4	3	3	2	2	5	5	3	5	4	5	4	4
5	5	2	2	6	2	2	3	2	5	5	2	5	2	5	4	4
6	5	5	3	3	3	3	2	2	4	2	2	5	4	5	4	3
7	3	2	3	6	3	3	2	2	3	5	3	5	4	5	3	4
8	4	4	6	4	3	2	2	3	3	4	3	5	4	4	4	4
9	5	5	6	2	2	3	3	3	5	2	3	5	3	3	4	3
10	2	5	3	6	2	3	3	3	5	4	2	2	3	3	2	4

R2$_{Mi}$																
	Emotions		**Flags**		**GposGO**		**GposPAC**		**Scene**		**VirusGO**		**VirusPAC**		**Yelp**	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	3	4	4	4	2	2	2	2	3	4	2	4	4	4	2	4
2	5	2	3	5	2	2	2	3	5	3	2	5	5	4	3	3
3	2	4	2	6	2	2	2	3	5	5	2	3	4	5	2	3
4	3	2	2	4	3	3	2	2	5	4	3	5	4	5	2	4
5	5	2	3	5	2	2	2	3	5	5	2	5	4	4	4	4
6	4	4	3	3	3	3	2	2	4	2	2	5	4	5	4	3
7	3	2	5	2	3	3	2	2	2	2	3	5	4	5	3	4
8	4	4	6	4	3	3	2	3	3	4	3	5	4	4	4	4
9	2	2	6	2	2	2	3	3	5	2	3	5	5	2	4	3
10	3	4	3	6	2	2	3	3	5	4	2	2	2	3	2	4

R2$_S$																
	Emotions		**Flags**		**GposGO**		**GposPAC**		**Scene**		**VirusGO**		**VirusPAC**		**Yelp**	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1	2	5	6	2	3	3	2	2	4	2	5	5	2	5	4	2
2	2	3	2	6	2	2	3	3	4	4	3	5	2	3	4	2
3	3	4	4	2	2	2	2	3	5	5	5	4	5	2	3	3
4	3	2	5	2	3	3	2	2	5	4	2	5	5	5	3	2
5	3	5	6	6	3	3	3	2	3	2	3	2	5	5	2	3
6	3	3	6	2	2	2	2	2	2	2	4	4	3	2	2	2
7	2	3	5	2	3	3	3	3	4	5	3	5	4	5	4	2
8	5	5	5	2	3	3	2	3	3	4	4	2	5	3	4	4
9	2	2	4	3	3	3	3	3	2	3	2	2	4	2	2	2
10	5	4	4	2	2	2	3	3	3	3	2	2	2	3	2	4

R3																
	Emotions		**Flags**		**GposGO**		**GposPAC**		**Scene**		**VirusGO**		**VirusPAC**		**Yelp**	
F	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**G**	**P**	**C**	**P**	**G**	**P**	**G**	**P**	**G**
1		5		6		2		2		3		5		4		3
2		5		4		3		3		2		5		5		2
3		3		6		3		3		3		4		4		4
4		5		4		2		3		5		2		4		3
5		3		2		2		3		5		4		2		3
6		4		5		3		2		3		2		2		2
7		5		5		2		3		4		4		3		4
8		5		3		3		3		2		3		4		2
9		2		6		3		3		2		3		4		4
10		5		4		2		2		3		2		3		4

Tabela 50 – Número de grupos, dentro da partição, mais escolhido nos 10 folds.

	$\mathbf{O}_{Ma}$	$\mathbf{O}_{Mi}$	$\mathbf{E}_{Ma}$	$\mathbf{E}_{Mi}$	$\mathbf{H.A}_{Ma}$	$\mathbf{H.A}_{Mi}$	$\mathbf{H.A}_{S}$	$\mathbf{H.A}_{Ma}$	$\mathbf{H.B}_{Mi}$	$\mathbf{H.B}_{S}$
emotions	3	3	2	2	2	2	2	2	3	2 e 3
flags	2	4	3 e 4	3	3 e 6	3 e 6	5	3	3	2
GpositiveGO	2	3	2	2	2	2	2	2	2	2
GpositivePseAAc	2	2	2	2	2	2	2	2 e 3	2	2
scene	3	3	3	3	5	4 e 5	4	3	3	2
VirusGO	4	4	2	2	2 e 4	4	4	2	2	2
VirusPseAAC	5	3	3 e 4	5	5	5	4	2 e 3	2 e 3	2
Yelp	5	5	4	5	4	4	3	5	5	2

	$\mathbf{O}_{Ma}$	$\mathbf{O}_{Mi}$	$\mathbf{R1}_{Ma}$	$\mathbf{R1}_{Mi}$	$\mathbf{R1}_{S}$	$\mathbf{R2}_{Ma}$	$\mathbf{R2}_{Mi}$	$\mathbf{R2}_{S}$	**R3**
emotions	3	3	5	5	3	2	2 e 5	3 e 5	5
flags	2	4	5	5	2	6	4	2	4 e 6
GpositiveGO	2	3	2	2	2	3	3	3	2 e 3
GpositivePseAAc	2	2	2	2	2	2 e 3	3	3	3
scene	3	3	4	4	2	4 e 5	5	2 e 4	3
VirusGO	4	4	3	3	2	5	5	2 e 5	2 e 4
VirusPseAAC	5	3	4	4	2	5	4 e 5	5	4
Yelp	5	5	4	4	2 e 3	4	4	2	4

em alguns conjuntos de dados. No dataset GpositivePseAAC, nos métodos HA_S, HB_S e R1_S uma partição com dois grupos foi escolhida mais vezes do que as outras, assim como no oráculo.

Outro fato interessante notado é que a melhor partição híbrida escolhida pelo oráculo, em alguns casos, está dentro da maior porcentagem na Tabela 46. No dataset emotions 44% das partições geradas possuem três grupos e, as melhores partições oráculo para este dataset são compostas por três grupos. Esse mesmo comportamento é observado nos conjuntos de dados GpositivePseAAC e scene. No dataset flags, 40% da partições geradas são compostas por 4 grupos e no O_{Mi} também, se repetindo no VirusPseACC e GpositiveGO.

As exceções são o VirusGO e o Yelp, em que a maior parte das partições geradas têm 3 grupos, mas a melhor partição encontrada pelo oráculo não tem 3 grupos. No caso do VirusGO, a melhor partição oraculo é composta por 4 grupos, e 32% de todas as partições geradas para este dataset são compostas dessa forma. Já o caso do Yelp é bem peculiar, a melhor partição oraculo é composta por 5 grupos e esta configuração de partição compõe apenas 2% de todas as partições geradas.

Além disso, com 40% de todas as partições escolhidas sendo compostas de 2 grupos, também seria possível concluir que esta configuração é adequada para esta amostra de conjuntos de dados e os métodos híbridos, mas há ressalvas. Uma partição como esta é mais próxima de uma partição global do que uma local e isto pode ser refletido no desempenho. No geral, as partições híbridas têm melhor desempenho preditivo que as globais, mas perdem para as locais em alguns casos. Pode-se concluir disto que uma partição próxima da global não seria interessante, já que o resultados do método global são os piores. Como bem demonstrado, a melhor partição oráculo para cada dataset nem sempre é composta por 2 grupos e, portanto, não se pode afirmar que uma partição com 2 grupos é a melhor para todos os casos, mas diante dos resultados e análise apresentados,

pode-se dizer que é a mais próxima da melhor.

Considerando as informações apresentadas, e com esta amostra de conjuntos de dados, pode-se concluir e generalizar que os métodos híbridos aqui propostos são capazes de escolher uma partição híbrida com uma configuração igual ou próxima da melhor partição de um dataset.

6.3 Comunidades

Neste experimento foi utilizado o HPML.C conforme já explicado no Capítulo 5. As Tabelas 52, 53 e 51 apresentam os resultados do desempenho preditivo para as medidas de avaliação multirrótulo CLP, MLP, WLP, Macro-F1, Macro-Revocação, Macro-Precisão, Macro-F1, Macro-Revocação e Macro-Precisão. Os melhores valores estão marcados na tabela na cor azul, e os piores na cor vermelha. A última linha da tabela apresenta a média dos 20 conjuntos de dados multirrótulo. Por questões de espaço, os nomes dos métodos nas tabelas e figuras foram simplificados[4].

Os grupos das partições encontrados pelos métodos de detecção de comunidade para cada conjunto de dados foram semelhantes e às vezes os mesmos e por isto, alguns valores de desempenho são idênticos para alguns métodos e conjuntos de dados. Ao que tudo indica, o HPML.C não é significativamente afetado pela esparsificação com k-NN. Observou-se também, de forma geral, que os resultados dos grafos construídos com o índice de Jaccard e os métodos hierárquicos superaram aqueles com a similaridade de Rogers-Tanimoto, enquanto nos métodos não hierárquicos, com a similaridade Rogers-Tanimoto, em geral foram melhores que o índice de Jaccard. Em vários conjuntos de dados, as partições aleatórias superaram as partições globais. Ao comparar H.Ra e NH.Ra com Lo_C, as partições aleatórias levaram a resultados competitivos. Com relação aos métodos de esparsificação, k-NN obteve desempenho ligeiramente melhor quando comparado com a esparsificação com threshold.

Com este experimento chegou-se à conclusão de que o classificador utilizado não conseguiu usufruir do aprendizado das correlações. Trabalhos com conclusões semelhantes também podem ser encontrados nos seguintes estudos (LUACES et al., 2012; MELO; PAULHEIM, 2017; RIVOLLI et al., 2020; BASGALUPP et al., 2021) onde os autores mostraram que classificadores multirrótulo não conseguiram aprender as correlações e predizer corretamente os rótulos. Conclui-se que não há grande melhoria na performance independente do particionamento utilizado. De acordo com os resultados obtidos, isso provavelmente ocorreu porque a maioria dos rótulos não foi aprendido pelo classificador, mesmo pelas abordagens tradicionais. Os resultados mostram que as abordagens

[4] Reforçando, os resultados completos para todos os experimentos conduzidos, incluindo pdfs com estatísticas básicas, estão disponíveis no repositório oficial da tese: <https://github.com/cissagatto/HPML>

locais e globais tradicionais podem não estar aprendendo corretamente as correlações entre rótulos, uma vez que as partições geradas aleatoriamente levaram a resultados muito semelhantes em vários casos.

Os resultados da medidas CLP, MLP e WLP suportam as conclusões. Na medida MLP, que calcula a proporção de rótulos que nunca são preditos, pode-se notar altos valores indicando um alto erro de predição. Na média dos 20 conjuntos de dados, entre 38% e 50% dos rótulos não foram preditos. O método global é o pior caso, onde 58% dos rótulos não foram preditos corretamente pelo classificador. Já o melhor caso é o método Local, em que 38% dos rótulos não foi predito, uma diferença de 20% com relação ao Global. O dataset derisi é o pior caso, com mais de 99% dos rótulos não preditos no método Global. Os melhores casos são os conjuntos de dados Yelp e Scene, que obtiveram valor igual a zero para MLP em todos os métodos, indicando que não houve erro na predição dos rótulos.

Já na medida WLP, que mede quando o rótulo pode ser predito para algumas instâncias, mas essas predições estão sempre erradas, na média dos 20 conjuntos de dados, o pior caso fica com o Global (58%) e o melhor caso com o Local (44%). Ainda assim, um valor entre 58% e 44% de erro é alto, indicando que muitos rótulos foram preditos erroneamente. Os conjuntos de dados cellcycle e derisi são os piores casos, com mais de 99% dos rótulos preditos errados e, novamente o Yelp e o scene são os melhores casos. Interessante notar que os conjuntos de dados Yelp e scene obtiveram 0% de erro nessas três medidas em todos os métodos. Isto pode estar relacionado com as características desses conjuntos de dados, em que o TCS (dificuldade de aprendizado) não é tão alto comparado com os outros e o ULD (nível de dependência dos rótulos) é similar entre eles.

Na CLP, e na média dos 20 conjuntos de dados, nota-se que a maioria dos métodos obteve um valor de desempenho próximo à $0,03$, indicando que o classificador não errou tanto. O pior caso é o dataset GpositiveGO no método Global, onde o erro chega a 97%, isto é, o classificador previu o mesmo rótulo para todas as instâncias 97% das vezes.

As tabelas de comparação pareada e os gráficos de vitórias-derrotas-empates dão suporte às conclusões e insights reportados. As Tabelas de 54 à 58 apresentam os resultados da comparação pareada, que apresenta o total de conjuntos de dados em que um método da linha teve melhor desempenho que o método da coluna. Valores em azul nas tabelas indicam o número máximo de conjuntos de dados onde um método foi melhor que outro, enquanto que vermelho o mínimo. Dessa forma, na medida WLP, o método local obteve melhores resultados de desempenho preditivo do que todos os outros métodos em uma média de 14,2 conjuntos de dados, enquanto o método global em apenas 2,2 conjuntos de dados. Os métodos NH.Ra, H.J.K1, H.J.K2 e H.J.K3 empataram com 8,9 conjuntos de dados na média, provando a competitividade entre eles.

De forma geral, todas as tabelas de comparação pareada apresentam um comportamento similar, menos a Micro-Precisão. Nesta medida constata-se que os resultados das

Tabela 51 – Resultados CLP, MLP e WLP

CLP

conjuntos de dados	Lo	G	Ra.H	Ra.NH	JHK1	RHK1	JHK2	RHK2	JHK3	RHK3	JHT0	RHT0	JHT1	RHT1	JNHK1	RNHK1	JNHK2	RNHK2	JNHK3	RNHK3	JNHT0	RNHT0	JNHT1	RNHT1
birds	0,0000	0,0526	0,0000	0,0000	0,0000	0,0105	0,0000	0,0105	0,0000	0,0105	0,0000	0,0105	0,0000	0,0105	0,0000	0,0000	0,0000	0,0000	0,0053	0,0053	0,0000	0,0000	0,0000	0,0000
cal500	0,0069	0,0466	0,0069	0,0420	0,0431	0,0172	0,0431	0,0172	0,0431	0,0172	0,0431	0,0431	0,0431	0,0431	0,0460	0,0420	0,0477	0,0414	0,0448	0,0425	0,0460	0,0402	0,0460	0,0425
cellcycle	0,0000	0,0056	0,0056	0,0000	0,0039	0,0034	0,0039	0,0034	0,0039	0,0034	0,0039	0,0034	0,0039	0,0034	0,0034	0,0006	0,0017	0,0006	0,0022	0,0000	0,0011	0,0006	0,0017	0,0000
derisi	0,0000	0,0056	0,0056	0,0000	0,0034	0,0011	0,0034	0,0011	0,0034	0,0011	0,0034	0,0011	0,0034	0,0011	0,0022	0,0000	0,0028	0,0011	0,0028	0,0000	0,0051	0,0017	0,0051	0,0006
eisen	0,0000	0,0061	0,0061	0,0000	0,0061	0,0042	0,0061	0,0042	0,0061	0,0042	0,0061	0,0042	0,0061	0,0042	0,0030	0,0006	0,0036	0,0012	0,0036	0,0000	0,0024	0,0012	0,0036	0,0000
emotions	0,0000	0,0000	0,0000	0,0000	0,7147	0,0000	0,7147	0,0000	0,7147	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
EukaryotePseAAC	0,0000	0,0045	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0045	0,0000	0,0045	0,0000
flags	0,2000	0,2429	0,2571	0,1857	0,2429	0,2286	0,2429	0,2286	0,2429	0,2286	0,2429	0,2286	0,2429	0,2286	0,2429	0,2571	0,2000	0,2714	0,2286	0,2714	0,2000	0,2286	0,2000	0,2286
gasch1	0,0000	0,0056	0,0056	0,0000	0,0017	0,0051	0,0017	0,0051	0,0017	0,0051	0,0017	0,0006	0,0017	0,0006	0,0034	0,0006	0,0022	0,0006	0,0034	0,0017	0,0022	0,0006	0,0034	0,0006
GnegativeGO	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
GpositiveGO	0,9689	0,9618	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
langlog	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
medical	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
pheno	0,0000	0,1879	0,1879	0,1897	0,1873	0,1867	0,1873	0,1867	0,1873	0,1867	0,1873	0,1867	0,1873	0,1867	0,1879	0,1879	0,1879	0,1879	0,1879	0,1879	0,1879	0,1721	0,1879	0,2139
PlantGO	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0333	0,0000	0,0333	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
scene	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
seq	0,0000	0,0056	0,0000	0,0000	0,0028	0,0028	0,0028	0,0028	0,0028	0,0028	0,0028	0,0056	0,0028	0,0056	0,0017	0,0000	0,0022	0,0006	0,0034	0,0000	0,0017	0,0006	0,0017	0,0000
VirusPseAAC	0,0000	0,0500	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
yeast	0,0000	0,0714	0,0000	0,0500	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0429	0,0500	0,0643	0,0429	0,0429	0,0286	0,0286	0,0429	0,0286	0,0286
Yelp	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
AVERAGE	0,0588	0,0823	0,0237	0,0234	0,0603	0,0230	0,0603	0,0230	0,0603	0,0230	0,0246	0,0259	0,0246	0,0259	0,0267	0,0269	0,0256	0,0274	0,0262	0,0269	0,0240	0,0244	0,0241	0,0257

MLP

conjuntos de dados	Lo	G	Ra.H	Ra.NH	JHK1	RHK1	JHK2	RHK2	JHK3	RHK3	JHT0	RHT0	JHT1	RHT1	JNHK1	RNHK1	JNHK2	RNHK2	JNHK3	RNHK3	JNHT0	RNHT0	JNHT1	RNHT1
birds	0,3316	0,9474	0,3632	0,4684	0,3368	0,4842	0,3368	0,4842	0,3368	0,4842	0,3368	0,4842	0,3368	0,4842	0,4789	0,4105	0,4421	0,4316	0,5632	0,5421	0,5211	0,4368	0,5368	0,4789
cal500	0,5954	0,9534	0,5971	0,9259	0,9109	0,6931	0,9109	0,6931	0,9109	0,6931	0,9109	0,9132	0,9109	0,9132	0,9448	0,9374	0,9460	0,9391	0,9402	0,9356	0,9408	0,9339	0,9402	0,9316
cellcycle	0,8916	0,9944	0,9944	0,9843	0,9612	0,9500	0,9612	0,9500	0,9612	0,9500	0,9612	0,9500	0,9612	0,9500	0,9910	0,9848	0,9882	0,9860	0,9882	0,9831	0,9882	0,9848	0,9876	0,9820
derisi	0,9730	0,9944	0,9944	0,9888	0,9843	0,9753	0,9843	0,9753	0,9843	0,9753	0,9843	0,9753	0,9843	0,9753	0,9910	0,9888	0,9916	0,9899	0,9916	0,9888	0,9938	0,9904	0,9938	0,9893
eisen	0,8485	0,9939	0,9939	0,9848	0,9933	0,9485	0,9933	0,9485	0,9933	0,9485	0,9933	0,9485	0,9933	0,9485	0,9903	0,9842	0,9903	0,9879	0,9909	0,9830	0,9897	0,9873	0,9885	0,9855
emotions	0,0000	0,0333	0,0000	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0000	0,0000	0,0167	0,0000	0,0167	0,0000	0,0167	0,0000	0,0167	0,0000
EukaryotePseAAC	0,4045	0,8727	0,4455	0,6273	0,4455	0,4455	0,4455	0,4455	0,4455	0,4455	0,4455	0,4455	0,4455	0,4455	0,6273	0,6000	0,6091	0,7045	0,6955	0,6182	0,6909	0,6545	0,6727	0,7045
flags	0,1000	0,0714	0,0857	0,0857	0,0714	0,1143	0,0714	0,1143	0,0714	0,1143	0,0714	0,1143	0,0714	0,1143	0,0571	0,1143	0,1143	0,1429	0,0429	0,1143	0,1143	0,1143	0,0857	0,1143
gasch1	0,8006	0,9944	0,9944	0,9888	0,8994	0,9764	0,8994	0,9764	0,8994	0,9764	0,8994	0,8506	0,8994	0,8506	0,9921	0,9893	0,9910	0,9893	0,9921	0,9904	0,9910	0,9893	0,9921	0,9893
GnegativeGO	0,0250	0,1375	0,0250	0,0250	0,1375	0,0250	0,1375	0,0250	0,1375	0,0250	0,1375	0,0250	0,1375	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0500	0,0250
GpositiveGO	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0750	0,0000	0,0750	0,0000	0,0750	0,0000	0,0750	0,0000	0,0500	0,0000
langlog	0,0933	0,6573	0,1533	0,5093	0,1160	0,3173	0,1160	0,3173	0,1160	0,3173	0,1160	0,3173	0,1160	0,3173	0,5133	0,5480	0,5533	0,5707	0,5920	0,5933	0,5760	0,5547	0,5813	0,5680
medical	0,6089	0,9556	0,8444	0,6200	0,7689	0,7022	0,7689	0,7022	0,7689	0,7022	0,7689	0,7022	0,7689	0,7022	0,6711	0,7067	0,7444	0,7378	0,6356	0,7378	0,6378	0,7733	0,6444	0,7733
pheno	0,9770	0,6970	0,6970	0,6879	0,6697	0,6570	0,6697	0,6570	0,6697	0,6570	0,6697	0,6576	0,6697	0,6576	0,6939	0,6945	0,6939	0,6915	0,6927	0,6897	0,6927	0,6921	0,6903	0,7009
PlantGO	0,0667	0,4000	0,1583	0,1167	0,2917	0,5083	0,2917	0,5083	0,2917	0,5083	0,2917	0,7037	0,2917	0,7037	0,0833	0,1000	0,0917	0,0917	0,1417	0,1583	0,1250	0,3417	0,1333	0,3417
scene	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
seq	0,6534	0,9944	0,6534	0,9888	0,9270	0,9270	0,9270	0,9270	0,9270	0,9270	0,9270	0,9944	0,9270	0,9944	0,9904	0,9888	0,9904	0,9893	0,9921	0,9888	0,9904	0,9888	0,9904	0,9888
VirusPseAAC	0,1500	0,5833	0,4833	0,2667	0,2333	0,2333	0,2333	0,2333	0,2333	0,2333	0,2333	0,2333	0,2333	0,2333	0,2500	0,1833	0,2333	0,1833	0,2500	0,1500	0,3000	0,2833	0,2167	0,2833
yeast	0,0857	0,2286	0,0857	0,1929	0,1000	0,1286	0,1000	0,1286	0,1000	0,1286	0,1000	0,1286	0,1000	0,1286	0,2143	0,2000	0,1929	0,1714	0,1643	0,1714	0,1929	0,1500	0,1786	0,1786
Yelp	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
AVERAGE	0,3803	0,5755	0,4285	0,4739	0,4432	0,4551	0,4432	0,4551	0,4432	0,4551	0,4432	0,4730	0,4432	0,4730	0,4795	0,4728	0,4845	0,4816	0,4895	0,4835	0,4931	0,4950	0,4875	0,5018

WLP

conjuntos de dados	Lo	G	Ra.H	Ra.NH	JHK1	RHK1	JHK2	RHK2	JHK3	RHK3	JHT0	RHT0	JHT1	RHT1	JNHK1	RNHK1	JNHK2	RNHK2	JNHK3	RNHK3	JNHT0	RNHT0	JNHT1	RNHT1
birds	0,4895	0,9474	0,5053	0,5684	0,4895	0,5789	0,4895	0,5789	0,4895	0,5789	0,4895	0,5789	0,4895	0,5789	0,6000	0,5579	0,5737	0,5526	0,6737	0,6474	0,6421	0,5737	0,6474	0,5895
cal500	0,7149	0,9534	0,7155	0,9264	0,9299	0,7891	0,9299	0,7891	0,9299	0,7891	0,9299	0,9322	0,9299	0,9322	0,9460	0,9385	0,9471	0,9397	0,9414	0,9362	0,9408	0,9356	0,9402	0,9328
cellcycle	0,9433	0,9944	0,9994	0,9848	0,9809	0,9719	0,9809	0,9719	0,9809	0,9719	0,9809	0,9719	0,9809	0,9719	0,9916	0,9871	0,9893	0,9882	0,9916	0,9876	0,9904	0,9876	0,9899	0,9865
derisi	0,9865	0,9944	0,9994	0,9888	0,9910	0,9888	0,9910	0,9888	0,9910	0,9888	0,9910	0,9888	0,9910	0,9888	0,9916	0,9910	0,9927	0,9927	0,9933	0,9921	0,9938	0,9944	0,9949	0,9921
eisen	0,9079	0,9939	0,9958	0,9855	0,9933	0,9709	0,9933	0,9709	0,9933	0,9709	0,9933	0,9709	0,9933	0,9709	0,9909	0,9873	0,9933	0,9897	0,9915	0,9861	0,9903	0,9915	0,9909	0,9909
emotions	0,0000	0,0333	0,0167	0,0500	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0167	0,0000	0,0333	0,0167	0,0167	0,0167	0,0500	0,0167	0,0000	0,0167	0,0000
EukaryotePseAAC	0,6091	0,8773	0,6227	0,6591	0,5636	0,5818	0,5636	0,5818	0,5636	0,5818	0,5636	0,5818	0,5636	0,5818	0,6455	0,6318	0,6409	0,7318	0,7000	0,6545	0,7136	0,7000	0,6909	0,7455
flags	0,1143	0,0714	0,0857	0,0857	0,1286	0,1286	0,1286	0,1286	0,1286	0,1286	0,1286	0,1286	0,1286	0,1286	0,0571	0,1429	0,1429	0,1429	0,0571	0,1143	0,1286	0,1286	0,0857	0,1286
gasch1	0,8876	0,9944	0,9994	0,9888	0,9461	0,9876	0,9461	0,9876	0,9461	0,9876	0,9461	0,9258	0,9461	0,9258	0,9944	0,9921	0,9927	0,9921	0,9938	0,9955	0,9916	0,9933	0,9927	0,9927
GnegativeGO	0,0250	0,1375	0,0250	0,0250	0,1500	0,0250	0,1500	0,0250	0,1500	0,0250	0,1500	0,0250	0,1375	0,0250	0,0375	0,0250	0,0250	0,0250	0,0250	0,0250	0,0375	0,0250	0,0625	0,0250
GpositiveGO	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,0250	0,1000	0,0250	0,1000	0,0250	0,1000	0,0250	0,1000	0,0250	0,0750	0,0250
langlog	0,1720	0,6573	0,2293	0,5373	0,1840	0,3560	0,1840	0,3560	0,1840	0,3560	0,1840	0,3560	0,1840	0,3560	0,5480	0,5747	0,5813	0,5867	0,6027	0,6067	0,5893	0,5787	0,5973	0,5853
medical	0,6178	0,9556	0,8444	0,6267	0,7756	0,7222	0,7756	0,7222	0,7756	0,7222	0,7756	0,7222	0,7756	0,7222	0,6889	0,7222	0,7600	0,7578	0,6511	0,7556	0,6511	0,7844	0,6556	0,7933
pheno	0,9861	0,6970	0,6970	0,6885	0,6733	0,6630	0,6733	0,6630	0,6733	0,6630	0,6733	0,6636	0,6733	0,6636	0,6939	0,6952	0,6939	0,6915	0,6927	0,6897	0,6927	0,6921	0,6903	0,7009
PlantGO	0,0917	0,4083	0,1750	0,1250	0,2917	0,5167	0,2917	0,5167	0,2917	0,5167	0,2917	0,8815	0,2917	0,8815	0,1083	0,1333	0,1083	0,1167	0,1667	0,1833	0,1583	0,3500	0,1583	0,3500
scene	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
seq	0,7989	0,9944	0,8028	0,9904	0,9528	0,9528	0,9528	0,9528	0,9528	0,9528	0,9528	0,9961	0,9528	0,9961	0,9933	0,9927	0,9927	0,9933	0,9944	0,9910	0,9921	0,9916	0,9921	0,9921
VirusPseAAC	0,3500	0,7000	0,5500	0,4167	0,3833	0,3833	0,3833	0,3833	0,3833	0,3833	0,3833	0,3833	0,3833	0,3833	0,4500	0,4167	0,4500	0,4167	0,4167	0,4167	0,3833	0,4167	0,4167	0,4167
yeast	0,1214	0,2429	0,1214	0,2571	0,1643	0,1857	0,1643	0,1857	0,1643	0,1857	0,1643	0,1857	0,1643	0,1857	0,2571	0,2643	0,2571	0,2214	0,2214	0,2000	0,2571	0,2357	0,2071	0,2214
Yelp	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
AVERAGE	0,4420	0,5839	0,4705	0,4965	0,4820	0,4922	0,4820	0,4922	0,4820	0,4922	0,4820	0,5167	0,4814	0,5167	0,5047	0,5055	0,5135	0,5090	0,5115	0,5128	0,5135	0,5202	0,5102	0,5234

Tabela 52 – Resultados Macro-Precisão e Macro-Revocação e Macro-F1

MACRO-PRECISÃO

conjuntos de dados	Lo_C	G_C	Ra.H	Ra.NH	JHK1	RHK1	JHK2	RHK2	JHK3	RHK3	JHT0	RHT0	JHT1	RHT1	JNHK1	RNHK1	JNHK2	RNHK2	JNHK3	RNHK3	JNHT0	RNHT0	JNHT1	RNHT1
birds	0,3178	0,0011	0,3095	0,2969	0,3220	0,2559	0,3220	0,2559	0,3220	0,2559	0,3220	0,2559	0,3220	0,2559	0,2633	0,3096	0,2803	0,3104	0,2238	0,2474	0,2427	0,3007	0,2357	0,2742
cal500	0,1339	0,0296	0,1336	0,0429	0,0410	0,1007	0,0410	0,1007	0,0410	0,1007	0,0410	0,0392	0,0410	0,0392	0,0337	0,0368	0,0328	0,0360	0,0352	0,0381	0,0361	0,0385	0,0361	0,0398
cellcycle	0,0265	0,0001	0,0000	0,0068	0,0085	0,0136	0,0085	0,0136	0,0085	0,0136	0,0085	0,0136	0,0085	0,0136	0,0025	0,0067	0,0047	0,0063	0,0036	0,0089	0,0044	0,0068	0,0052	0,0092
derisi	0,0057	0,0001	0,0000	0,0042	0,0031	0,0050	0,0031	0,0050	0,0031	0,0050	0,0031	0,0050	0,0031	0,0050	0,0024	0,0042	0,0022	0,0033	0,0020	0,0043	0,0005	0,0028	0,0005	0,0035
eisen	0,0530	0,0001	0,0000	0,0070	0,0004	0,0153	0,0004	0,0153	0,0004	0,0153	0,0004	0,0153	0,0004	0,0153	0,0033	0,0075	0,0015	0,0053	0,0020	0,0101	0,0033	0,0058	0,0035	0,0071
emotions	0,5927	0,6322	0,6083	0,6207	0,6363	0,6363	0,6363	0,6363	0,6363	0,6363	0,6363	0,6363	0,6363	0,6363	0,6500	0,6107	0,6235	0,6042	0,6293	0,6321	0,6397	0,6143	0,6257	0,6143
EukaryotePseAAC	0,1185	0,0626	0,1134	0,1302	0,1291	0,1271	0,1291	0,1271	0,1291	0,1271	0,1291	0,1271	0,1291	0,1271	0,1302	0,1439	0,1226	0,1089	0,1087	0,1351	0,0996	0,1189	0,1087	0,1075
flags	0,6538	0,6595	0,6365	0,6283	0,6632	0,6071	0,6632	0,6071	0,6632	0,6071	0,6632	0,6071	0,6632	0,6071	0,6692	0,6404	0,6148	0,6158	0,6887	0,6227	0,6114	0,6183	0,6623	0,6183
gasch1	0,0498	0,0001	0,0000	0,0049	0,0202	0,0046	0,0202	0,0046	0,0202	0,0046	0,0202	0,0307	0,0202	0,0307	0,0021	0,0044	0,0031	0,0043	0,0021	0,0033	0,0031	0,0044	0,0021	0,0044
GnegativeGO	0,9252	0,8263	0,9063	0,9054	0,7923	0,9249	0,7923	0,9249	0,7923	0,9249	0,7923	0,9249	0,8138	0,9249	0,9001	0,8820	0,9052	0,8879	0,9111	0,8764	0,9176	0,9349	0,8746	0,9349
GpositiveGO	0,8760	0,8851	0,8503	0,8755	0,8549	0,8549	0,8549	0,8549	0,8549	0,8549	0,8549	0,8549	0,8549	0,8549	0,8035	0,8795	0,8043	0,8824	0,8071	0,8695	0,8097	0,8609	0,8335	0,8609
langlog	0,4373	0,2320	0,4176	0,2954	0,4221	0,3557	0,4221	0,3557	0,4221	0,3557	0,4221	0,3557	0,4221	0,3557	0,2934	0,2778	0,2743	0,2727	0,2607	0,2605	0,2688	0,2758	0,2630	0,2710
medical	0,3300	0,0221	0,1202	0,3222	0,1814	0,2401	0,1814	0,2401	0,1814	0,2401	0,1814	0,2401	0,1814	0,2401	0,2723	0,2393	0,1993	0,2127	0,2989	0,2136	0,2977	0,1815	0,2933	0,1796
pheno	0,0014	0,2867	0,2867	0,2933	0,3053	0,3132	0,3053	0,3132	0,3053	0,3132	0,3053	0,3113	0,3053	0,3113	0,2888	0,2882	0,2891	0,2914	0,2904	0,2931	0,2900	0,2902	0,2922	0,2841
PlantGO	0,6581	0,4145	0,6328	0,6397	0,5310	0,3234	0,5310	0,3234	0,5310	0,3234	0,5310	0,0708	0,5310	0,0708	0,6679	0,6192	0,6596	0,6469	0,6105	0,5973	0,6407	0,4797	0,6294	0,4797
scene	0,6215	0,6602	0,6481	0,6282	0,6433	0,6433	0,6433	0,6433	0,6433	0,6433	0,6433	0,6433	0,6433	0,6433	0,6301	0,6475	0,6282	0,6365	0,6283	0,6409	0,6251	0,6288	0,6339	0,6357
seq	0,0720	0,0001	0,0720	0,0048	0,0141	0,0141	0,0141	0,0141	0,0141	0,0141	0,0141	0,0000	0,0141	0,0000	0,0023	0,0045	0,0025	0,0044	0,0019	0,0048	0,0031	0,0048	0,0033	0,0048
VirusPseAAC	0,2838	0,1184	0,1906	0,2582	0,2764	0,2764	0,2764	0,2764	0,2764	0,2764	0,2764	0,2764	0,2764	0,2764	0,2279	0,2420	0,2476	0,2671	0,2459	0,2681	0,2908	0,2504	0,2447	0,2504
yeast	0,4186	0,5116	0,4223	0,4436	0,4070	0,3996	0,4070	0,3996	0,4070	0,3996	0,4070	0,3996	0,4070	0,3996	0,4649	0,4463	0,4695	0,4779	0,4776	0,4780	0,4604	0,4520	0,4720	0,4610
Yelp	0,7300	0,7092	0,7117	0,7219	0,6910	0,6910	0,6910	0,6910	0,6910	0,6910	0,6910	0,6910	0,6910	0,6910	0,7031	0,7281	0,7051	0,7219	0,7058	0,7262	0,7135	0,7037	0,7085	0,7037
AVERAGE	0,3653	0,3026	0,3530	0,3565	0,3471	0,3401	0,3471	0,3401	0,3471	0,3401	0,3471	0,3249	0,3482	0,3249	0,3506	0,3509	0,3435	0,3498	0,3467	0,3465	0,3479	0,3387	0,3464	0,3372

MACRO-RECALL

conjuntos de dados	Lo_C	G_C	Ra.H	Ra.NH	JHK1	RHK1	JHK2	RHK2	JHK3	RHK3	JHT0	RHT0	JHT1	RHT1	JNHK1	RNHK1	JNHK2	RNHK2	JNHK3	RNHK3	JNHT0	RNHT0	JNHT1	RNHT1
birds	0,2657	0,0526	0,2572	0,2079	0,2574	0,2248	0,2574	0,2248	0,2574	0,2248	0,2574	0,2248	0,2574	0,2248	0,1859	0,2167	0,2141	0,2250	0,1609	0,1870	0,1883	0,2071	0,1879	0,1957
cal500	0,0970	0,0466	0,0969	0,0537	0,0524	0,0809	0,0524	0,0809	0,0524	0,0809	0,0524	0,0515	0,0524	0,0515	0,0489	0,0517	0,0497	0,0510	0,0501	0,0521	0,0503	0,0532	0,0518	0,0529
cellcycle	0,0130	0,0056	0,0006	0,0075	0,0059	0,0065	0,0059	0,0065	0,0059	0,0065	0,0059	0,0065	0,0059	0,0065	0,0056	0,0056	0,0057	0,0050	0,0042	0,0031	0,0053	0,0050	0,0043	0,0035
derisi	0,0083	0,0056	0,0006	0,0079	0,0057	0,0055	0,0057	0,0055	0,0057	0,0055	0,0057	0,0055	0,0057	0,0055	0,0062	0,0056	0,0055	0,0047	0,0052	0,0045	0,0060	0,0032	0,0048	0,0049
eisen	0,0290	0,0061	0,0042	0,0087	0,0063	0,0092	0,0063	0,0092	0,0063	0,0092	0,0063	0,0092	0,0063	0,0092	0,0071	0,0067	0,0055	0,0065	0,0070	0,0068	0,0075	0,0042	0,0062	0,0043
emotions	0,5579	0,5406	0,5327	0,5419	0,5741	0,5741	0,5741	0,5741	0,5741	0,5741	0,5741	0,5741	0,5741	0,5741	0,5768	0,5706	0,5815	0,5646	0,5716	0,5471	0,5644	0,5361	0,5590	0,5361
EukaryotePseAAC	0,0958	0,0521	0,1011	0,0843	0,1054	0,1042	0,1054	0,1042	0,1054	0,1042	0,1054	0,1042	0,1054	0,1042	0,0846	0,0974	0,0864	0,0749	0,0774	0,0829	0,0781	0,0801	0,0817	0,0652
flags	0,5890	0,6357	0,6485	0,6281	0,5817	0,6348	0,5817	0,6348	0,5817	0,6348	0,5817	0,6348	0,5817	0,6348	0,6298	0,5802	0,5895	0,6038	0,6480	0,6102	0,6334	0,6151	0,6004	0,6151
gasch1	0,0222	0,0056	0,0006	0,0080	0,0150	0,0048	0,0150	0,0048	0,0150	0,0048	0,0150	0,0181	0,0150	0,0181	0,0044	0,0049	0,0053	0,0048	0,0050	0,0021	0,0066	0,0038	0,0060	0,0044
GnegativeGO	0,9344	0,7924	0,9438	0,9366	0,7999	0,9222	0,7999	0,9222	0,7999	0,9222	0,7999	0,9222	0,8145	0,9222	0,9150	0,9374	0,9289	0,9410	0,9349	0,9321	0,9229	0,9169	0,8940	0,9169
GpositiveGO	0,9103	0,9092	0,9353	0,8921	0,9292	0,9292	0,9292	0,9292	0,9292	0,9292	0,9292	0,9292	0,9292	0,9292	0,8509	0,8919	0,8468	0,9066	0,8511	0,8999	0,8508	0,9417	0,8633	0,9417
langlog	0,3171	0,2050	0,3043	0,2356	0,3114	0,2739	0,3114	0,2739	0,3114	0,2739	0,3114	0,2739	0,3114	0,2739	0,2329	0,2230	0,2256	0,2236	0,2214	0,2207	0,2246	0,2282	0,2184	0,2272
medical	0,3424	0,0443	0,1415	0,3345	0,2035	0,2493	0,2035	0,2493	0,2035	0,2493	0,2035	0,2493	0,2035	0,2493	0,2812	0,2513	0,2137	0,2208	0,3127	0,2199	0,3128	0,1957	0,3075	0,1877
pheno	0,0071	0,3002	0,3002	0,3008	0,3043	0,3062	0,3043	0,3062	0,3043	0,3062	0,3043	0,3057	0,3043	0,3057	0,3002	0,2997	0,3005	0,3004	0,3007	0,3007	0,3009	0,2990	0,3009	0,2902
PlantGO	0,7258	0,4365	0,6061	0,6897	0,5335	0,4027	0,5335	0,4027	0,5335	0,4027	0,5335	0,0926	0,5335	0,0926	0,6893	0,6903	0,6935	0,6999	0,6319	0,6304	0,6606	0,5170	0,6466	0,5170
scene	0,6676	0,5899	0,6058	0,6449	0,6043	0,6043	0,6043	0,6043	0,6043	0,6043	0,6043	0,6043	0,6043	0,6043	0,6349	0,6529	0,6333	0,6443	0,6311	0,6528	0,6409	0,6332	0,6386	0,6479
seq	0,0386	0,0056	0,0377	0,0068	0,0123	0,0123	0,0123	0,0123	0,0123	0,0123	0,0123	0,0039	0,0123	0,0039	0,0045	0,0042	0,0058	0,0043	0,0045	0,0062	0,0056	0,0052	0,0058	0,0051
VirusPseAAC	0,3692	0,1792	0,2442	0,3289	0,3611	0,3611	0,3611	0,3611	0,3611	0,3611	0,3611	0,3611	0,3611	0,3611	0,3019	0,3137	0,3016	0,3204	0,3220	0,3505	0,3454	0,3262	0,3102	0,3262
yeast	0,3704	0,2850	0,3619	0,3160	0,3749	0,3797	0,3749	0,3797	0,3749	0,3797	0,3749	0,3797	0,3749	0,3797	0,3140	0,3354	0,3311	0,3303	0,3352	0,3274	0,3181	0,3362	0,3245	0,3418
Yelp	0,6823	0,5931	0,6519	0,6412	0,5893	0,5893	0,5893	0,5893	0,5893	0,5893	0,5893	0,5893	0,5893	0,5893	0,6276	0,6550	0,6332	0,6499	0,6329	0,6470	0,6332	0,6126	0,6282	0,6126
AVERAGE	0,3522	0,2845	0,3388	0,3438	0,3314	0,3337	0,3314	0,3337	0,3314	0,3337	0,3314	0,3170	0,3321	0,3170	0,3351	0,3397	0,3329	0,3391	0,3354	0,3342	0,3378	0,3260	0,3320	0,3248

MACRO-F1

conjuntos de dados	Lo_C	G_C	Ra.H	Ra.NH	JHK1	RHK1	JHK2	RHK2	JHK3	RHK3	JHT0	RHT0	JHT1	RHT1	JNHK1	RNHK1	JNHK2	RNHK2	JNHK3	RNHK3	JNHT0	RNHT0	JNHT1	RNHT1
birds	0,2602	0,0022	0,2541	0,2084	0,2588	0,2074	0,2588	0,2074	0,2588	0,2074	0,2588	0,2074	0,2588	0,2074	0,1961	0,2270	0,2128	0,2249	0,1590	0,1806	0,1827	0,2183	0,1790	0,1951
cal500	0,1020	0,0359	0,1018	0,0446	0,0433	0,0812	0,0433	0,0812	0,0433	0,0812	0,0433	0,0421	0,0433	0,0421	0,0385	0,0419	0,0388	0,0411	0,0400	0,0423	0,0401	0,0432	0,0413	0,0434
cellcycle	0,0115	0,0001	0,0000	0,0028	0,0034	0,0060	0,0034	0,0060	0,0034	0,0060	0,0034	0,0060	0,0034	0,0060	0,0010	0,0027	0,0019	0,0026	0,0013	0,0030	0,0018	0,0027	0,0016	0,0029
derisi	0,0039	0,0001	0,0000	0,0030	0,0018	0,0031	0,0018	0,0031	0,0018	0,0031	0,0018	0,0031	0,0018	0,0031	0,0016	0,0030	0,0015	0,0024	0,0015	0,0031	0,0005	0,0019	0,0004	0,0026
eisen	0,0295	0,0002	0,0000	0,0041	0,0004	0,0077	0,0004	0,0077	0,0004	0,0077	0,0004	0,0077	0,0004	0,0077	0,0023	0,0042	0,0012	0,0033	0,0013	0,0054	0,0024	0,0034	0,0019	0,0042
emotions	0,5633	0,5553	0,5423	0,5609	0,5837	0,5837	0,5837	0,5837	0,5837	0,5837	0,5837	0,5837	0,5837	0,5837	0,5945	0,5747	0,5868	0,5679	0,5806	0,5631	0,5803	0,5476	0,5706	0,5476
EukaryotePseAAC	0,0984	0,0162	0,0990	0,0818	0,1012	0,0996	0,1012	0,0996	0,1012	0,0996	0,1012	0,0996	0,1012	0,0996	0,0822	0,0889	0,0811	0,0588	0,0628	0,0791	0,0612	0,0718	0,0671	0,0596
flags	0,5842	0,6122	0,6155	0,6064	0,5741	0,5936	0,5741	0,5936	0,5741	0,5936	0,5741	0,5936	0,5741	0,5936	0,6142	0,5745	0,5743	0,5766	0,6313	0,5864	0,6081	0,5906	0,5920	0,5906
gasch1	0,0249	0,0001	0,0000	0,0032	0,0122	0,0023	0,0122	0,0023	0,0122	0,0023	0,0122	0,0173	0,0122	0,0173	0,0015	0,0028	0,0019	0,0027	0,0014	0,0021	0,0021	0,0028	0,0014	0,0028
GnegativeGO	0,9270	0,8049	0,9204	0,9168	0,7897	0,9182	0,7897	0,9182	0,7897	0,9182	0,7897	0,9182	0,8098	0,9182	0,9043	0,8998	0,9149	0,9036	0,9193	0,8928	0,9183	0,9230	0,8812	0,9230
GpositiveGO	0,8847	0,8859	0,8766	0,8734	0,8769	0,8769	0,8769	0,8769	0,8769	0,8769	0,8769	0,8769	0,8769	0,8769	0,8183	0,8730	0,8161	0,8835	0,8202	0,8689	0,8215	0,8853	0,8373	0,8853
langlog	0,3467	0,2119	0,3328	0,2465	0,3385	0,2949	0,3385	0,2949	0,3385	0,2949	0,3385	0,2949	0,3385	0,2949	0,2446	0,2346	0,2361	0,2353	0,2296	0,2294	0,2343	0,2394	0,2278	0,2345
medical	0,3285	0,0262	0,1241	0,3212	0,1829	0,2382	0,1829	0,2382	0,1829	0,2382	0,1829	0,2382	0,1829	0,2382	0,2699	0,2381	0,2002	0,2109	0,2970	0,2113	0,2963	0,1810	0,2926	0,1784
pheno	0,0015	0,2925	0,2925	0,2939	0,2998	0,3033	0,2998	0,3033	0,2998	0,3033	0,2998	0,3025	0,2998	0,3025	0,2930	0,2926	0,2933	0,2935	0,2938	0,2942	0,2938	0,2925	0,2941	0,2848
PlantGO	0,6649	0,4068	0,5914	0,6368	0,5053	0,3316	0,5053	0,3316	0,5053	0,3316	0,5053	0,0648	0,5053	0,0648	0,6494	0,6175	0,6482	0,6365	0,5910	0,5814	0,6196	0,4659	0,6137	0,4659
scene	0,6299	0,6006	0,6076	0,6205	0,6001	0,6001	0,6001	0,6001	0,6001	0,6001	0,6001	0,6001	0,6001	0,6001	0,6154	0,6337	0,6133	0,6252	0,6125	0,6311	0,6175	0,6153	0,6178	0,6251
seq	0,0452	0,0002	0,0451	0,0035	0,0089	0,0089	0,0089	0,0089	0,0089	0,0089	0,0089	0,0001	0,0089	0,0001	0,0015	0,0032	0,0018	0,0033	0,0015	0,0035	0,0022	0,0031	0,0025	0,0036
VirusPseAAC	0,2741	0,1197	0,1736	0,2459	0,2725	0,2725	0,2725	0,2725	0,2725	0,2725	0,2725	0,2725	0,2725	0,2725	0,2184	0,2293	0,2181	0,2379	0,2283	0,2705	0,2692	0,2374	0,2224	0,2374
yeast	0,3855	0,3093	0,3800	0,3385	0,3792	0,3822	0,3792	0,3822	0,3792	0,3822	0,3792	0,3822	0,3792	0,3822	0,3367	0,3545	0,3520	0,3562	0,3556	0,3524	0,3404	0,3559	0,3471	0,3613
Yelp	0,7016	0,6372	0,6739	0,6715	0,6271	0,6271	0,6271	0,6271	0,6271	0,6271	0,6271	0,6271	0,6271	0,6271	0,6570	0,6837	0,6611	0,6777	0,6598	0,6776	0,6646	0,6479	0,6589	0,6479
AVERAGE	0,3434	0,2759	0,3315	0,3342	0,3230	0,3219	0,3230	0,3219	0,3230	0,3219	0,3230	0,3069	0,3240	0,3069	0,3270	0,3290	0,3228	0,3272	0,3244	0,3239	0,3279	0,3164	0,3225	0,3148

Tabela 53 – Resultados Micro-Precisão, Micro-Revocação e Micro-F1

MICRO-PRECISÃO

conjuntos de dados	Lo_C	G_C	Ra.H	Ra.NH	JHK1	RHK1	JHK2	RHK2	JHK3	RHK3	JHT0	RHT0	JHT1	RHT1	JNHK1	RNHK1	JNHK2	RNHK2	JNHK3	RNHK3	JNHT0	RNHT0	JNHT1	RNHT1
birds	0,2993	0,0217	0,3083	0,2819	0,3023	0,2541	0,3023	0,2541	0,3023	0,2541	0,3023	0,2541	0,3023	0,2541	0,2762	0,2906	0,3010	0,2885	0,2412	0,2542	0,2736	0,2832	0,2685	0,2515
cal500	0,4815	0,6355	0,4819	0,6207	0,6148	0,5246	0,6148	0,5246	0,6148	0,5246	0,6148	0,6146	0,6148	0,6146	0,6275	0,6228	0,6283	0,6222	0,6233	0,6235	0,6275	0,6155	0,6233	0,6170
cellcycle	0,0796	0,0123	0,0003	0,0240	0,0371	0,0350	0,0371	0,0350	0,0371	0,0350	0,0371	0,0350	0,0371	0,0350	0,0154	0,0343	0,0184	0,0315	0,0138	0,0208	0,0173	0,0274	0,0139	0,0304
derisi	0,0373	0,0124	0,0003	0,0268	0,0245	0,0322	0,0245	0,0322	0,0245	0,0322	0,0245	0,0322	0,0245	0,0322	0,0181	0,0342	0,0163	0,0223	0,0158	0,0335	0,0099	0,0163	0,0093	0,0233
eisen	0,1434	0,0149	0,0029	0,0503	0,0170	0,0397	0,0170	0,0397	0,0170	0,0397	0,0170	0,0397	0,0170	0,0397	0,0351	0,0399	0,0136	0,0438	0,0211	0,0535	0,0278	0,0376	0,0259	0,0371
emotions	0,5907	0,6413	0,6134	0,6286	0,6404	0,6404	0,6404	0,6404	0,6404	0,6404	0,6404	0,6404	0,6404	0,6404	0,6424	0,6234	0,6185	0,6164	0,6350	0,6301	0,6323	0,6025	0,6326	0,6025
EukaryotePseAAC	0,2598	0,0364	0,3327	0,2337	0,2601	0,2911	0,2601	0,2911	0,2601	0,2911	0,2601	0,2911	0,2601	0,2911	0,2450	0,2424	0,2443	0,1553	0,1854	0,2239	0,2005	0,2096	0,1924	0,1638
flags	0,7291	0,7198	0,7031	0,7140	0,7379	0,7068	0,7379	0,7068	0,7379	0,7068	0,7379	0,7068	0,7379	0,7068	0,7097	0,7467	0,7247	0,7251	0,7321	0,7197	0,7239	0,7147	0,7230	0,7147
gasch1	0,1241	0,0122	0,0003	0,0293	0,0726	0,0172	0,0726	0,0172	0,0726	0,0172	0,0726	0,0902	0,0726	0,0902	0,0185	0,0240	0,0205	0,0228	0,0141	0,0158	0,0251	0,0234	0,0208	0,0232
GnegativeGO	0,9441	0,9580	0,9448	0,9443	0,9548	0,9533	0,9548	0,9533	0,9548	0,9533	0,9548	0,9533	0,9564	0,9533	0,9462	0,9451	0,9434	0,9426	0,9435	0,9451	0,9462	0,9557	0,9469	0,9557
GpositiveGO	0,9334	0,9442	0,9307	0,9328	0,9337	0,9337	0,9337	0,9337	0,9337	0,9337	0,9337	0,9337	0,9337	0,9337	0,9206	0,9305	0,9203	0,9342	0,9258	0,9243	0,9291	0,9356	0,9277	0,9356
langlog	0,6251	0,6909	0,6331	0,6744	0,6194	0,6442	0,6194	0,6442	0,6194	0,6442	0,6194	0,6442	0,6194	0,6442	0,6787	0,6885	0,6828	0,6899	0,6851	0,6848	0,6810	0,6810	0,6812	0,6792
medical	0,7893	0,3757	0,5850	0,7897	0,5862	0,6733	0,5862	0,6733	0,5862	0,6733	0,5862	0,6733	0,5862	0,6733	0,6968	0,6519	0,5797	0,5972	0,7361	0,5950	0,7339	0,5463	0,7319	0,5577
pheno	0,0358	0,9542	0,9542	0,9536	0,9566	0,9578	0,9566	0,9578	0,9566	0,9578	0,9566	0,9577	0,9566	0,9577	0,9552	0,9555	0,9551	0,9556	0,9558	0,9566	0,9548	0,9555	0,9557	0,9669
PlantGO	0,7305	0,7144	0,7389	0,7297	0,7240	0,6522	0,7240	0,6522	0,7240	0,6522	0,7240	0,1963	0,7240	0,1963	0,7334	0,7306	0,7313	0,7257	0,7116	0,7236	0,7391	0,6986	0,7278	0,6986
scene	0,5724	0,6165	0,6016	0,5759	0,5889	0,5889	0,5889	0,5889	0,5889	0,5889	0,5889	0,5889	0,5889	0,5889	0,5840	0,5940	0,5745	0,5872	0,5763	0,5880	0,5761	0,5766	0,5812	0,5834
seq	0,1576	0,0153	0,1553	0,0306	0,0493	0,0493	0,0493	0,0493	0,0493	0,0493	0,0493	0,0064	0,0493	0,0064	0,0148	0,0278	0,0232	0,0275	0,0173	0,0303	0,0199	0,0268	0,0298	0,0336
VirusPseAAC	0,3698	0,2356	0,3365	0,3624	0,3802	0,3802	0,3802	0,3802	0,3802	0,3802	0,3802	0,3802	0,3802	0,3802	0,3393	0,3388	0,3287	0,3591	0,3615	0,3869	0,3886	0,3623	0,3448	0,3623
yeast	0,6012	0,6962	0,6053	0,6735	0,6149	0,6128	0,6149	0,6128	0,6149	0,6128	0,6149	0,6128	0,6149	0,6128	0,6832	0,6720	0,6753	0,6796	0,6690	0,6742	0,6755	0,6681	0,6754	0,6632
Yelp	0,7546	0,7202	0,7246	0,7347	0,7138	0,7138	0,7138	0,7138	0,7138	0,7138	0,7138	0,7138	0,7138	0,7138	0,7255	0,7407	0,7232	0,7393	0,7229	0,7407	0,7236	0,7274	0,7269	0,7274
AVERAGE	**0,4629**	**0,4514**	**0,4827**	**0,5005**	**0,4914**	**0,4850**	**0,4914**	**0,4850**	**0,4914**	**0,4850**	**0,4914**	**0,4682**	**0,4915**	**0,4682**	**0,4933**	**0,4967**	**0,4861**	**0,4883**	**0,4893**	**0,4912**	**0,4953**	**0,4832**	**0,4919**	**0,4814**

MICRO-REVOCAÇÃO

conjuntos de dados	Lo_C	G_C	H.Ra	NH.Ra	H.J.K1	H.Ro.K1	H.J.K2	H.Ro.K2	H.J.K3	H.Ro.K3	H.J.T0	H.Ro.T0	H.J.T1	H.Ro.T1	NH.J.K1	NH.Ro.K1	NH.J.K2	NH.Ro.K2	NH.J.K3	NH.Ro.K3	NH.J.T0	NH.Ro.T0	NH.J.T1	NH.Ro.T1
birds	0,3756	0,0214	0,3831	0,3254	0,3770	0,3145	0,3770	0,3145	0,3770	0,3145	0,3770	0,3145	0,3770	0,3145	0,3150	0,3370	0,3535	0,3355	0,2738	0,2934	0,3149	0,3308	0,3072	0,2936
cal500	0,2711	0,1976	0,2710	0,2177	0,2061	0,2484	0,2061	0,2484	0,2061	0,2484	0,2061	0,2050	0,2061	0,2050	0,2047	0,2112	0,2066	0,2099	0,2056	0,2136	0,2076	0,2155	0,2118	0,2147
cellcycle	0,0377	0,0056	0,0001	0,0109	0,0173	0,0168	0,0173	0,0168	0,0173	0,0168	0,0173	0,0168	0,0173	0,0168	0,0071	0,0157	0,0084	0,0143	0,0063	0,0095	0,0079	0,0125	0,0063	0,0138
derisi	0,0170	0,0056	0,0001	0,0122	0,0111	0,0147	0,0111	0,0147	0,0111	0,0147	0,0111	0,0147	0,0111	0,0147	0,0081	0,0156	0,0075	0,0102	0,0072	0,0153	0,0046	0,0075	0,0041	0,0106
eisen	0,0664	0,0063	0,0012	0,0215	0,0072	0,0186	0,0072	0,0186	0,0072	0,0186	0,0072	0,0186	0,0072	0,0186	0,0150	0,0171	0,0058	0,0186	0,0090	0,0229	0,0120	0,0162	0,0111	0,0159
emotions	0,5667	0,5686	0,5550	0,5561	0,5895	0,5895	0,5895	0,5895	0,5895	0,5895	0,5895	0,5895	0,5895	0,5895	0,5894	0,5886	0,5912	0,5822	0,5867	0,5633	0,5770	0,5469	0,5796	0,5469
EukaryotePseAAC	0,2823	0,0318	0,3564	0,2232	0,2817	0,3041	0,2817	0,3041	0,2817	0,3041	0,2817	0,3041	0,2817	0,3041	0,2386	0,2359	0,2386	0,1471	0,1780	0,2173	0,1920	0,2027	0,1855	0,1578
flags	0,7084	0,7393	0,7644	0,7372	0,7026	0,7513	0,7026	0,7513	0,7026	0,7513	0,7026	0,7513	0,7026	0,7513	0,7294	0,7048	0,7160	0,7338	0,7488	0,7386	0,7540	0,7315	0,7063	0,7315
gasch1	0,0631	0,0056	0,0001	0,0134	0,0362	0,0081	0,0362	0,0081	0,0362	0,0081	0,0362	0,0461	0,0362	0,0461	0,0084	0,0109	0,0094	0,0105	0,0064	0,0073	0,0113	0,0107	0,0095	0,0106
GnegativeGO	0,9595	0,9231	0,9609	0,9616	0,9293	0,9375	0,9293	0,9375	0,9293	0,9375	0,9293	0,9375	0,9334	0,9375	0,9533	0,9540	0,9574	0,9567	0,9512	0,9560	0,9547	0,9334	0,9505	0,9334
GpositiveGO	0,9618	0,9369	0,9656	0,9522	0,9599	0,9599	0,9599	0,9599	0,9599	0,9599	0,9599	0,9599	0,9599	0,9599	0,9464	0,9445	0,9426	0,9485	0,9466	0,9502	0,9465	0,9618	0,9484	0,9618
langlog	0,4846	0,4103	0,4760	0,4420	0,4836	0,4562	0,4836	0,4562	0,4836	0,4562	0,4836	0,4562	0,4836	0,4562	0,4381	0,4275	0,4287	0,4294	0,4290	0,4275	0,4295	0,4323	0,4224	0,4349
medical	0,7975	0,3013	0,5477	0,7925	0,5682	0,6502	0,5682	0,6502	0,5682	0,6502	0,5682	0,6502	0,5682	0,6502	0,6900	0,6337	0,5677	0,5783	0,7379	0,5733	0,7344	0,5191	0,7353	0,5322
pheno	0,0166	0,9392	0,9392	0,9393	0,9403	0,9405	0,9403	0,9405	0,9403	0,9405	0,9403	0,9405	0,9403	0,9405	0,9391	0,9384	0,9392	0,9390	0,9394	0,9392	0,9395	0,9291	0,9392	0,9373
PlantGO	0,7918	0,7195	0,7906	0,7804	0,7424	0,6620	0,7424	0,6620	0,7424	0,6620	0,7424	0,2476	0,7424	0,2476	0,7833	0,7822	0,7843	0,7777	0,7653	0,7672	0,7822	0,7131	0,7813	0,7131
scene	0,6561	0,5741	0,5908	0,6328	0,5880	0,5880	0,5880	0,5880	0,5880	0,5880	0,5880	0,5880	0,5880	0,5880	0,6217	0,6402	0,6209	0,6317	0,6175	0,6406	0,6290	0,6186	0,6259	0,6360
seq	0,0842	0,0053	0,0829	0,0106	0,0226	0,0226	0,0226	0,0226	0,0226	0,0226	0,0226	0,0022	0,0226	0,0022	0,0051	0,0096	0,0081	0,0095	0,0060	0,0105	0,0069	0,0093	0,0103	0,0117
VirusPseAAC	0,4070	0,1991	0,3024	0,3681	0,4081	0,4081	0,4081	0,4081	0,4081	0,4081	0,4081	0,4081	0,4081	0,4081	0,3333	0,3486	0,3143	0,3487	0,3501	0,4117	0,3743	0,3753	0,3348	0,3753
yeast	0,5530	0,5109	0,5476	0,5373	0,5868	0,5913	0,5868	0,5913	0,5868	0,5913	0,5868	0,5913	0,5868	0,5913	0,5390	0,5630	0,5577	0,5544	0,5584	0,5480	0,5387	0,5633	0,5484	0,5646
Yelp	0,7496	0,6646	0,7145	0,7189	0,6660	0,6660	0,6660	0,6660	0,6660	0,6660	0,6660	0,6660	0,6660	0,6660	0,7031	0,7224	0,7053	0,7236	0,7120	0,7191	0,7014	0,6829	0,7033	0,6829
AVERAGE	**0,4425**	**0,3883**	**0,4625**	**0,4627**	**0,4562**	**0,4574**	**0,4562**	**0,4574**	**0,4562**	**0,4574**	**0,4562**	**0,4354**	**0,4564**	**0,4354**	**0,4534**	**0,4551**	**0,4482**	**0,4480**	**0,4518**	**0,4512**	**0,4559**	**0,4406**	**0,4511**	**0,4389**

MICRO-F1

conjuntos de dados	Lo	G	Ra.H	Ra.NH	JHK1	RHK1	JHK2	RHK2	JHK3	RHK3	JHT0	RHT0	JHT1	RHT1	JNHK1	RNHK1	JNHK2	RNHK2	JNHK3	RNHK3	JNHT0	RNHT0	JNHT1	RNHT1
birds	0,3330	0,0215	0,3415	0,3019	0,3353	0,2806	0,3353	0,2806	0,3353	0,2806	0,3353	0,2806	0,3353	0,2806	0,2941	0,3119	0,3248	0,3101	0,2562	0,2722	0,2922	0,3051	0,2863	0,2706
cal500	0,3465	0,3014	0,3466	0,3223	0,3073	0,3316	0,3073	0,3316	0,3073	0,3316	0,3073	0,3062	0,3073	0,3062	0,3085	0,3153	0,3108	0,3137	0,3089	0,3179	0,3117	0,3190	0,3153	0,3185
cellcycle	0,0511	0,0077	0,0002	0,0150	0,0236	0,0227	0,0236	0,0227	0,0236	0,0227	0,0236	0,0227	0,0236	0,0227	0,0097	0,0215	0,0115	0,0197	0,0087	0,0130	0,0108	0,0172	0,0087	0,0190
derisi	0,0233	0,0077	0,0002	0,0168	0,0152	0,0202	0,0152	0,0202	0,0152	0,0202	0,0152	0,0202	0,0152	0,0202	0,0112	0,0215	0,0102	0,0140	0,0099	0,0210	0,0062	0,0102	0,0057	0,0146
eisen	0,0908	0,0089	0,0017	0,0301	0,0101	0,0253	0,0101	0,0253	0,0101	0,0253	0,0101	0,0253	0,0101	0,0253	0,0210	0,0239	0,0081	0,0262	0,0126	0,0321	0,0168	0,0227	0,0156	0,0222
emotions	0,5775	0,6012	0,5822	0,5888	0,6129	0,6129	0,6129	0,6129	0,6129	0,6129	0,6129	0,6129	0,6129	0,6129	0,6140	0,6045	0,6037	0,5981	0,6083	0,5943	0,6010	0,5726	0,6038	0,5726
EukaryotePseAAC	0,2706	0,0340	0,3441	0,2283	0,2704	0,2974	0,2704	0,2974	0,2704	0,2974	0,2704	0,2974	0,2704	0,2974	0,2417	0,2391	0,2414	0,1510	0,1815	0,2205	0,1961	0,2060	0,1889	0,1607
flags	0,7179	0,7276	0,7319	0,7243	0,7184	0,7273	0,7184	0,7273	0,7184	0,7273	0,7184	0,7273	0,7184	0,7273	0,7184	0,7224	0,7199	0,7285	0,7395	0,7278	0,7384	0,7220	0,7123	0,7220
gasch1	0,0836	0,0077	0,0002	0,0184	0,0483	0,0110	0,0483	0,0110	0,0483	0,0110	0,0483	0,0610	0,0483	0,0610	0,0115	0,0150	0,0129	0,0143	0,0088	0,0100	0,0156	0,0147	0,0130	0,0145
GnegativeGO	0,9516	0,9402	0,9526	0,9527	0,9418	0,9453	0,9418	0,9453	0,9418	0,9453	0,9418	0,9453	0,9447	0,9453	0,9497	0,9494	0,9502	0,9495	0,9473	0,9505	0,9503	0,9443	0,9486	0,9443
GpositiveGO	0,9473	0,9405	0,9476	0,9423	0,9464	0,9464	0,9464	0,9464	0,9464	0,9464	0,9464	0,9464	0,9464	0,9464	0,9332	0,9373	0,9311	0,9411	0,9359	0,9369	0,9375	0,9484	0,9378	0,9484
langlog	0,5458	0,5139	0,5424	0,5339	0,5431	0,5313	0,5431	0,5313	0,5431	0,5313	0,5431	0,5313	0,5431	0,5313	0,5323	0,5270	0,5262	0,5289	0,5271	0,5258	0,5264	0,5287	0,5208	0,5297
medical	0,7931	0,3343	0,5655	0,7909	0,5757	0,6607	0,5757	0,6607	0,5757	0,6607	0,5757	0,6607	0,5757	0,6607	0,6925	0,6417	0,5719	0,5861	0,7363	0,5827	0,7336	0,5311	0,7329	0,5433
pheno	0,0227	0,9466	0,9466	0,9464	0,9484	0,9491	0,9484	0,9491	0,9484	0,9491	0,9484	0,9490	0,9484	0,9490	0,9471	0,9468	0,9471	0,9472	0,9475	0,9478	0,9471	0,9416	0,9474	0,9514
PlantGO	0,7593	0,7169	0,7637	0,7539	0,7329	0,6567	0,7329	0,6567	0,7329	0,6567	0,7329	0,2149	0,7329	0,2149	0,7569	0,7551	0,7563	0,7501	0,7369	0,7443	0,7595	0,7054	0,7530	0,7054
scene	0,6113	0,5945	0,5961	0,6030	0,5883	0,5883	0,5883	0,5883	0,5883	0,5883	0,5883	0,5883	0,5883	0,5883	0,6022	0,6162	0,5967	0,6085	0,5961	0,6131	0,6012	0,5968	0,6025	0,6085
seq	0,1097	0,0079	0,1081	0,0158	0,0307	0,0307	0,0307	0,0307	0,0307	0,0307	0,0307	0,0033	0,0307	0,0033	0,0076	0,0143	0,0120	0,0142	0,0089	0,0156	0,0102	0,0138	0,0153	0,0173
VirusPseAAC	0,3849	0,2153	0,3180	0,3644	0,3931	0,3931	0,3931	0,3931	0,3931	0,3931	0,3931	0,3931	0,3931	0,3931	0,3353	0,3417	0,3205	0,3529	0,3552	0,3989	0,3803	0,3670	0,3391	0,3670
yeast	0,5759	0,5882	0,5748	0,5975	0,6003	0,6015	0,6003	0,6015	0,6003	0,6015	0,6003	0,6015	0,6003	0,6015	0,6018	0,6123	0,6041	0,6104	0,6084	0,6041	0,5983	0,6112	0,6044	0,6096
Yelp	0,7521	0,6913	0,7194	0,7267	0,6890	0,6890	0,6890	0,6890	0,6890	0,6890	0,6890	0,6890	0,6890	0,6890	0,7141	0,7313	0,7141	0,7313	0,7173	0,7297	0,7123	0,7044	0,7148	0,7044
AVERAGE	**0,4474**	**0,4104**	**0,4692**	**0,4737**	**0,4666**	**0,4661**	**0,4666**	**0,4661**	**0,4666**	**0,4661**	**0,4666**	**0,4438**	**0,4667**	**0,4438**	**0,4651**	**0,4674**	**0,4590**	**0,4598**	**0,4626**	**0,4629**	**0,4673**	**0,4541**	**0,4633**	**0,4522**

Tabela 54 – Comparação Pareada Macro e Micro Precisão

MACRO-PRECISÃO

	Lo.c	G.c	Ra.H	Ra.NH	J.H.K1	R.H.K1	J.H.K2	R.H.K2	J.H.K3	R.H.K3	J.H.T0	R.H.T0	J.H.T1	R.H.T1	J.NH.K1	R.NH.K1	J.NH.K2	R.NH.K2	J.NH.K3	R.NH.K3	J.NH.T0	R.NH.T0	J.NH.T1	R.NH.T1	Av.
Lo.c	0	14	16	15	14	16	14	16	14	16	14	16	14	16	13	14	14	15	15	15	15	14	15	15	**14,2**
G.c	6	0	10	5	5	6	5	6	5	6	5	7	5	7	4	5	6	5	5	5	4	6	5	7	**5,4**
Ra.H	4	10	0	7	7	9	7	9	7	9	7	9	7	9	8	6	9	8	9	8	8	8	10	10	**7,7**
Ra.Nh	5	15	13	0	10	9	10	9	10	9	10	10	10	10	15	12	16	14	15	11	15	15	16	14	**11,4**
J.H.K1	6	15	13	10	0	7	0	7	0	7	0	8	0	8	11	10	14	12	13	10	12	14	14	13	**8,5**
R.H.K1	4	14	11	11	7	0	7	0	7	0	7	4	7	4	12	12	15	14	15	14	13	13	15	13	**9,1**
J.H.K2	6	15	13	10	0	7	0	7	0	7	0	8	0	8	11	10	14	12	13	10	12	14	14	13	**8,5**
R.H.K2	4	14	11	11	7	0	7	0	7	0	7	4	7	4	12	12	15	14	15	14	13	13	15	13	**9,1**
J.H.K3	6	15	13	10	0	7	0	7	0	7	0	8	0	8	11	10	14	12	13	10	12	14	14	13	**8,5**
R.H.K3	4	14	11	11	7	0	7	0	7	0	7	4	7	4	12	12	15	14	15	14	13	13	15	13	**9,1**
J.H.T0	6	15	13	10	0	7	0	7	0	7	0	8	0	8	11	10	14	12	13	10	12	14	14	13	**8,5**
R.H.T0	4	13	11	10	7	1	7	1	7	1	7	0	7	0	11	11	14	13	14	13	12	12	14	11	**8,4**
J.H.T1	6	15	13	10	1	7	1	7	1	7	1	8	0	8	11	10	14	12	13	10	12	14	14	13	**8,7**
R.H.T1	4	13	11	10	7	1	7	1	7	1	7	0	7	0	11	11	14	13	14	13	12	12	14	11	**8,4**
J.NH.K1	7	16	12	5	9	8	9	8	9	8	9	9	9	9	0	8	10	7	10	7	9	8	9	8	**8,5**
R.NH.K1	6	15	14	8	10	8	10	8	10	8	10	9	10	9	12	0	14	13	13	11	13	12	13	13	**10,4**
J.NH.K2	6	14	11	4	6	5	6	5	6	5	6	6	6	6	10	6	0	5	9	4	10	6	8	9	**6,6**
R.NH.K2	5	15	12	6	8	6	8	6	8	6	8	7	8	7	13	7	15	0	16	6	15	10	15	11	**9,1**
J.NH.K3	5	15	11	5	7	5	7	5	7	5	7	6	7	6	10	7	11	4	0	5	7	7	8	8	**6,9**
R.NH.K3	5	15	12	9	10	6	10	6	10	6	10	7	10	7	13	9	16	14	15	0	14	14	16	13	**10,3**
J.NH.T0	5	16	12	5	8	7	8	7	8	7	8	8	8	8	11	7	10	5	13	6	0	6	9	6	**7,8**
R.NH.T0	6	14	12	5	6	7	6	7	6	7	6	8	6	8	12	8	14	10	13	6	14	0	12	7	**8,3**
J.NH.T1	5	15	10	4	6	5	6	5	6	5	6	6	6	6	11	7	12	5	12	4	11	8	0	8	**7,0**
R.NH.T1	5	13	10	6	7	7	7	7	7	7	7	9	7	9	12	7	11	9	12	7	14	6	12	0	**8,3**
Av.	**5,0**	**13,8**	**11,5**	**7,8**	**6,2**	**5,9**	**6,2**	**5,9**	**6,2**	**5,9**	**6,2**	**7,0**	**6,2**	**7,0**	**10,7**	**8,8**	**12,5**	**10,1**	**12,3**	**8,9**	**11,3**	**10,5**	**12,1**	**10,6**	

MICRO-PRECISÃO

	Lo.c	G.c	Ra.H	Ra.NH	J.H.K1	R.H.K1	J.H.K2	R.H.K2	J.H.K3	R.H.K3	J.H.T0	R.H.T0	J.H.T1	R.H.T1	J.NH.K1	R.NH.K1	J.NH.K2	R.NH.K2	J.NH.K3	R.NH.K3	J.NH.T0	R.NH.T0	J.NH.T1	R.NH.T1	Av.
Lo.c	0	12	10	12	9	10	9	10	9	10	9	10	9	10	12	11	12	13	13	12	11	12	13	12	**10,4**
G.c	8	0	12	9	8	10	8	10	8	10	8	11	8	11	8	7	8	7	8	8	8	9	8	9	**8,4**
Ra.H	10	8	0	7	7	6	7	6	7	6	7	6	7	6	6	6	10	6	8	6	6	7	6	7	**6,6**
Ra.Nh	8	11	13	0	8	10	8	10	8	10	8	10	8	10	11	8	12	10	14	8	10	12	12	12	**9,6**
J.H.K1	11	12	13	12	0	8	0	8	0	8	0	8	0	8	12	10	15	12	14	11	12	13	13	12	**8,8**
R.H.K1	10	10	14	10	6	0	6	0	6	0	6	3	6	3	10	9	12	10	13	10	10	10	11	10	**7,7**
J.H.K2	11	12	13	12	0	8	0	8	0	8	0	8	0	8	12	10	15	12	14	11	12	13	13	12	**8,8**
R.H.K2	10	10	14	10	6	0	6	0	6	0	6	3	6	3	10	9	12	10	13	10	10	10	11	10	**7,7**
J.H.K3	11	12	13	12	0	8	0	8	0	8	0	8	0	8	12	10	15	12	14	11	12	13	13	12	**8,8**
R.H.K3	10	10	14	10	6	0	6	0	6	0	6	3	6	3	10	9	12	10	13	10	10	10	11	10	**7,7**
J.H.T0	11	12	13	12	0	8	0	8	0	8	0	8	0	8	12	10	15	12	14	11	12	13	13	12	**8,8**
R.H.T0	10	9	14	10	7	2	7	2	7	2	7	0	7	0	10	9	12	10	12	9	10	10	11	10	**7,8**
J.H.T1	11	12	13	12	1	8	1	8	1	8	1	8	0	8	12	10	15	12	14	11	12	14	13	13	**9,1**
R.H.T1	10	9	14	10	7	2	7	2	7	2	7	0	7	0	10	9	12	10	12	9	10	10	11	10	**7,8**
J.NH.K1	8	12	14	9	8	10	8	10	8	10	8	10	8	10	0	8	13	7	13	9	10	8	10	8	**9,1**
R.NH.K1	9	13	14	12	10	11	10	11	10	11	10	11	10	11	12	0	15	14	15	11	12	17	12	15	**11,5**
J.NH.K2	8	12	10	8	5	8	5	8	5	8	5	8	5	8	7	5	0	6	10	7	9	9	8	9	**7,2**
R.NH.K2	7	13	14	10	8	10	8	10	8	10	8	10	8	10	13	6	14	0	12	9	12	15	12	12	**10,0**
J.NH.K3	7	12	12	6	6	7	6	7	6	7	6	8	6	8	7	5	10	8	0	5	7	8	7	8	**7,0**
R.NH.K3	8	12	14	12	9	10	9	10	9	10	9	11	9	11	11	9	13	11	15	0	9	15	11	14	**10,5**
J.NH.T0	9	12	14	10	8	10	8	10	8	10	8	10	8	10	10	8	11	8	13	11	0	8	13	11	**9,5**
R.NH.T0	8	11	13	8	7	10	7	10	7	10	7	10	6	10	12	3	11	5	12	5	12	0	10	6	**8,3**
J.NH.T1	7	12	14	8	7	9	7	9	7	9	7	9	7	9	10	8	12	8	13	9	7	10	0	9	**8,6**
R.NH.T1	8	11	13	8	8	10	8	10	8	10	8	10	7	10	12	5	11	8	12	6	9	7	11	0	**8,8**
Av.	**8,8**	**10,8**	**12,6**	**9,5**	**5,9**	**7,3**	**5,9**	**7,3**	**5,9**	**7,3**	**5,9**	**7,6**	**5,8**	**7,6**	**10,0**	**7,7**	**12,0**	**9,2**	**12,1**	**8,7**	**9,7**	**10,5**	**10,5**	**10,1**	

Tabela 55 – Comparação Pareada Macro e Micro Revocação

MACRO-REVOCAÇÃO

	Lo.c	G.c	Ra.H	Ra.NH	J.H.K1	R.H.K1	J.H.K2	R.H.K2	J.H.K3	R.H.K3	J.H.T0	R.H.T0	J.H.T1	R.H.T1	J.NH.K1	R.NH.K1	J.NH.K2	R.NH.K2	J.NH.K3	R.NH.K3	J.NH.T0	R.NH.T0	J.NH.T1	R.NH.T1	Av.
Lo.c	**0**	18	15	17	15	14	15	14	15	14	15	14	15	14	17	16	17	16	16	18	17	17	17	17	**15,1**
G.c	2	**0**	5	2	2	5	2	5	2	5	2	5	2	5	5	6	5	6	5	5	3	8	4	8	**4,1**
Ra.H	5	15	**0**	10	8	10	8	10	8	10	8	10	8	10	10	10	10	10	10	10	10	12	10	11	**9,3**
Ra.Nh	3	18	10	**0**	10	8	10	8	10	8	10	9	10	9	17	12	15	13	17	14	15	18	17	17	**11,6**
J.H.K1	5	18	12	10	**0**	6	0	6	0	6	0	7	0	7	11	14	13	13	13	13	12	14	14	14	**8,7**
R.H.K1	6	15	10	12	8	**0**	8	0	8	0	8	4	8	4	14	13	13	15	13	16	13	16	15	16	**9,8**
J.H.K2	5	18	12	10	0	6	**0**	6	0	6	0	7	0	7	11	14	13	13	13	13	12	14	14	14	**8,7**
R.H.K2	6	15	10	12	8	0	8	**0**	8	0	8	4	8	4	14	13	13	15	13	16	13	16	15	16	**9,8**
J.H.K3	5	18	12	10	0	6	0	6	**0**	6	0	7	0	7	11	14	13	13	13	13	12	14	14	14	**8,7**
R.H.K3	6	15	10	12	8	0	8	0	8	**0**	8	4	8	4	14	13	13	15	13	16	13	16	15	16	**9,8**
J.H.T0	5	18	12	10	0	6	0	6	0	6	**0**	7	0	7	11	14	13	13	13	13	12	14	14	14	**8,7**
R.H.T0	6	15	10	11	8	1	8	1	8	1	8	**0**	8	0	13	12	13	14	13	14	13	14	14	14	**9,1**
J.H.T1	5	18	12	10	1	6	1	6	1	6	1	7	**0**	7	11	14	13	13	13	13	12	14	14	14	**8,8**
R.H.T1	6	15	10	11	8	1	8	1	8	1	8	0	8	**0**	13	12	13	14	13	14	13	14	14	14	**9,1**
J.NH.K1	3	15	10	3	9	6	9	6	9	6	9	7	9	7	**0**	9	8	9	9	10	7	13	9	12	**8,1**
R.NH.K1	4	14	10	8	6	7	6	7	6	7	6	8	6	8	11	**0**	12	11	12	13	11	13	14	13	**8,9**
J.NH.K2	3	15	10	5	7	7	7	7	7	7	7	7	7	7	12	8	**0**	9	11	9	10	14	11	13	**8,3**
R.NH.K2	4	14	10	7	7	5	7	5	7	5	7	6	7	6	11	9	11	**0**	9	12	9	12	13	11	**8,1**
J.NH.K3	4	15	10	3	7	7	7	7	7	7	7	7	7	7	11	8	9	11	**0**	11	5	10	10	12	**7,9**
R.NH.K3	2	15	10	6	7	4	7	4	7	4	7	6	7	6	10	7	11	8	9	**0**	9	12	12	11	**7,5**
J.NH.T0	3	17	10	5	8	7	8	7	8	7	8	7	8	7	13	9	10	11	15	11	**0**	14	14	14	**9,2**
R.NH.T0	3	12	8	2	6	4	6	4	6	4	6	6	6	6	7	7	6	8	10	8	6	**0**	9	8	**6,2**
J.NH.T1	3	16	10	3	6	5	6	5	6	5	6	6	6	6	11	6	9	7	10	8	6	11	**0**	10	**7,0**
R.NH.T1	3	12	9	3	6	4	6	4	6	4	6	6	6	6	8	7	7	9	8	9	6	5	10	**0**	**6,3**
Av.	**4,0**	**15,0**	**9,9**	**7,6**	**6,0**	**5,2**	**6,0**	**5,2**	**6,0**	**5,2**	**6,0**	**6,3**	**6,0**	**6,3**	**11,1**	**10,3**	**10,8**	**11,1**	**11,3**	**11,6**	**10,0**	**12,7**	**12,2**	**12,6**	

MICRO-REVOCAÇÃO

	Lo.c	G.c	Ra.H	Ra.NH	J.H.K1	R.H.K1	J.H.K2	R.H.K2	J.H.K3	R.H.K3	J.H.T0	R.H.T0	J.H.T1	R.H.T1	J.NH.K1	R.NH.K1	J.NH.K2	R.NH.K2	J.NH.K3	R.NH.K3	J.NH.T0	R.NH.T0	J.NH.T1	R.NH.T1	Av.
Lo.c	**0**	17	14	17	15	14	15	14	15	14	15	14	15	14	17	17	16	16	16	17	17	16	18	16	**15,0**
G.c	3	**0**	5	2	1	1	1	1	1	1	1	2	1	2	3	2	2	2	0	2	1	4	2	4	**1,8**
Ra.H	6	15	**0**	9	10	11	10	11	10	11	10	11	10	11	12	10	10	10	10	9	11	13	10	13	**10,1**
Ra.Nh	3	18	11	**0**	9	8	9	8	9	8	9	9	9	9	16	11	15	15	16	12	14	15	17	14	**11,0**
J.H.K1	5	19	10	11	**0**	5	0	5	0	5	0	6	0	6	13	11	13	12	13	10	12	13	12	13	**8,1**
R.H.K1	6	19	9	12	9	**0**	9	0	9	0	9	4	9	4	13	13	13	13	15	13	12	14	14	15	**9,8**
J.H.K2	5	19	10	11	0	5	**0**	5	0	5	0	6	0	6	13	11	13	12	13	10	12	13	12	13	**8,1**
R.H.K2	6	19	9	12	9	0	9	**0**	9	0	9	4	9	4	13	13	13	13	15	13	12	14	14	15	**9,8**
J.H.K3	5	19	10	11	0	5	0	5	**0**	5	0	6	0	6	13	11	13	12	13	10	12	13	12	13	**8,1**
R.H.K3	6	19	9	12	9	0	9	0	9	**0**	9	4	9	4	13	13	13	13	15	13	12	14	14	15	**9,8**
J.H.T0	5	19	10	11	0	5	0	5	0	5	**0**	6	0	6	13	11	13	12	13	10	12	13	12	13	**8,1**
R.H.T0	6	18	9	11	9	1	9	1	9	1	9	**0**	9	0	13	12	12	12	13	11	11	13	13	14	**9,0**
J.H.T1	5	19	10	11	1	5	1	5	1	5	1	6	**0**	6	13	11	13	12	13	10	12	14	12	14	**8,3**
R.H.T1	6	18	9	11	9	1	9	1	9	1	9	0	9	**0**	13	12	12	12	13	11	11	13	13	14	**9,0**
J.NH.K1	3	17	8	4	7	7	7	7	7	7	7	7	7	7	**0**	8	8	6	11	7	9	10	10	9	**7,5**
R.NH.K1	3	18	10	9	9	7	9	7	9	7	9	8	9	8	12	**0**	12	12	14	11	12	14	14	13	**9,8**
J.NH.K2	4	18	10	5	7	7	7	7	7	7	7	8	7	8	12	8	**0**	7	11	9	10	10	12	8	**8,2**
R.NH.K2	4	18	10	5	8	7	8	7	8	7	8	8	8	8	14	8	13	**0**	13	10	12	13	14	10	**9,2**
J.NH.K3	4	20	10	4	7	5	7	5	7	5	7	7	7	7	9	6	9	7	**0**	6	6	7	11	8	**7,1**
R.NH.K3	3	18	11	8	10	7	10	7	10	7	10	9	10	9	13	9	11	10	14	**0**	12	13	13	12	**9,8**
J.NH.T0	3	19	9	6	8	8	8	8	8	8	8	9	8	9	11	8	10	8	14	8	**0**	9	13	10	**8,8**
R.NH.T0	4	16	7	5	7	6	7	6	7	6	7	7	6	7	10	6	10	7	13	7	11	**0**	12	5	**7,5**
J.NH.T1	2	18	10	3	8	6	8	6	8	6	8	7	8	7	10	6	8	6	9	7	7	8	**0**	8	**7,3**
R.NH.T1	4	16	7	6	7	5	7	5	7	5	7	6	6	6	11	7	12	10	12	8	10	8	12	**0**	**7,7**
Av.	**4,2**	**17,3**	**9,0**	**8,2**	**6,6**	**5,3**	**6,6**	**5,3**	**6,6**	**5,3**	**6,6**	**6,4**	**6,5**	**6,4**	**11,7**	**9,3**	**11,0**	**10,0**	**12,0**	**9,3**	**10,4**	**11,4**	**11,9**	**11,2**	

Tabela 56 – Comparação Pareada Macro e Micro F1

MACRO-F1

	Lo.c	G.c	Ra.H	Ra.NH	J.H.K1	R.H.K1	J.H.K2	R.H.K2	J.H.K3	R.H.K3	J.H.T0	R.H.T0	J.H.T1	R.H.T1	J.NH.K1	R.NH.K1	J.NH.K2	R.NH.K2	J.NH.K3	R.NH.K3	J.NH.T0	R.NH.T0	J.NH.T1	R.NH.T1	Av.
Lo.c	0	17	17	18	17	16	17	16	17	16	17	16	17	16	17	17	18	18	17	17	17	17	17	17	**16,2**
G.c	3	0	6	2	5	5	5	5	5	5	5	6	4	6	1	2	2	2	1	2	2	4	2	4	**3,5**
Ra.H	3	14	0	10	8	9	8	9	8	9	8	9	8	9	10	9	10	8	10	10	10	10	10	10	**8,7**
Ra.Nh	2	18	10	0	9	7	9	7	9	7	9	8	9	8	16	12	16	14	16	11	15	16	17	13	**10,8**
J.H.K1	3	15	12	11	0	5	0	5	0	5	0	6	0	6	12	12	12	11	13	12	13	13	13	12	**8,0**
R.H.K1	4	15	11	13	9	0	9	0	9	0	9	4	9	4	14	15	15	14	14	17	14	13	16	14	**10,1**
J.H.K2	3	15	12	11	0	5	0	5	0	5	0	6	0	6	12	12	12	11	13	12	13	13	13	12	**8,0**
R.H.K2	4	15	11	13	9	0	9	0	9	0	9	4	9	4	14	15	15	14	14	17	14	13	16	14	**10,1**
J.H.K3	3	15	12	11	0	5	0	5	0	5	0	6	0	6	12	12	12	11	13	12	13	13	13	12	**8,0**
R.H.K3	4	15	11	13	9	0	9	0	9	0	9	4	9	4	14	15	15	14	14	17	14	13	16	14	**10,1**
J.H.T0	3	15	12	11	0	5	0	5	0	5	0	6	0	6	12	12	12	11	13	12	13	13	13	12	**8,0**
R.H.T0	4	14	11	12	9	1	9	1	9	1	9	0	9	0	13	15	14	14	13	15	13	12	15	13	**9,4**
J.H.T1	3	16	12	11	1	5	1	5	1	5	1	6	0	6	12	12	12	11	13	12	13	13	13	12	**8,2**
R.H.T1	4	14	11	12	9	1	9	1	9	1	9	0	9	0	13	15	14	14	13	15	13	12	15	13	**9,4**
J.NH.K1	3	19	10	4	8	6	8	6	8	6	8	7	8	7	0	7	11	7	10	8	7	9	10	9	**7,8**
R.NH.K1	3	18	11	8	8	5	8	5	8	5	8	5	8	5	13	0	15	11	14	12	13	13	17	12	**9,4**
J.NH.K2	2	18	10	4	8	5	8	5	8	5	8	6	8	6	9	5	0	5	10	6	8	6	11	8	**7,0**
R.NH.K2	2	18	12	6	9	6	9	6	9	6	9	6	9	6	13	9	15	0	14	9	13	11	14	9	**9,2**
J.NH.K3	3	19	10	4	7	6	7	6	7	6	7	7	7	7	10	6	10	6	0	7	8	6	10	7	**7,2**
R.NH.K3	3	18	10	9	8	3	8	3	8	3	8	5	8	5	12	8	14	11	13	0	12	12	16	11	**8,7**
J.NH.T0	3	18	10	5	7	6	7	6	7	6	7	7	7	7	13	7	12	7	12	8	0	8	13	8	**8,0**
R.NH.T0	3	16	10	4	7	7	7	7	7	7	7	8	7	8	11	7	14	9	14	8	12	0	13	6	**8,3**
J.NH.T1	3	18	10	3	7	4	7	4	7	4	7	5	7	5	10	3	9	6	10	4	7	7	0	7	**6,4**
R.NH.T1	3	16	10	7	8	6	8	6	8	6	8	7	8	7	11	8	12	11	13	9	12	7	13	0	**8,5**
Av.	**3,0**	**15,7**	**10,5**	**8,4**	**6,8**	**4,9**	**6,8**	**4,9**	**6,8**	**4,9**	**6,8**	**6,0**	**6,7**	**6,0**	**11,4**	**9,8**	**12,1**	**10,0**	**12,0**	**10,5**	**11,2**	**10,6**	**12,8**	**10,4**	

MICRO-F1

	Lo.c	G.c	Ra.H	Ra.NH	J.H.K1	R.H.K1	J.H.K2	R.H.K2	J.H.K3	R.H.K3	J.H.T0	R.H.T0	J.H.T1	R.H.T1	J.NH.K1	R.NH.K1	J.NH.K2	R.NH.K2	J.NH.K3	R.NH.K3	J.NH.T0	R.NH.T0	J.NH.T1	R.NH.T1	Av.
Lo.c	0	16	11	15	14	14	14	14	14	14	14	14	14	14	16	15	16	16	16	14	15	16	17	16	**14,1**
G.c	4	0	7	3	3	4	3	4	3	4	3	5	3	5	3	2	3	1	1	2	3	4	3	3	**3,2**
Ra.H	9	13	0	9	10	11	10	11	10	11	10	11	10	11	10	9	10	9	10	9	9	12	10	11	**9,8**
Ra.Nh	5	17	11	0	9	9	9	9	9	9	9	10	9	10	15	10	14	12	16	11	14	15	17	13	**10,9**
J.H.K1	6	17	10	11	0	5	0	5	0	5	0	6	0	6	10	10	13	11	11	9	12	12	12	11	**7,6**
R.H.K1	6	16	9	11	9	0	9	0	9	0	9	4	9	4	10	12	13	11	13	11	12	13	12	13	**9,0**
J.H.K2	6	17	10	11	0	5	0	5	0	5	0	6	0	6	10	10	13	11	11	9	12	12	12	11	**7,6**
R.H.K2	6	16	9	11	9	0	9	0	9	0	9	4	9	4	10	12	13	11	13	11	12	13	12	13	**9,0**
J.H.K3	6	17	10	11	0	5	0	5	0	5	0	6	0	6	10	10	13	11	11	9	12	12	12	11	**7,6**
R.H.K3	6	16	9	11	9	0	9	0	9	0	9	4	9	4	10	12	13	11	13	11	12	13	12	13	**9,0**
J.H.T0	6	17	10	11	0	5	0	5	0	5	0	6	0	6	10	10	13	11	11	9	12	12	12	11	**7,6**
R.H.T0	6	15	9	10	9	1	9	1	9	1	9	0	9	0	9	11	12	10	11	9	11	12	11	12	**8,2**
J.H.T1	6	17	10	11	1	5	1	5	1	5	1	6	0	6	10	10	13	11	11	9	12	13	12	12	**7,8**
R.H.T1	6	15	9	10	9	1	9	1	9	1	9	0	9	0	9	11	12	10	11	9	11	12	11	12	**8,2**
J.NH.K1	4	17	10	5	10	10	10	10	10	10	10	11	10	11	0	7	11	6	11	7	9	10	10	8	**9,0**
R.NH.K1	5	18	11	10	10	8	10	8	10	8	10	9	10	9	13	0	15	13	14	13	12	16	15	14	**10,9**
J.NH.K2	4	17	10	6	7	7	7	7	7	7	7	8	7	8	9	5	0	6	11	7	8	8	9	8	**7,5**
R.NH.K2	4	19	11	8	9	9	9	9	9	9	9	10	9	10	14	7	14	0	14	11	12	14	13	11	**10,2**
J.NH.K3	4	19	10	4	9	7	9	7	9	7	9	9	9	9	9	6	9	6	0	5	8	7	9	7	**7,8**
R.NH.K3	6	18	11	9	11	9	11	9	11	9	11	11	11	11	13	7	13	9	15	0	12	13	13	12	**10,6**
J.NH.T0	5	17	11	6	8	8	8	8	8	8	8	9	8	9	11	8	12	8	12	8	0	10	12	10	**8,8**
R.NH.T0	4	16	8	5	8	7	8	7	8	7	8	8	7	8	10	4	12	6	13	7	10	0	12	6	**7,9**
J.NH.T1	3	17	10	3	8	8	8	8	8	8	8	9	8	9	10	5	11	7	11	7	8	8	0	7	**7,9**
R.NH.T1	4	17	9	7	9	7	9	7	9	7	9	8	8	8	12	6	12	9	13	8	10	7	13	0	**8,7**
Av.	**5,0**	**16,0**	**9,4**	**8,3**	**7,1**	**6,0**	**7,1**	**6,0**	**7,1**	**6,0**	**7,1**	**7,3**	**7,0**	**7,3**	**10,1**	**8,3**	**11,7**	**9,0**	**11,4**	**8,5**	**10,3**	**11,0**	**11,3**	**10,2**	

Tabela 57 – Comparação Pareada CLP e MLP

CLP

	Lo.c	G.c	Ra.H	Ra.NH	J.H.K1	R.H.K1	J.H.K2	R.H.K2	J.H.K3	R.H.K3	J.H.T0	R.H.T0	J.H.T1	R.H.T1	J.NH.K1	R.NH.K1	J.NH.K2	R.NH.K2	J.NH.K3	R.NH.K3	J.NH.T0	R.NH.T0	J.NH.T1	R.NH.T1	Av.
Lo.c	**0**	12	7	3	9	9	9	9	9	9	8	10	8	10	9	7	8	9	10	6	9	9	9	6	**8,1**
G.c	1	**0**	3	1	2	0	2	0	2	0	1	1	1	1	1	1	1	1	0	1	0	0	0	1	**0,9**
Ra.H	1	10	**0**	3	3	3	3	3	3	3	2	4	2	4	3	2	3	4	4	4	4	3	4	3	**3,3**
Ra.Nh	2	12	5	**0**	8	7	8	7	8	7	7	9	7	9	7	4	8	6	8	4	8	6	8	5	**6,7**
J.H.K1	1	12	5	2	**0**	2	0	2	0	2	0	3	0	3	4	3	4	3	6	4	6	1	6	2	**3,0**
R.H.K1	1	13	6	3	7	**0**	7	0	7	0	6	3	6	3	5	4	4	4	5	4	5	3	5	3	**4,3**
J.H.K2	1	12	5	2	0	2	**0**	2	0	2	0	3	0	3	4	3	4	3	6	4	6	1	6	2	**3,0**
R.H.K2	1	13	6	3	7	0	7	**0**	7	0	6	3	6	3	5	4	4	4	5	4	5	3	5	3	**4,3**
J.H.K3	1	12	5	2	0	2	0	2	**0**	2	0	3	0	3	4	3	4	3	6	4	6	1	6	2	**3,0**
R.H.K3	1	13	6	3	7	0	7	0	7	**0**	6	3	6	3	5	4	4	4	5	4	5	3	5	3	**4,3**
J.H.T0	1	12	5	2	1	2	1	2	1	2	**0**	3	0	3	4	3	4	3	6	4	6	1	6	2	**3,1**
R.H.T0	1	13	6	2	7	1	7	1	7	1	6	**0**	6	0	6	3	5	3	5	4	6	2	6	2	**4,2**
J.H.T1	1	12	5	2	1	2	1	2	1	2	0	3	**0**	3	4	3	4	3	6	4	6	1	6	2	**3,1**
R.H.T1	1	13	6	2	7	1	7	1	7	1	6	0	6	**0**	6	3	5	3	5	4	6	2	6	2	**4,2**
J.NH.K1	1	12	5	2	5	4	5	4	5	4	4	4	4	4	**0**	2	5	1	4	2	2	0	3	1	**3,5**
R.NH.K1	1	12	4	1	7	6	7	6	7	6	6	7	6	7	6	**0**	7	4	7	4	7	3	7	3	**5,5**
J.NH.K2	2	12	5	1	6	6	6	6	6	6	5	6	5	6	3	1	**0**	1	5	2	2	1	3	2	**4,1**
R.NH.K2	1	12	4	3	7	5	7	5	7	5	6	6	6	6	6	2	7	**0**	7	3	7	1	7	2	**5,1**
J.NH.K3	1	13	5	2	5	4	5	4	5	4	4	5	4	5	3	2	2	1	**0**	1	3	0	3	1	**3,4**
R.NH.K3	1	12	4	2	6	6	6	6	6	6	5	7	5	7	7	3	7	5	7	**0**	7	5	7	2	**5,4**
J.NH.T0	2	13	5	2	5	6	5	6	5	6	4	6	4	6	5	2	5	2	7	2	**0**	2	3	2	**4,4**
R.NH.T0	1	13	6	3	9	6	9	6	9	6	8	7	8	7	8	4	8	3	8	5	8	**0**	8	2	**6,3**
J.NH.T1	2	13	5	2	5	6	5	6	5	6	4	6	4	6	3	2	3	2	5	2	0	2	**0**	2	**4,0**
R.NH.T1	1	12	5	1	8	6	8	6	8	6	7	7	7	7	8	4	7	6	8	3	7	5	7	**0**	**6,0**
Av.	**1,1**	**11,8**	**4,9**	**2,0**	**5,1**	**3,6**	**5,1**	**3,6**	**5,1**	**3,6**	**4,2**	**4,5**	**4,2**	**4,5**	**4,8**	**2,9**	**4,7**	**3,3**	**5,6**	**3,3**	**5,0**	**2,3**	**5,3**	**2,3**	

MLP

	Lo.c	G.c	Ra.H	Ra.NH	J.H.K1	R.H.K1	J.H.K2	R.H.K2	J.H.K3	R.H.K3	J.H.T0	R.H.T0	J.H.T1	R.H.T1	J.NH.K1	R.NH.K1	J.NH.K2	R.NH.K2	J.NH.K3	R.NH.K3	J.NH.T0	R.NH.T0	J.NH.T1	R.NH.T1	Av.
Lo.c	**0**	14	11	14	14	15	14	15	14	15	14	15	14	15	14	14	16	14	15	13	16	14	15	14	**13,7**
G.c	3	**0**	2	2	0	3	0	3	0	3	0	3	0	3	2	2	3	2	2	2	3	2	2	3	**1,9**
Ra.H	2	10	**0**	7	5	8	5	8	5	8	5	8	5	8	7	7	9	7	8	7	9	8	8	9	**6,8**
Ra.Nh	2	15	8	**0**	3	4	3	4	3	4	3	5	3	5	12	7	11	9	13	8	15	11	13	9	**7,1**
J.H.K1	3	15	11	13	**0**	7	0	7	0	7	0	8	0	8	13	12	13	12	13	12	14	15	12	15	**8,8**
R.H.K1	1	14	7	11	6	**0**	6	0	6	0	6	4	6	4	12	11	12	12	13	12	13	12	12	12	**8,0**
J.H.K2	3	15	11	13	0	7	**0**	7	0	7	0	8	0	8	13	12	13	12	13	12	14	15	12	15	**8,8**
R.H.K2	1	14	7	11	6	0	6	**0**	6	0	6	4	6	4	12	11	12	12	13	12	13	12	12	12	**8,0**
J.H.K3	3	15	11	13	0	7	0	7	**0**	7	0	8	0	8	13	12	13	12	13	12	14	15	12	15	**8,8**
R.H.K3	1	14	7	11	6	0	6	0	6	**0**	6	4	6	4	12	11	12	12	13	12	13	12	12	12	**8,0**
J.H.T0	3	15	11	13	0	7	0	7	0	7	**0**	8	0	8	13	12	13	12	13	12	14	15	12	15	**8,8**
R.H.T0	1	13	7	10	6	1	6	1	6	1	6	**0**	6	0	11	10	11	11	12	11	12	11	11	11	**7,3**
J.H.T1	3	15	11	13	0	7	0	7	0	7	0	8	**0**	8	13	12	13	12	13	12	14	15	12	15	**8,8**
R.H.T1	1	13	7	10	6	1	6	1	6	1	6	0	6	**0**	11	10	11	11	12	11	12	11	11	11	**7,3**
J.NH.K1	2	16	9	4	5	5	5	5	5	5	5	6	5	6	**0**	5	8	5	8	5	8	6	7	7	**5,9**
R.NH.K1	1	15	8	6	5	4	5	4	5	4	5	5	5	5	11	**0**	13	9	13	6	13	8	13	8	**7,1**
J.NH.K2	1	15	8	4	3	2	3	2	3	2	3	3	3	3	6	3	**0**	3	8	4	6	5	6	7	**4,3**
R.NH.K2	1	15	8	5	5	4	5	4	5	4	5	5	5	5	11	3	13	**0**	13	4	14	4	12	6	**6,5**
J.NH.K3	2	16	9	3	4	3	4	3	4	3	4	4	4	4	6	4	6	4	**0**	5	6	4	4	7	**4,7**
R.NH.K3	1	15	7	6	5	3	5	3	5	3	5	4	5	4	11	6	12	8	12	**0**	12	8	11	7	**6,6**
J.NH.T0	1	14	7	0	3	2	3	2	3	2	3	3	3	3	6	3	4	3	7	4	**0**	2	5	4	**3,6**
R.NH.T0	1	15	7	3	2	3	2	3	2	3	2	4	2	4	10	3	11	7	13	5	14	**0**	12	5	**5,5**
J.NH.T1	3	15	8	3	5	5	5	5	5	5	5	6	5	6	9	5	9	5	10	6	9	6	**0**	7	**6,1**
R.NH.T1	1	14	6	3	2	3	2	3	2	3	2	4	2	4	8	3	9	6	10	5	12	4	10	**0**	**4,9**
Av.	**1,7**	**13,8**	**7,8**	**7,4**	**3,8**	**4,2**	**3,8**	**4,2**	**3,8**	**4,2**	**3,8**	**5,3**	**3,8**	**5,3**	**9,8**	**7,4**	**10,3**	**8,3**	**10,8**	**8,0**	**11,3**	**9,0**	**9,8**	**9,4**	

Tabela 58 – Comparação Pareada WLP

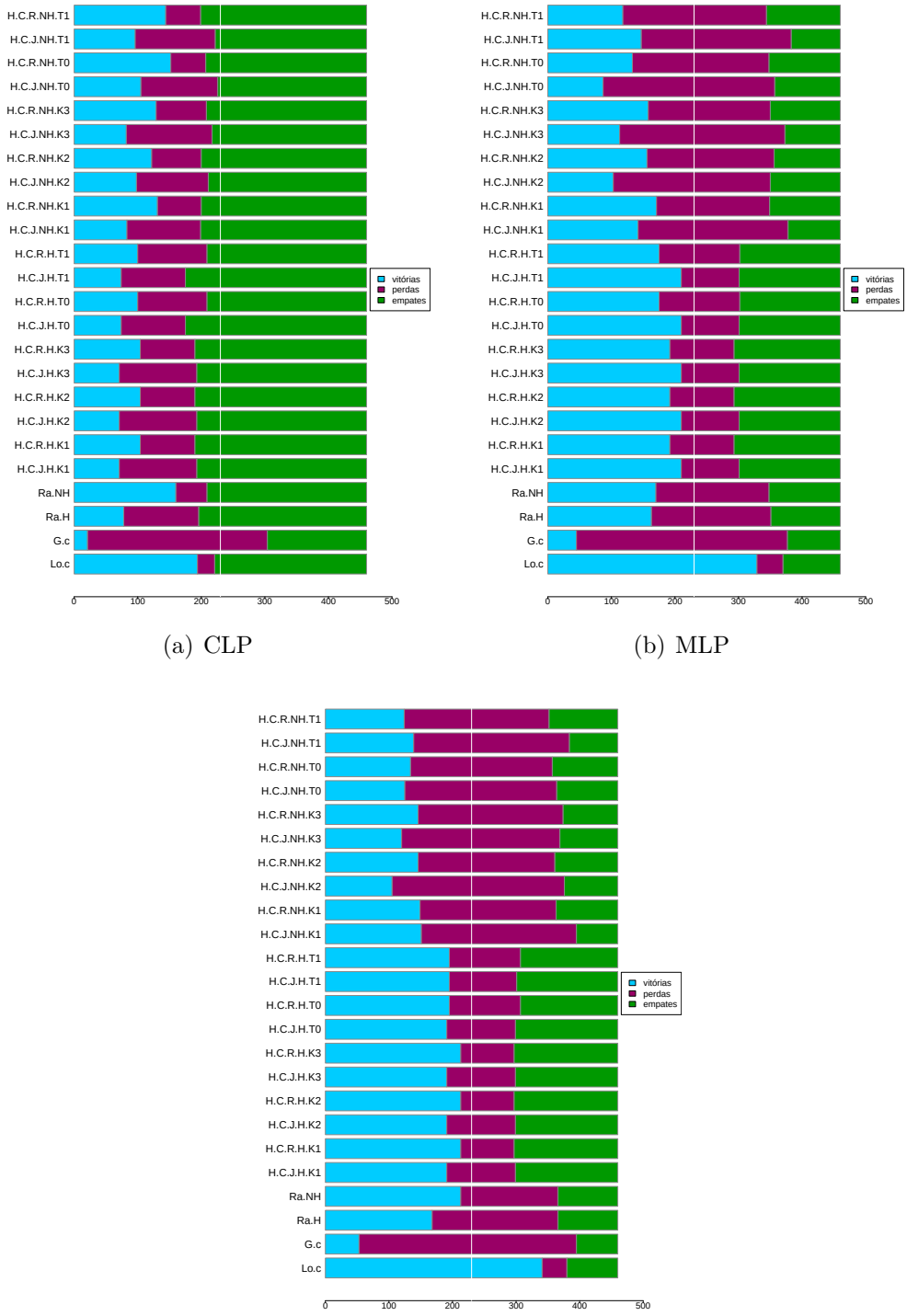
WLP																									
	Lo.c	**G.c**	**Ra.H**	**Ra.NH**	**J.H.K1**	**R.H.K1**	**J.H.K2**	**R.H.K2**	**J.H.K3**	**R.H.K3**	**J.H.T0**	**R.H.T0**	**J.H.T1**	**R.H.T1**	**J.NH.K1**	**R.NH.K1**	**J.NH.K2**	**R.NH.K2**	**J.NH.K3**	**R.NH.K3**	**J.NH.T0**	**R.NH.T0**	**J.NH.T1**	**R.NH.T1**	**Av.**
Lo.c	0	15	14	14	15	14	15	14	15	14	15	14	15	14	15	15	17	15	15	15	17	14	16	14	**14,2**
G.c	2	0	5	3	2	2	2	2	2	2	2	3	1	3	2	2	3	1	2	3	3	1	3	2	**2,2**
Ra.H	2	12	0	7	6	7	6	7	6	7	6	7	6	7	8	8	9	7	7	9	9	8	8	9	**7,0**
Ra.Nh	2	14	8	0	7	4	7	4	7	4	7	6	7	6	13	12	14	11	13	12	15	13	14	13	**8,9**
J.H.K1	2	15	10	10	0	6	0	6	0	6	0	7	0	7	12	12	13	12	12	12	11	13	12	13	**8,0**
R.H.K1	2	15	8	11	7	0	7	0	7	0	7	4	7	4	14	13	15	13	13	14	13	12	14	13	**8,9**
J.H.K2	2	15	10	10	0	6	0	6	0	6	0	7	0	7	12	12	13	12	12	12	11	13	12	13	**8,0**
R.H.K2	2	15	8	11	7	0	7	0	7	0	7	4	7	4	14	13	15	13	13	14	13	12	14	13	**8,9**
J.H.K3	2	15	10	10	0	6	0	6	0	6	0	7	0	7	12	12	13	12	12	12	11	13	12	13	**8,0**
R.H.K3	2	15	8	11	7	0	7	0	7	0	7	4	7	4	14	13	15	13	13	14	13	12	14	13	**8,9**
J.H.T0	2	15	10	10	0	6	0	6	0	6	0	7	0	7	12	12	13	12	12	12	11	13	12	13	**8,0**
R.H.T0	2	14	8	9	7	1	7	1	7	1	7	0	7	0	13	12	14	12	12	13	12	11	13	12	**8,1**
J.H.T1	2	15	10	10	1	6	1	6	1	6	1	7	0	7	12	12	13	12	12	12	11	13	12	13	**8,1**
R.H.T1	2	14	8	9	7	1	7	1	7	1	7	0	7	0	13	12	14	12	12	13	12	11	13	12	**8,1**
J.NH.K1	2	16	10	4	6	4	6	4	6	4	6	5	6	5	0	7	7	7	8	9	7	7	8	7	**6,3**
R.NH.K1	1	15	8	3	4	2	4	2	4	2	4	3	4	3	11	0	12	8	11	9	10	9	11	9	**6,2**
J.NH.K2	1	15	8	3	3	2	3	2	3	2	3	3	3	3	5	4	0	4	8	6	6	5	7	6	**4,4**
R.NH.K2	1	16	8	5	4	2	4	2	4	2	4	3	4	3	10	6	11	0	12	6	11	9	11	8	**6,1**
J.NH.K3	2	16	9	3	5	3	5	3	5	3	5	4	5	4	7	5	8	3	0	6	5	5	4	5	**5,0**
R.NH.K3	1	14	7	2	5	2	5	2	5	2	5	3	5	3	9	6	12	10	10	0	11	9	10	8	**6,1**
J.NH.T0	1	15	8	2	4	2	4	2	4	2	4	3	4	3	8	8	8	6	9	7	0	6	8	7	**5,2**
R.NH.T0	1	16	8	2	3	3	3	3	3	3	3	4	3	4	10	6	12	7	9	5	11	0	9	6	**5,6**
J.NH.T1	2	15	8	2	5	3	5	3	5	3	5	4	5	4	9	6	9	6	12	7	7	8	0	6	**5,8**
R.NH.T1	1	15	7	2	3	2	3	2	3	2	3	3	3	3	9	6	11	7	10	6	9	6	8	0	**5,2**
Av.	**1,6**	**14,3**	**8,3**	**6,4**	**4,5**	**3,5**	**4,5**	**3,5**	**4,5**	**3,5**	**4,5**	**4,7**	**4,4**	**4,7**	**10,2**	**8,9**	**11,3**	**9,0**	**10,4**	**9,5**	**10,0**	**9,3**	**10,2**	**9,5**	

partições locais não tiverem melhor desempenho do que todos os outros métodos, assim como o método global também não configura o pior. Na Micro-Precisão, o método NH.Ro.K1 obteve melhor desempenho preditivo que todos os outros métodos em uma média de 11,5 conjuntos de dados, e o método H.Ra o pior desempenho, apenas 6.6 conjuntos de dados na média.

Nas outras tabelas de comparação pareada, o método local obteve melhor desempenho preditivo em mais conjuntos de dados do que os outros métodos, e o global em menos conjuntos de dados. Além disso, os métodos usando Jaccard index foram melhores em mais conjuntos de dados que aqueles que usaram Rogers, e os métodos hierárquicos também obtiveram melhores resultados em mais conjuntos de dados do que os não hierárquicos. Outro comportamento notado nas tabelas pareadas foi o do método NH.Ra, que foi melhor em mais conjuntos de dados - na média - do que as partições híbridas nas medidas CLP, Macro-F1, Macro-Precisão, Macro-Revocação e Micro-Revocação. Por fim, em algumas medidas os métodos com sparcificação usando threshold foram melhores em mais conjuntos de dados (na média) do que os que usaram k-NN. Há casos em que, na mesma medida, métodos hierárquicos com threshold foram melhores em mais conjuntos de dados dos que os métodos hierárquicos com k-NN, e nos métodos não hierárquicos a situação é inversa. O contrário também ocorre, indicando que existe alta competitividade entre os próprios métodos híbridos.

Já os gráficos de vitórias-derrotas-empates são apresentados nas Figuras 71 a 73. As diferenças entre as tabelas de comparação pareada e gráficos de vitórias-derrotas-empatess foram explicadas na seção 6.1. No gráfico da medida CLP, confirma-se que as partições locais têm melhor desempenho preditivo que todos os outros métodos e as partições globais as piores. Vê-se uma disputa entre as partições híbridas e aleatórias, onde os métodos baseados na medida Rogers-Tanimoto venceram mais vezes que os métodos com jaccard. Esse mesmo comportamento é notado em todos os outros gráficos de vitórias-derrotas-empates, e em alguns casos, os métodos com jaccard são mais vitoriosos que os métodos com rogers. Por fim, as observações feitas para as tabelas pareadas são também coerentes com o que os gráficos de vitórias-derrotas-empates apresentam.

Com os métodos hierárquicos são geradas várias partições e então uma é escolhida como a mais adequada, o que não ocorre com os métodos não hierárquicos. Este pode ser um motivo que levou alguns conjuntos de dados a ter um melhor desempenho com partições geradas por métodos de detecção de comunidade hierárquicos quando comparado com os outros. Considerar ou não considerar a correlação nula entre rótulos também parece não afetar demasiadamente o resultado final, já que tanto as partições híbridas geradas com Jaccard quanto com Rogers foram capazes de melhorar o desempenho preditivo para alguns conjuntos de dados.

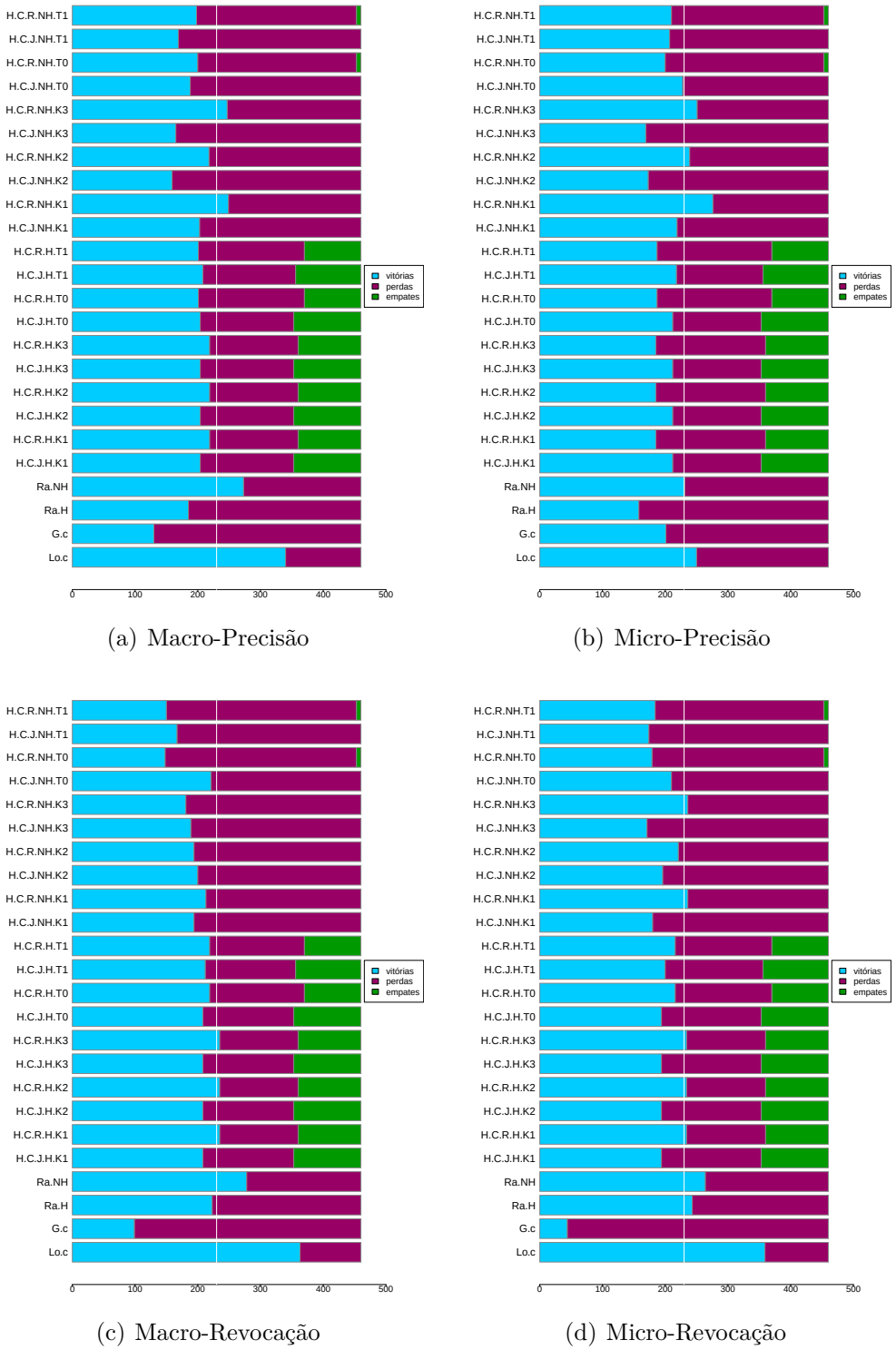


(a) CLP

(b) MLP

(c) WLP

Figura 71 – Gráficos de Vitórias, Derrotas e Empates para CLP, MLP e WLP. É possível notar que as partições locais tem o maior número de vitórias, enquanto as globais o menor. As partições híbridas e aleatórias são competitivas entre si e com as locais.



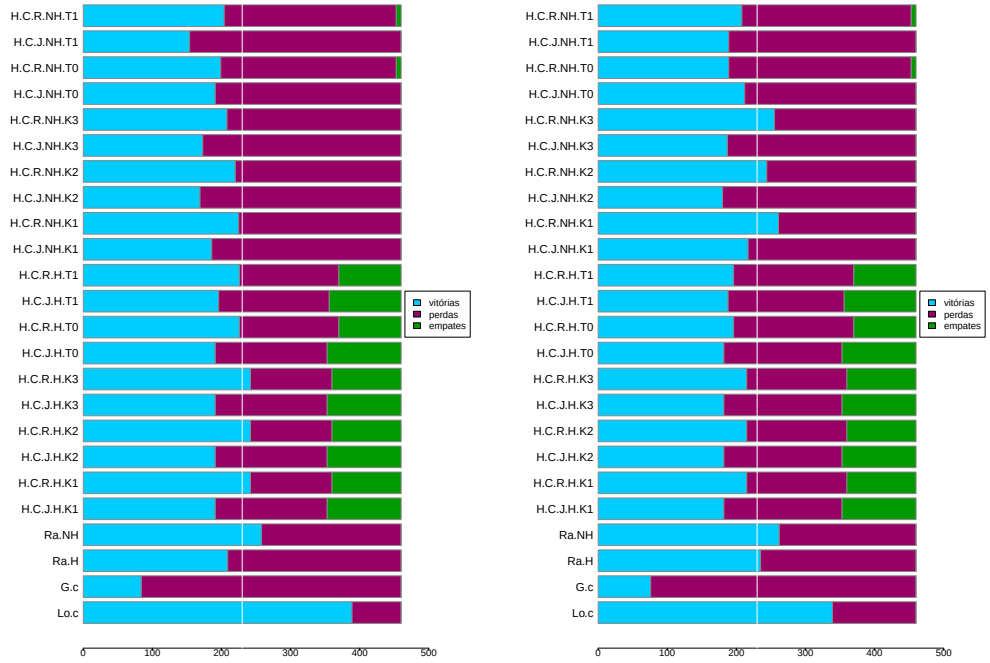
(a) Macro-Precisão

(b) Micro-Precisão

(c) Macro-Revocação

(d) Micro-Revocação

Figura 72 – Gráficos de Vitórias, Derrotas e Empates para Macro e Micro Precisão e Revocação. É possível notar que as partições locais tem o maior número de vitórias, enquanto as globais o menor. As partições híbridas e aleatórias são competitivas entre si e com as locais.



(a) Macro-F1 (b) Micro-F1

Figura 73 – Gráficos de Vitórias, Derrotas e Empates para Macro e Micro F1. É possível notar que as partições locais tem o maior número de vitórias, enquanto as globais o menor. As partições híbridas e aleatórias são competitivas entre si e com as locais.

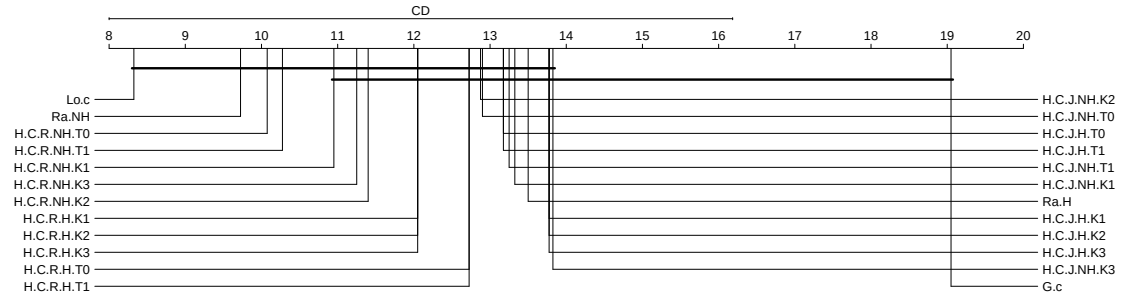
6.3.1 Teste Estatístico

Para verificar a significância estatística dos resultados, foi executado o teste de Friedman ($\alpha = 0,05$), seguido do teste post-hoc de Nemenyi. Eles são adequados ao comparar classificadores em vários conjuntos de dados (DEMSAR, 2006). Os respectivos valores de p de Friedman estão na Tabela 59, com uma distância crítica de 8,184. As Figuras de 74 à 76 apresentam os gráficos de distância crítica para as medidas de avaliação analisadas. Entre todas as medidas, apenas na Micro-Precisão não existe diferença estatística entre todos os métodos comparados, pois o gráfico mostra uma única linha conectando todos eles. Nas demais medidas, há diferenças estatísticas entre dois ou três grupos de métodos. Em todas as medidas analisadas, os resultados dos métodos Global e Local não estão conectados diretamente, indicando que existe diferença estatística significativa entre eles.

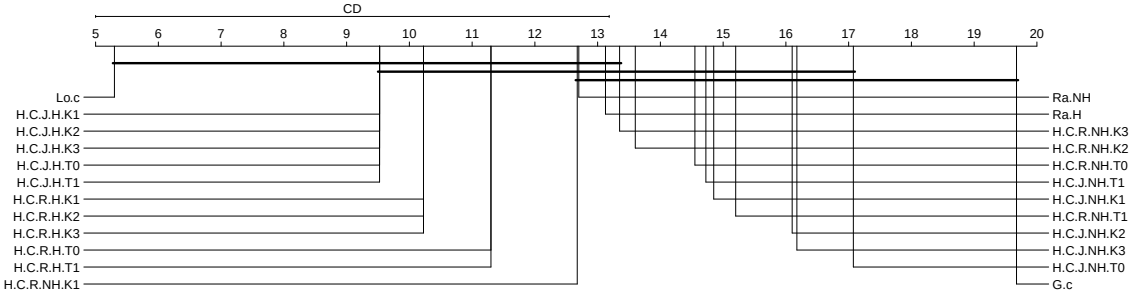
Nas medidas MLP, WLP, Macro-F1, Macro-Revocação, Micro-F1, e Micro-Rovocação, os gráficos mostram que não há diferenças estatísticas entre os resultados das partições aleatórias e híbridas. Nessas medidas, os métodos estão conectados em três grupos diferentes. O primeiro grupo conecta os resultados das partições locais com os outros resultados, o segundo grupo conecta os resultados das partições aleatórias e híbridas, e o terceiro grupo conecta os resultados do método global com outros. Sabe-se que não há diferença estatística entre os métodos conectados em uma linha.

Tabela 59 – Resultados do Teste de Friedman.

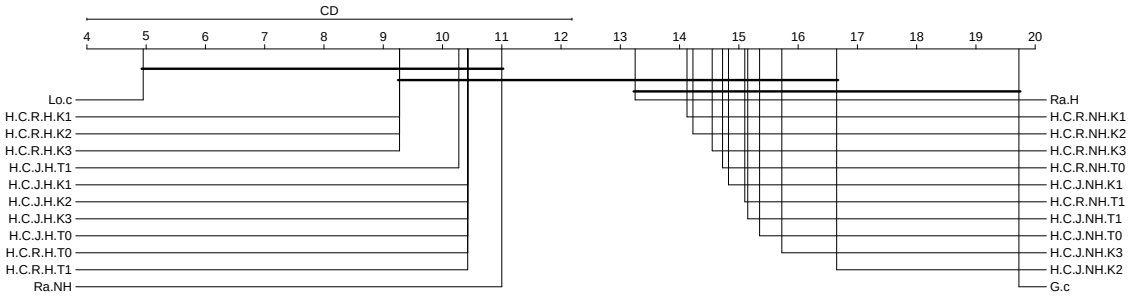
Medidas	**ChiSquare**	$p-$**Values**	**Hipótese**
CLP	37,935	0,025886943806181800	Ha:Different
MLP	95,095	0,000000000097734376	Ha:Different
WLP	97,482	0,000000000038182346	Ha:Different
Macro-Precision	52,763	0,000394669470347386	Ha:Different
Micro-Precision	24,387	0,382671915325202000	H0:Identical
Macro-Recall	77,643	0,000000076142308680	Ha:Different
Micro-Recall	79,830	0,000000033834945290	Ha:Different
Macro-F1	81,956	0,000000015271290787	Ha:Different
Micro-F1	50,838	0,000714514912773745	Ha:Different



(a) CLP

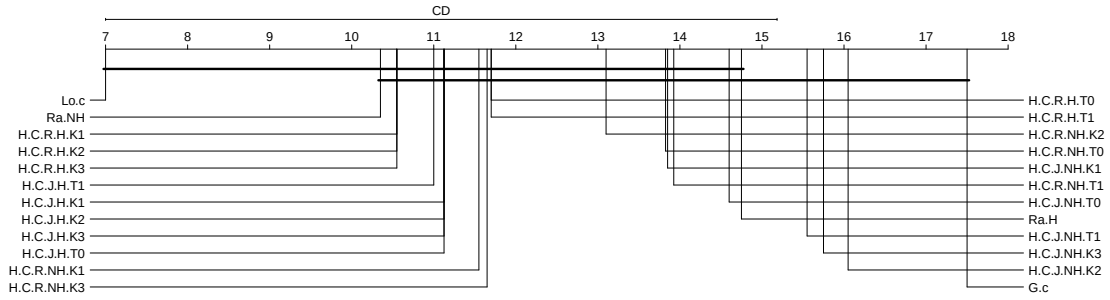


(b) MLP

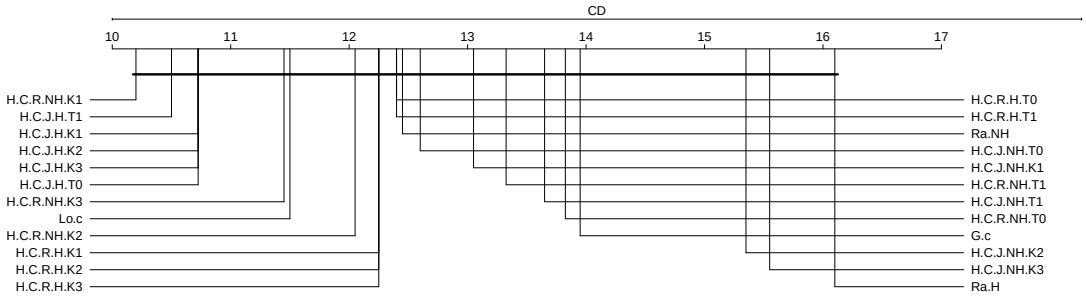


(c) WLP

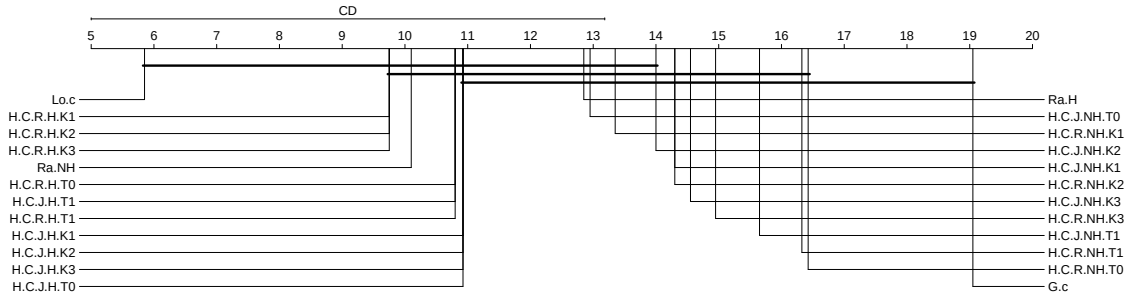
Figura 74 – Gráficos de Distância Crítica de Nemenyi para CLP, MLP e WLP.



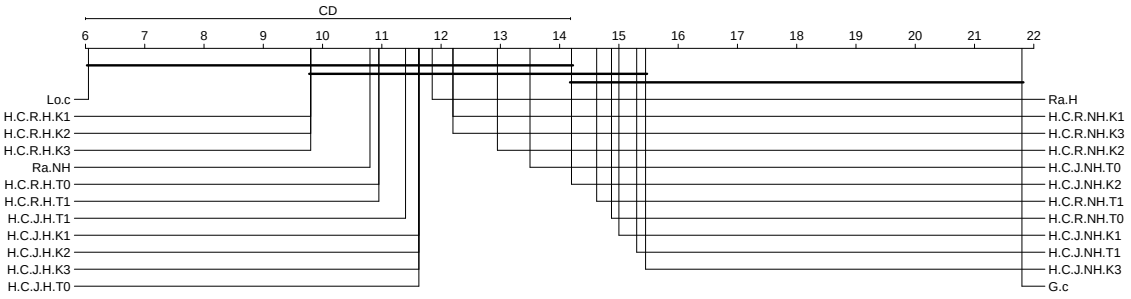
(a) Macro-Precisão



(b) Micro-Precisão

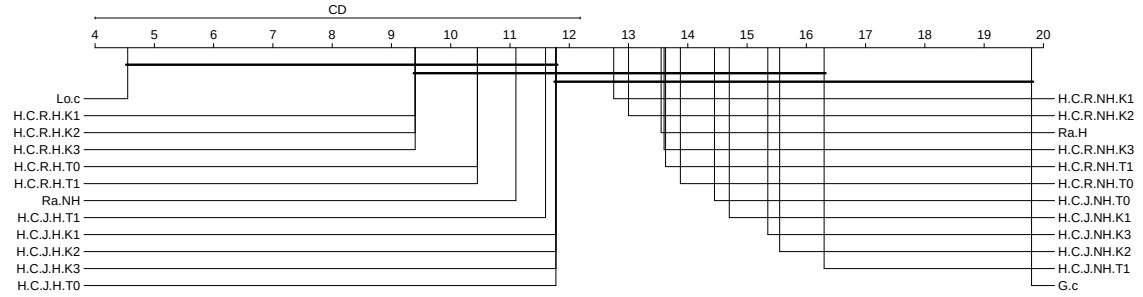


(c) Macro-Revocação

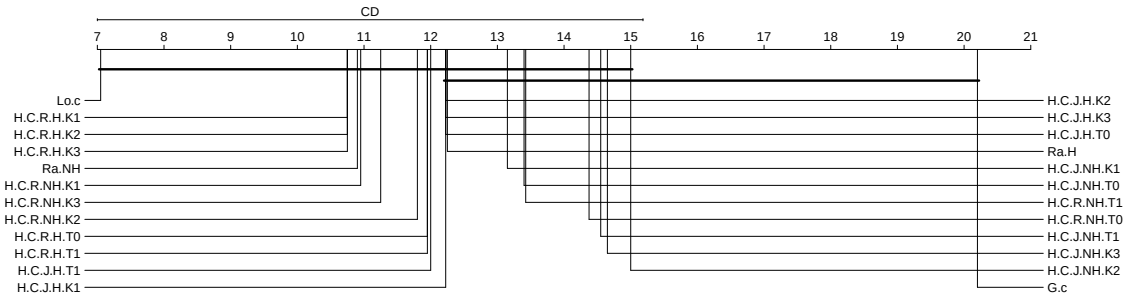


(d) Micro-Revocação

Figura 75 – Gráficos de Distância Crítica de Nemenyi para para Macro e Micro Precisão e Revocação.



(a) Macro-F1



(b) Micro-F1

Figura 76 – Gráficos de Distância Crítica de Nemenyi para Macro e Micro F1.

Dada esta observação, pode-se concluir que as diferenças estatísticas apresentadas nos gráficos são limitadas pelos métodos global e local. Na Micro-F1, por exemplo, onde há dois grupos que não possuem diferenças entre eles, no primeiro grupo todos os métodos estão conectados com o local, enquanto que no segundo grupo, boa parte dos métodos estão conectados com o global.

Como vários métodos estão conectados nos dois grupos, pode-se concluir que essa diferença estatística existente entre os grupos é baixa, o que pode ser constatado pelos valores da Tabela 59. Isto leva à conclusão de que os métodos locais e globais ainda precisam de melhorias, já que diferenças estatísticas significativas não foram encontradas entre eles, os métodos híbridos e aleatórios. Portanto, de acordo com tudo o que foi apresentado até aqui (desempenho, comparação pareada, vitórias-derrotas-empates), pode-se concluir que as abordagens locais e globais convencionais e amplamente utilizadas podem não estar aprendendo corretamente as correlações entre rótulos e/ou predizendo corretamente os rótulos, ainda que se utilize diferentes partições de dados.

6.3.2 Métodos de Detecção de Comunidade

Para a medida de similaridade de Rogers-Tanimoto, os métodos de detecção de comunidade hierárquicos mais escolhidos foram Edge Betweenness para esparsificação com threshold e WalkTrap para esparsificação k-NN. Já para o Jaccard Index, o Walktrap foi o mais escolhido em ambos os tipos de esparsificação. No caso dos métodos não hierár-

quicos, o InfoMap foi o método de detecção de comunidades mais escolhido, tanto para medidas de similaridade quanto para tipos de esparsificação. Por fim, para Comunidades Aleatórias, o método hierárquico mais escolhido foi o WalkTrap, enquanto o InfoMap foi o mais escolhido para os métodos não hierárquicos.

6.3.3 Análise das Partições

Um resumo de todas as partições híbridas e aleatórias escolhidas como as melhores é apresentado na Tabela 60. O número na coluna P. indica o número da partição. Para ilustrar, considere os conjuntos de dados birds, que possui 19 rótulos. A partição 1 é a partição global, enquanto a partição 19 é a partição local. As partições de número 2 a 18 são partições híbridas; portanto, a partição 2 é mais semelhante à partição global, enquanto a partição 18 é mais semelhante à partição local.

Para os conjuntos de dados *birds*, *emotions*, *EukaryotePseAAC* e *yeast*, as melhores partições escolhidas foram próximas à partição local. Em contraste, os conjuntos de dados *eisen*, *GnegativeGO*, *GpostiveGO*, *scene* e *Yelp* tiveram as melhores partições escolhidas próximas à partição global. Considerando o conjunto de dados *Langlog*, a partição escolhida com mais frequência foi semelhante à local. Em contraste, no conjunto de dados *pheno*, há um equilíbrio entre as partições próximas das globais ou locais.

Finalmente, as melhores partições para os outros conjuntos de dados variam, mas uma partição próxima à configuração da partição global foi escolhida com maior frequência. Observa-se pelos resultados que as melhores partições híbridas geradas pelos métodos de detecção de comunidade escolhidos são semelhantes à partição global. Esse pode ser um dos motivos pelos quais os resultados de desempenho das partições híbridas são competitivos em comparação com outras partições, superam o global e não são superiores aos locais para alguns conjuntos de dados.

6.4 Encadeamento

Os resultados deste experimento foram avaliados em sete diferentes medidas de avaliação multi-Label. As Florestas Aleatórias Multirrótulo produzem predições de probabilidades, e portanto, é possível calcular as curvas de precisão-revocação e curvas ROC (Receiver Operating Characteristic). As medidas de avaliação AUPRC-Macro e AUPRC-Micro foram calculadas a partir da implementação do Python[5], enquanto que as medidas ROC-AUC-Macro, ROC-AUC-Micro e ROC-AUC foram calculadas a partir da implementação do R[6].

Não são apresentados para este experimento, portanto, os resultados da Macro e Micro Revocação, Precisão e F1, mas sim para a AUPRC Macro e Micro, ROC-AUC-Macro e

[5] <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score.html>
[6] <https://github.com/rivolli/utiml>

Tabela 60 – Partições Escolhidas

	Híbridas		Aleatórias			Híbridas		Aleatórias	
Dataset	**P.**	**%**	**P.**	**%**	**Dataset**	**P.**	**%**	**P.**	**%**
birds	*18*	100%	*18*	100%	**GnegativeGO**	*2*	100%	*6*	100%
	2	80%			**GpositiveGO**	*2*	100%	*3*	100%
cal500	*4*	10%				*62*	10%	*2*	10%
	173	10%	*173*	100%		*68*	10%		
	2	50%	*2*	100%	**langlog**	*69*	10%		
	3	10%				*70*	10%		
	12	10%				*73*	10%		
cellcycle	*176*	10%				*74*	50%	*74*	90%
	177	20%				*2*	50%	*2*	100%
	2	40%	*2*	100%		*26*	10%		
	14	10%			**medical**	*41*	10%		
	23	10%				*43*	20%		
derisi	*173*	10%				*44*	10%		
	176	10%				*2*	40%	*2*	100%
	177	20%			**pheno**	*163*	20%		
eisen	*2*	100%	*2*	100%		*164*	40%		
emotions	*5*	100%	*2*	100%	**PlantGO**	*3*	100%	*2*	100%
EukaPseAAC	*21*	100%	*21*	100%	**scene**	*2*	100%	*2*	100%
flags	*4*	100%	*3*	100%		*2*	80%		
	2	30%	*2*	100%	**seq**	*177*	20%	*176*	100%
	50	10%			**VirusPseAAC**	*4*	100%	*2*	100%
	61	10%			**yeast**	*12*	100%	*13*	100%
gasch1	*70*	10%			**Yelp**	*2*	100%	*2*	100%
	175	10%							
	176	10%							
	177	20%							

Micro, e também as medidas CLP, MLP E WLP. O principal motivo é que, de acordo com os resultados dos experimentos anteriores, os resultados dos testes estatísticos mostravam que não havia diferença estatística significativa entre os métodos comparados. Para os primeiros três experimentos, foram analisados os resultados do CLUS com threshold igual a 0.5, o que pode, de certa forma ter influenciado também os resultados finais apresentados.

Vale ressaltar que em uma análise inicial (bruta e superficial) de todas as medidas de avaliação multirrótulo calculadas pelas bibliotecas do R e Python para o HPML.D, foi verificado para as medidas macro e micro média o mesmo comportamento dos experimentos anteriores. Portanto, analisar os resultados probabilísticos, sem um threshold definido seria interessante. Esta foi uma decisão tomada em conjunto com o time da Bélgica onde este projeto foi desenvolvido. Também é necessário ressaltar que, para os três primeiros experimentos não foram calculadas as curvas ROC e de precisão-revocação, portanto, estes resultados não estão disponíveis e constam como trabalhos futuros.

6.4.1 Análise das Partições

As melhores partições escolhidas de acordo com o critério do coeficiente de silhueta, para cada dataset e cada fold, são apresentadas na Tabela 61. Importante ressaltar que exatamente as mesmas partições foram usadas em todos as versões do método HPML.D. Os nomes dos conjuntos de dados foram simplificados por questões de espaço: entenda-se *c.*16*k* como *corel*16*k*00. No dataset *bibtex* (159 rótulos) uma partição composta por 126 grupos foi escolhida em mais folds do que as outras partições, enquanto que uma partição composta por 124 grupos foi escolhida em um fold, e as outras foram escolhidas em 2 folds. Nota-se também que as partições são próximas umas das outras.

Tabela 61 – Partições Escolhidas

fold	bibtex	cal500	c.16k1	c.16k3	c.16k4	c.16k5	c.16k6	c.16k7	c.16k8	c.16k9	c.16k10	r1s1	r1s2	s_ch
1	128	2	134	4	143	141	140	149	144	147	131	100	100	3
2	124	2	135	4	140	139	139	149	148	150	131	2	100	2
3	131	2	136	4	139	144	138	149	149	150	131	100	100	2
4	126	2	135	141	141	142	138	152	144	150	131	100	100	172
5	126	2	135	141	141	144	140	151	145	148	131	100	100	174
6	131	2	133	4	140	143	137	149	145	147	131	100	100	172
7	128	173	136	145	140	140	138	146	145	149	131	100	100	174
8	125	2	136	141	143	141	137	151	144	148	131	100	100	174
9	124	2	134	140	143	142	141	151	144	146	131	100	100	174
10	126	2	132	141	143	144	138	149	144	147	131	2	100	2

Já no dataset *cal*500 (174 rótulos), a partição mais escolhida foi uma composta por 2 grupos, sendo portanto, muito próxima de uma partição global (um único grupo). Uma partição composta por 135 grupos e outra por 136 grupos foram escolhidas, ambas, em 3 dos 10 folds no dataset *corel*16*k*001 (153 rótulos), enquanto que uma partição com 132 grupos e outra com 133 grupos foram escolhidas uma única vez cada. Novamente nota-se aqui um comportamento de escolha de partições próximas umas às outras.

No dataset *corel*16*k*004 duas partições bem distintas foram escolhidas como as melhores em 4 dos 10 folds, uma composta por 4 grupos e outra composta por 141 grupos, enquanto que outras foram escolhidas em um único fold cada. Esse dataset é o mais diverso com relação aos outros conjuntos de dados corel16k00, os quais escolheram sempre partições com um número de grupos próximo.

Uma partição composta por 144 grupos foi selecionada mais vezes do que todas as outros - em 3 dos 10 folds - no dataset *corel*16*k*005 (160 rótulos), enquanto que no dataset *corel*16*k*006 (162 rótulos), uma partição com 138 grupos foi escolhida mais vezes (4 folds). Já no dataset *corel*16*k*007 (174 rótulos), uma partição com 149 grupos foi escolhida com mais frequência, em 5 dos 10 folds. Situação similar ocorre no dataset *corel*16*k*008 (168 rótulos), mas neste caso é uma partição composta por 144 grupos.

Duas diferentes partições foram escolhidas com maior frequência no dataset *corel*16*k*009 (173 rótulos), uma composta por 147 grupos e outra por 150, ambas em 3 folds cada. Diferentemente de todos os outros conjuntos de dados corel16k00, no *corel*16*k*010

(144 rótulos) a mesma partição foi escohida em todos os folds, uma composta por 131 rótulos.

No dataset $rcv1sub1$ (101 rótulos) uma partição com 100 rótulos foi escolhida em 8 dos 10 folds, enquanto que no $rcv1sub2$ (101 rótulos) a mesma partição foi escolhida em todos os folds, também uma composta por 100 grupos. No caso desses dois conjuntos de dados, a partição 100, que contém 100 grupos, é muito próxima da configuração de uma partição local (um rótulo por grupo). Neste caso, 99 grupos contêm um único rótulo enquanto que um único grupo contém 2 rótulos.

Finalizando as partições escolhidas com maior frequência, uma partição composta por 174 grupos foi escolhida mais frequentemente no dataset *stackex_chemistry* (175 rótulos), em 4 dos 10 folds. Também neste dataset, uma partição mais próxima da local é escolhida como a melhor.

É possível notar consistência na escolha das partições na maioria dos conjuntos de dados, indicando que existe uma faixa de partições compostas por número de grupos similares que são consideradas as com maior qualidade de acordo com o coeficiente da silhueta.

6.4.2 Desempenho

As Tabelas 62 e 63 mostram os resultados de desempenho para as medidas de avaliação multirrótulo onde a cor azul indica os melhores resultados e a cor vermelha os piores. É possível notar que os métodos propostos foram capazes de obter resultados competitivos. Na média dos 14 conjuntos de dados, o método HPML.D$_{CE}$ obteve os melhores resultados nas medidas AUPRC-Macro, Roc-Auc-Macro e Roc-Auc-Micro, ao mesmo tempo que obteve os piores resultados nas medidas CLP e MLP. Isto indica que o método conseguiu classificar mais corretamente as classes positivas e negativas, mas ao mesmo tempo o método predisse um mesmo rótulo para todas as instâncias com maior frequência do que os outros, e a proporção de rótulos que nunca são preditos também é maior.

O método G$_{RF}$ obteve os melhores resultados na medida AUPRC-Micro, enquanto o Lo$_{RF}$ obteve o melhor desempenho na medida WLP (menor porcentagem de predições que estão sempre erradas), ambos os métodos não obtiveram o pior desempenho entre todos. O método HPML.D$_{CEI}$ obteve os piores resultados preditivios nas medidas AUPRC-Macro e AUPRC-Micro (robusto ao desbalanceamento de classes) , enquanto o método HPML.D obteve os piores resultados nas medidas Roc-Auc-Macro e Roc-Auc-Micro (menos robusto ao desbalanceamento de classes).

Já o método HPML.D$_{CI}$ obteve os melhores resultados preditivos nas medidas CLP (predisse menos vezes o mesmo rótulo para todas as instâncias) e MLP (menor proporção de rótulos que nunca são preditos), enquanto na medida WLP (um rótulo pode ser predito para algumas instâncias, mas essas predições estão sempre erradas), o método ECC obteve os piores resultados. Neste contexto, o método HPML.D$_{CI}$ não obteve o pior desempenho,

Tabela 62 – Desempenho CLP, MLP e WLP

	CLP							MLP						
Datasets	**Grf**	**Lorf**	**ECC**	**HPML.D**	**HPML.D.CI**	**HPML.D.CE**	**HPML.D.CEI**	**Grf**	**Lorf**	**ECC**	**HPML.D**	**HPML.D.CI**	**HPML.D.CE**	**HPML.D.CEI**
bibtex	0,39190	0,39052	**0,39681**	**0,38994**	0,39153	0,39671	0,39423	0,39190	0,39052	**0,39681**	**0,38994**	0,39153	0,39671	0,39423
cal500	0,39003	0,38867	0,39017	0,34503	**0,34158**	**0,39166**	0,38623	0,39003	0,38867	0,39017	0,34503	**0,34158**	**0,39166**	0,38623
corel16k001	**0,17920**	0,17422	0,17776	0,17399	0,17320	0,17877	**0,17191**	**0,17920**	0,17422	0,17776	0,17399	0,17320	0,17877	**0,17191**
corel16k003	**0,17748**	0,17162	0,17621	**0,17144**	0,17309	0,17439	0,17356	**0,17748**	0,17162	0,17621	**0,17144**	0,17309	0,17439	0,17356
corel16k004	**0,17793**	0,17315	0,17665	0,17153	**0,17101**	0,17702	0,17119	**0,17793**	0,17315	0,17665	0,17153	**0,17101**	0,17702	0,17119
corel16k005	0,17369	0,16993	0,17460	0,16878	**0,16769**	**0,17463**	0,16887	0,17369	0,16993	0,17460	0,16878	**0,16769**	**0,17463**	0,16887
corel16k006	0,16930	0,16557	**0,16952**	0,16234	0,16242	0,16824	**0,16234**	0,16930	0,16557	**0,16952**	0,16234	0,16242	0,16824	**0,16234**
corel16k007	**0,17390**	0,17048	0,17179	0,16895	0,16879	0,17386	**0,16579**	**0,17390**	0,17048	0,17179	0,16895	0,16879	0,17386	**0,16579**
corel16k008	0,17157	0,16788	0,17068	0,16639	0,16603	**0,17197**	**0,16564**	0,17157	0,16788	0,17068	0,16639	0,16603	**0,17197**	**0,16564**
corel16k009	**0,16588**	0,16200	0,16472	0,16014	0,15989	0,16538	**0,15814**	**0,16588**	0,16200	0,16472	0,16014	0,15989	0,16538	**0,15814**
corel16k010	0,18666	0,18162	0,18628	**0,17996**	0,18010	**0,18879**	0,18122	0,18666	0,18162	0,18628	**0,17996**	0,18010	**0,18879**	0,18122
rcv1sub1	0,35078	**0,34732**	**0,35486**	0,34820	0,34956	0,35481	0,35258	0,35078	**0,34732**	**0,35486**	0,34820	0,34956	0,35481	0,35258
rcv1sub2	0,39943	0,39632	0,40378	0,39632	0,39632	**0,40700**	**0,39481**	0,39943	0,39632	0,40378	0,39632	0,39632	**0,40700**	**0,39481**
stackex_chemistry	0,25802	0,25756	**0,26175**	0,26028	0,26055	0,25816	**0,25667**	0,25802	0,25756	**0,26175**	0,26028	0,26055	0,25816	**0,25667**
Média	**0,24041**	**0,23692**	**0,24111**	**0,23309**	**0,23298**	**0,24153**	**0,23594**	**0,24041**	**0,23692**	**0,24111**	**0,23309**	**0,23298**	**0,24153**	**0,23594**
Desvio Padrão	0,09681	0,09780	0,09877	**0,09393**	0,09403	0,09906	**0,09920**	0,09681	0,09780	0,09877	**0,09393**	0,09403	0,09906	**0,09920**

	WLP						
Datasets	**Grf**	**Lorf**	**ECC**	**HPML.D**	**HPML.D.CI**	**HPML.D.CE**	**HPML.D.CEI**
bibtex	**0,54528**	0,52893	0,52642	0,53270	0,53333	0,53333	**0,51635**
cal500	**0,85747**	**0,78851**	0,82069	0,83161	0,82356	0,83563	0,83851
corel16k001	0,80131	**0,80000**	**0,82288**	0,80131	0,80719	0,81765	0,81307
corel16k003	**0,79805**	0,80909	**0,82597**	0,80974	0,81104	0,82468	0,81104
corel16k004	**0,81049**	0,81296	**0,83704**	0,81852	0,81852	0,83519	0,82716
corel16k005	0,82063	**0,80813**	**0,83000**	0,81063	0,81188	**0,83000**	0,81500
corel16k006	0,80926	**0,80432**	**0,82901**	0,80988	0,80988	0,82222	0,81049
corel16k007	0,82644	**0,81494**	**0,83736**	0,81782	0,82126	0,83103	0,82414
corel16k008	0,80298	**0,80179**	0,80833	0,80298	0,81071	**0,81310**	0,80238
corel16k009	0,83873	**0,83179**	**0,85607**	0,83410	0,84220	0,85434	0,84682
corel16k010	0,80208	0,78819	0,81944	**0,78750**	0,79167	**0,82153**	0,81181
rcv1sub1	**0,72277**	**0,70198**	0,71089	0,70396	0,70495	0,71089	0,70297
rcv1sub2	0,71386	0,70891	**0,71881**	0,70891	0,70891	0,71485	**0,70495**
stackex_chemistry	0,80514	0,80743	0,81943	**0,79657**	0,80914	0,80800	**0,82171**
Média	**0,78246**	**0,77193**	**0,79017**	**0,77616**	**0,77887**	**0,78946**	**0,78189**
Desvio Padrão	**0,07851**	0,07954	0,08688	0,08053	0,08134	0,08512	**0,08795**

Tabela 63 – Desempenho Auprc-Macro, Auprc-Micro, Roc-Auc-Macro, Roc-Auc-Micro

	AUPRC-Macro							AUPRC-Micro						
Datasets	**Grf**	**Lorf**	**ECC**	**HPML.D**	**HPML.D.CI**	**HPML.D.CE**	**HPML.D.CEI**	**Grf**	**Lorf**	**ECC**	**HPML.D**	**HPML.D.CI**	**HPML.D.CE**	**HPML.D.CEI**
bibtex	**0,41903**	***0,39051***	0,41504	0,39389	0,39575	0,41745	0,39812	**0,49869**	**0,47108**	0,48666	0,47374	0,47486	0,48910	0,47451
cal500	0,24488	***0,23687***	0,23988	0,24631	0,23858	**0,25296**	0,23743	**0,47849**	0,45845	0,46348	0,47688	**0,45564**	0,47752	0,45787
corel16k001	0,10991	0,10516	0,11576	0,10509	0,10692	**0,11578**	***0,10366***	0,21223	0,20263	0,21238	0,20237	0,20259	**0,21256**	**0,19658**
corel16k003	0,11598	0,11041	**0,12142**	***0,10992***	0,11384	0,12135	0,11413	**0,21792**	0,20852	0,21670	0,20926	0,21095	0,21703	**0,20540**
corel16k004	0,12743	0,12431	0,13460	0,12382	0,12525	**0,13471**	***0,12103***	**0,21490**	0,20505	0,21306	0,20502	0,20402	0,21218	**0,19551**
corel16k005	0,10296	0,10162	**0,11218**	0,10226	0,10308	0,11080	***0,09826***	0,20995	0,20203	**0,21037**	0,20195	0,20172	0,20893	**0,19305**
corel16k006	0,12701	0,12323	**0,13380**	0,12323	0,12490	0,13339	***0,11969***	**0,21432**	0,20468	0,21175	0,20435	0,20440	0,21152	**0,19431**
corel16k007	0,11703	0,11079	0,12029	0,10984	0,11171	**0,12145**	***0,10887***	**0,21001**	0,20199	0,20862	0,20173	0,20107	0,20991	**0,19321**
corel16k008	0,11583	0,10907	**0,11873**	0,10895	0,11020	0,11868	***0,10727***	0,21279	0,20405	**0,21378**	0,20380	0,20414	0,21331	**0,19713**
corel16k009	0,10068	0,09503	**0,10486**	0,09460	0,09665	0,10469	***0,09200***	**0,20284**	0,19245	0,20167	0,19200	0,19238	0,20042	**0,18360**
corel16k010	0,12463	0,12120	**0,13286**	0,12236	0,12291	0,13048	***0,11645***	**0,22346**	0,21501	0,22059	0,21500	0,21497	0,22070	**0,20345**
rcv1sub1	0,39084	***0,36302***	0,38787	0,36921	0,36716	**0,39465**	0,36520	0,48095	0,47924	0,48593	0,48103	0,48034	**0,48715**	**0,47818**
rcv1sub2	**0,36803**	0,32516	0,35570	0,32482	0,32741	0,35592	***0,31946***	0,46604	**0,44762**	0,46439	0,44764	0,44767	**0,46751**	0,44954
stackex_chemistry	0,17529	***0,15413***	0,16677	0,16618	0,15971	**0,17675**	0,16157	**0,27039**	**0,24926**	0,26048	0,25744	0,25332	0,26712	0,25097
Média	0,18854	0,17646	0,18998	0,17860	0,17886	**0,19208**	***0,17594***	**0,29379**	0,28158	0,29070	0,28373	0,28200	0,29250	**0,27666**
Desvio Padrão	**0,11705**	***0,10598***	0,11206	0,10764	0,10690	0,11388	0,10771	0,12407	**0,12066**	0,12193	0,12326	0,12089	0,12430	**0,12475**

	Roc-Auc-Macro							Roc-Auc-Micro						
Datasets	**Grf**	**Lorf**	**ECC**	**HPML.D**	**HPML.D.CI**	**HPML.D.CE**	**HPML.D.CEI**	**Grf**	**Lorf**	**ECC**	**HPML.D**	**HPML.D.CI**	**HPML.D.CE**	**HPML.D.CEI**
bibtex	0,92238	0,90787	0,92353	***0,58927***	0,60321	**0,92452**	0,90945	**0,94282**	0,93006	0,94175	**0,54699**	0,54800	0,94229	0,93065
cal500	0,63994	0,63479	0,63649	0,63127	***0,62872***	**0,64803**	0,63221	**0,82110**	0,80534	0,80680	**0,58881**	0,58980	0,82008	0,80314
corel16k001	0,72043	0,71057	**0,73839**	***0,53440***	0,53733	0,73835	0,70534	0,85673	0,84801	**0,86308**	**0,48926**	0,48963	0,86288	0,84528
corel16k003	0,72294	0,71547	0,74318	***0,54906***	0,56145	**0,74597**	0,72329	0,86045	0,85337	0,86890	**0,53773**	0,53918	**0,86984**	0,85725
corel16k004	0,73497	0,73024	0,75845	***0,53333***	0,53572	**0,75862**	0,72495	0,86318	0,85595	**0,87210**	**0,50168**	0,50272	0,87175	0,85266
corel16k005	0,70753	0,70462	0,73360	***0,54140***	0,54311	**0,73414**	0,70072	0,85462	0,84867	**0,86407**	**0,55478**	0,55497	0,86395	0,84549
corel16k006	0,73241	0,72689	0,75400	***0,52975***	0,53111	**0,75413**	0,72257	0,86013	0,85410	**0,86924**	0,50837	**0,50835**	0,86906	0,85103
corel16k007	0,72480	0,71832	0,74505	***0,52448***	0,52677	**0,74610**	0,71323	0,86059	0,85444	0,86915	0,49449	**0,49441**	**0,86931**	0,85035
corel16k008	0,72543	0,71614	**0,74572**	***0,52697***	0,53004	0,74541	0,71133	0,86004	0,85230	**0,86828**	0,48338	**0,48297**	0,86786	0,84934
corel16k009	0,71989	0,71067	0,73923	***0,52826***	0,52950	**0,73945**	0,70384	0,85743	0,84910	**0,86465**	0,49925	**0,49888**	0,86396	0,84434
corel16k010	0,73236	0,72366	0,75143	***0,55071***	0,55275	**0,75183**	0,72011	0,86243	0,85418	**0,86916**	0,55378	**0,55282**	0,86908	0,85128
rcv1sub1	**0,91328**	0,87806	0,91035	***0,82621***	0,83086	0,91206	0,88529	**0,95661**	0,94855	0,95476	**0,85460**	0,85484	0,95560	0,94938
rcv1sub2	**0,91209**	0,87726	0,90779	0,87765	0,87845	0,90740	***0,87571***	**0,95433**	0,94412	0,95037	0,94414	0,94424	0,95033	**0,94332**
stackex_chemistry	0,77580	0,74630	0,77308	***0,59509***	0,61285	**0,78155**	0,75603	**0,89851**	0,87489	0,88933	**0,59611**	0,59855	0,89561	0,87969
Média	0,76316	0,75006	0,77573	***0,59556***	0,60013	**0,77768**	0,74886	0,87921	0,86951	0,88226	**0,58238**	0,58281	**0,88369**	0,86809
Desvio Padrão	0,08744	***0,07890***	0,08113	**0,11349**	0,11328	0,07981	0,08124	0,04205	0,04149	0,04036	0,14009	**0,14011**	**0,03889**	0,04272

e o ECC também não obteve o melhor desempenho entre todos os métodos comparados. Ainda dentro da média dos 14 datasets, pode-se concluir que o método HPML.D$_{CE}$ obteve os melhores resultados preditivos contabilizando as 7 medidas de avaliação, ficando em primeiro lugar em 5 delas, enquanto os métodos HPML.D$_{CEI}$, HPML.D$_{CI}$ e HPML.D ficam em último lugar com os piores resultados (2 medidas cada).

Contado a quantidade de datasets por método, e por melhor e pior melhor desempenho (contagem dos azuis e vermelhos por coluna), nota-se que na AUPRC-Macro os métodos ECC e HPML.D$_{CEI}$ tem o maior número de datasets com melhor desempenho, seis cada, enquanto HPML.D$_{CEI}$ tem o maior número de datasets com o pior desempenho, 9 dos 14. O método G$_{RF}$ é o com o maior número de datasets com melhor desempenho na AUPRC-Micro, 9 de 10, enquanto o método HPML.D$_{CEI}$ obteve o pior desempenho em 10 dos 14 datasets neta medida. Considerando a Roc-Auc-Micro, HPML.D$_{CEI}$ obteve melhor desempenho em 10 dos 14 datasets, enquanto HPML.D o pior em 13 dos 14 datasets. Já na Roc-Auc-Micro, o método ECC obteve os melhores desempenhos em 7 dos 14, enquanto o HPML.D o pior desempenho em 8 dos 14 datasets.

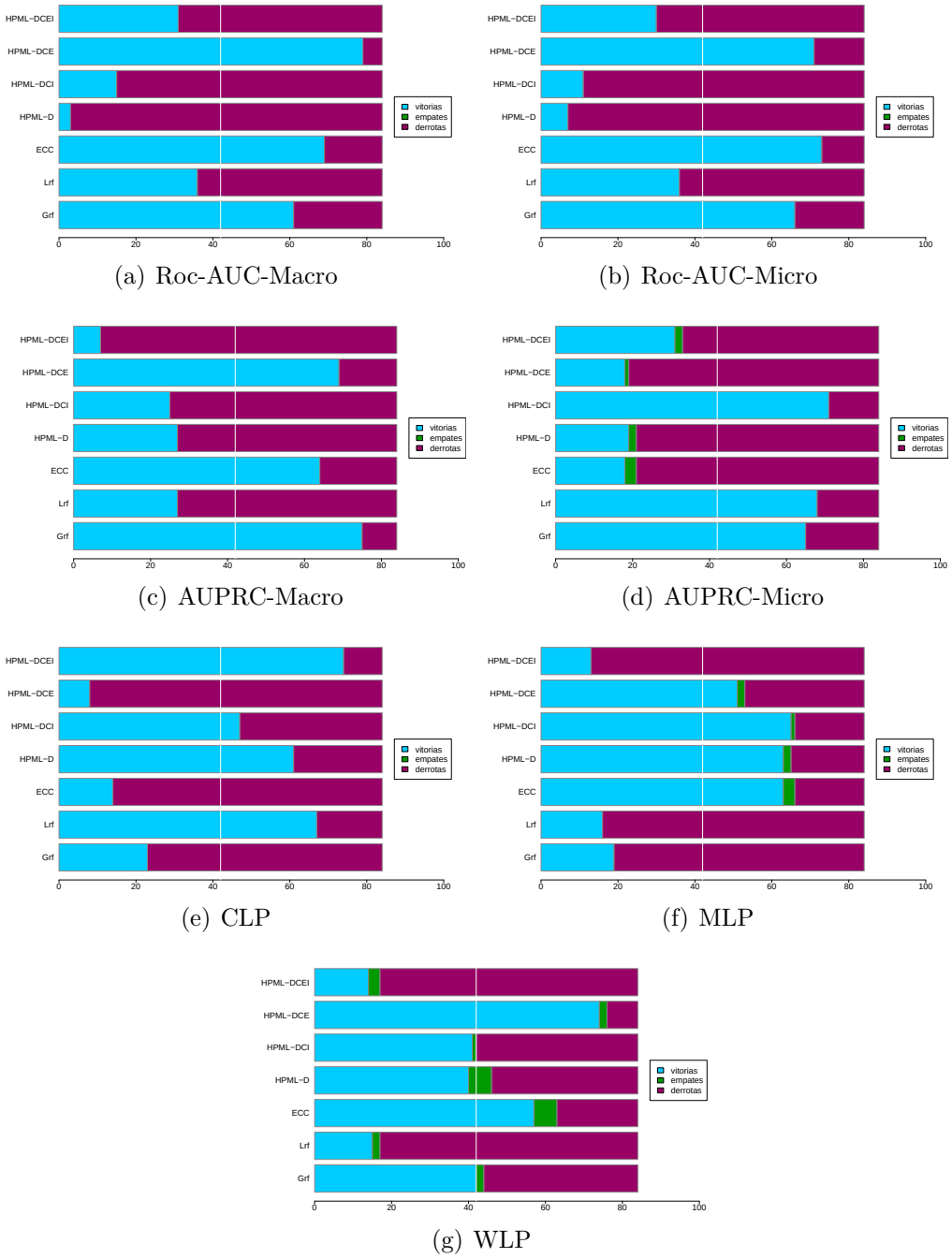
Na medida CLP e MLP, o método HPML.D$_{CEI}$ tem o maior número de datasets com o melhor desempenho, 5 de 14, enquanto os métodos G$_{RF}$ e HPML.D$_{CE}$ obtiveram pior desempenho em 7 datastets cada. Finalizando, na medida WLP, o método Lo$_{RF}$ obteve melhor desempenho em 8 dos 14 datasets, enquanto o ECC obteve o pior desempenho também em 8 dos 14 datasets. O que se nota neste contexto é que há uma grande disputa entre os métodos, ressaltando que o método HPML.D$_{CE}$ foi o que obteve melhor desempenho em mais datasets em todas as medidas, e o HPML.D o método que obteve o pior desempenho em mais datasets em todas as medidas de avaliação, quando comparados com todos os outros métodos. Isso indica que encadear os grupos disjuntos de rótulos correlacionados pode aumentar a qualidade das predições realizadas pelo classificador.

Considerando todos os 14 datasets, e todas as 7 medidas, verifica-se que o dataset bibtex obteve os melhores resultados entre todos os datasets nas medidas AUPRC-Macro (G$_{RF}$ e HPML.D$_{CE}$), AUPRC-Micro (G$_{RF}$ e HPML.D$_{CE}$), Roc-Auc-Macro (G$_{RF}$ e HPML.D$_{CE}$) e WLP (ECC e HPML.D$_{CEI}$). Isto indica que os métodos foram capazes de aprender melhor as correlações entre rótulos para este dataset.

O dataset rcvsub1 também obeve os melhores resultados nas medidas Roc-Auc-Micro (G$_{RF}$ e HPML.D$_{CE}$), mais uma vez evidenciando a disputa entre os métodos.O dataset Corel16k009 nas medidas CLP (HPML.D$_{CI}$ e HPML.D$_{CEI}$) e MLP (HPML.D$_{CI}$ e HPML.D$_{CEI}$), mostrando que a porcentagem de erros de predições de rótulos nesse dataset em particular foi menor do que os outros para os métodos listados. Esses são os três datasets com os melhores resultados dentre todos os datasets em cada uma das medidas avaliadas.

Cinco são os datasets que obtiveram os piores resultados entre todos os datasets experimentados. Corel16k009 obteve os piores resultados nas medidas Auprc-Macro (HPML.D

e HPML.D$_{CEI}$), Auprc-Micro (HPML.D e HPML.D$_{CEI}$) e WLP (ECC), indicando que os métodos apontados não foram capazes de lidar bem com o desbalanceamento das classes neste dataset.



(a) Roc-AUC-Macro (b) Roc-AUC-Micro

(c) AUPRC-Macro (d) AUPRC-Micro

(e) CLP (f) MLP

(g) WLP

Figura 77 – Gráficos de Vitórias, Empates e Derrotas para todas as medidas.

O dataset Corel1k007 obteve os piores resultados na medida Roc-Auc-Macro (HPML.D e HPML.D$_{CI}$), enquanto o Corel16k008 na medida Roc-Auc-Micro (HPML.D e HPML.D$_{CI}$). O dataset Rcv1Sub1 obteve os piores resultados nas medidas CLP (ECC e HPML.D$_{CE}$) e MLP (ECC e HPML.D$_{CE}$), enquanto o Cal500 na medida WLP (G$_{RF}$),

indicando que os métodos tiveram maior dificuldade em predizer os rótulos nesses datasets.

A Figura 77 mostra os gráficos de vitórias-derrotas-empates, com o intuito de elucidar a competitividade entre os métodos comparados. Na medida AUPRC-Macro, o método G_{RF} tem o maior número de vitórias, enquanto o HPML.D_{CEI} o menor, e com o segundo maior número de vitórias está o método HPML.D_{CE}. Já na medida AUPRC-Micro, o método HPML.D_{CE} tem o maior número de vitórias, enquanto o o método HPML.D_{CEI} o menor, sendo o L_{RF} o segundo método com o maior número de vitórias.

Na medida Roc-Auc-Macro, o método como maior número de vitórias é o HPML.D_{CE}. e o método HPML.D o método com o menor número de vitórias, sendo o ECC o segundo método com o maior número de vitórias. No entanto, na medida Roc-Auc-Micro, o ECC é o método com o maior número de vitórias, enquanto o HPML.D tem o menor número de vitórias, e o HPML.D_{CE} fica em segundo lugar. Nessas quatro medidas de avaliação, o método HPML.D_{CE} aparece em primeiro ou segundo lugar, evidenciando mais uma vez que encadear as partições híbridas pode levar a um melhor aprendizado e melhores predições.

Tratando agora das medidas de erro de predições de rótulos, o método HPML.D_{CEI} tem o maior número de vitórias na medida CLP, enquanto o HPML.D_{CE} o menor, e o L_{RF} o segundo maior número de vitórias. Já na medida CLP, o método HPML.D_{CI} possui o maior número de vitórias, enquanto o método HPML.D_{CEI} possui o menor número, e o ECC fica sem segundo lugar. Na medida WLP, o método com o maior número de vitórias é o HPML.D_{CE}, enquanto o HPML.D_{CEI} tem o menor, e o ECC fica em segundo lugar. No caso desses métodos, o nível de erro de predições foi menor do que para os outros, indicando que aprender as partições híbridas nas diferentes formas de encadeamento é uma melhor opção do que as outras abordagens e do que o próprio HPML.

Portanto, as informações do win-tie-loss mostram que há grande competitividade entre os métodos comparados, havendo pouca diferença no número de vitórias entre o primeiro e segundo lugar. Isto está refletido nos testes estatísticos realizados, conforme mostram os resultados apresentados na Subseção 6.4.3.

Para finalizar a análise do desempenho, a Tabela 64 apresenta uma comparação entre pares de métodos, indicando em cada célula o número total de datasets em que o método da linha obteve melhor desempenho com relação ao método da coluna. Os valores zeros nas diagonais indicam onde o método cruza consigo mesmo.

A partir da Tabela 64 nota-se que em várias comparações um método foi melhor que o outro em todos os 14 datasets. Olhando apenas para os métodos HPML.D, na AUPRC-Macro e AUPRC-Micro, a versão HPML.D_{CE} obtiveram melhores resultados de desempenho, em todos os 14 datasets, quando comparados aos métodos L_{RF}, HPML.D, HPML.D_{CI} e HPML.D_{CEI}, e o método HPML.D_{CI} melhor nos 14 do que o L_{RF} na medida AUPRC-Macro, elucidando que o HPML com encadeamento interno conseguiu superar o

Tabela 64 – Comparação Pareada

AUPRC-MACRO

Método	Grf	Lrf	ECC	HPML.D	HPML.DCI	HPML.DCE	HPML.DCEI	Média
Grf	0	14	5	13	13	2	14	10,17
Lrf	0	0	0	8	0	0	9	2,83
ECC	9	14	0	13	14	6	14	11,67
HPML-D	1	6	1	0	3	0	12	3,83
HPML-DCI	1	14	0	11	0	0	11	6,17
HPML-DCE	12	14	8	14	14	0	14	**12,67**
HPML-DCEI	0	5	0	2	3	0	0	1,67

AUPRC-MICRO

Método	Grf	Lrf	ECC	HPML.D	HPML.DCI	HPML.DCE	HPML.DCEI	Média
Grf	0	14	10	13	14	10	14	**12,50**
Lrf	0	0	0	8	8	0	11	4,50
ECC	4	14	0	13	14	5	14	10,67
HPML-D	1	6	1	0	7	0	12	4,50
HPML-DCI	0	6	0	7	0	0	12	4,17
HPML-DCE	4	14	9	14	14	0	14	11,50
HPML-DCEI	0	3	0	2	2	0	0	1,17

ROC-AUC-MACRO

Método	Grf	Lrf	ECC	HPML.D	HPML.DCI	HPML.DCE	HPML.DCEI	Média
Grf	0	14	4	14	14	2	13	10,17
Lrf	0	0	0	13	13	0	10	6,00
ECC	10	14	0	14	14	3	14	11,50
HPML-D	0	1	0	0	1	0	1	0,50
HPML-DCI	0	1	0	13	0	0	1	2,50
HPML-DCE	12	14	11	14	14	0	14	**13,17**
HPML-DCEI	1	4	0	13	13	0	0	5,17

ROC-AUC-MICRO

Método	Grf	Lrf	ECC	HPML.D	HPML.DCI	HPML.DCE	HPML.DCEI	Média
Grf	0	14	5	14	14	5	14	11,00
Lrf	0	0	0	13	13	0	10	6,00
ECC	9	14	0	14	14	8	14	**12,17**
HPML-D	0	1	0	0	5	0	1	1,17
HPML-DCI	0	1	0	9	0	0	1	1,83
HPML-DCE	9	14	6	14	14	0	14	11,83
HPML-DCEI	0	4	0	13	13	0	0	5,00

CLP

Método	Grf	Lrf	ECC	HPML.D	HPML.DCI	HPML.DCE	HPML.DCEI	Média
Grf	0	0	7	1	1	8	2	3,17
Lrf	14	0	14	2	4	14	3	8,50
ECC	7	0	0	0	0	9	0	2,67
HPML-D	13	11	14	0	6	13	6	10,50
HPML-DCI	13	9	14	7	0	13	7	10,50
HPML-DCE	6	0	5	1	1	0	0	2,17
HPML-DCEI	12	11	14	7	7	14	0	**10,83**

MLP

Método	Grf	Lrf	ECC	HPML.D	HPML.DCI	HPML.DCE	HPML.DCEI	Média
Grf	0	0	7	1	1	8	2	3,17
Lrf	14	0	14	2	4	14	3	8,50
ECC	7	0	0	0	0	9	0	**2,67**
HPML-D	13	11	14	0	6	13	6	10,50
HPML-DCI	13	9	14	7	0	13	7	10,50
HPML-DCE	6	0	5	1	1	0	0	2,17
HPML-DCEI	12	11	14	7	7	14	0	**10,83**

WLP

Método	Grf	Lrf	ECC	HPML.D	HPML.DCI	HPML.DCE	HPML.DCEI	Média
Grf	0	3	11	3	7	11	7	7,00
Lrf	11	0	13	11	13	14	12	**12,33**
ECC	3	1	0	2	3	4	2	2,50
HPML-D	9	2	12	0	10	14	10	9,50
HPML-DCI	7	0	11	1	0	12	9	6,67
HPML-DCE	3	0	8	0	1	0	2	2,33
HPML-DCEI	7	2	12	4	4	12	0	6,83

método L_{RF}.

Nas medidas Roc-Auc-Macro e Roc-Auc-Micro, o método HPML.D_{CE} também obteve melhores resultados nos 14 datasets quando comparados com os métodos L_{RF}, HPML.D, HPML.D_{CI}, e HPML.D_{CEI}, mostrando que usar o encadeamento externo é uma opção melhor do que a abordagem local, e também um encadeamento melhor que o interno e o combinado.

O método HPML.D_{CEI}, no entanto, obteve os melhores resultados nos 14 datasets quando comparados com os métodos HPML.D_{CE} e ECC nas medidas CLP e MLP, e na medida WLP o método HPML.D foi melhor que o HPML.D_{CE}. Os métodos HPMLD. e HPML.D_{CI} também obtiveram melhores resultados nos 14 datasets que o ECC. Isso alerta para o fato de que esses métodos tem uma porcentagem de erro de predições de rótulos menor do que os outros métodos, ainda que esses outros métodos tenham obtido resultados melhores nas outras medidas.

Quanto aos métodos não HPML, o método G_{RF} foi melhor que o L_{RF} e o HPML.D_{CEI} nos 14 datasets na medida AUPRC-Macro, e também o ECC foi melhor nos 14 datasets quanto comparado com os métodos L_{RF} e HPML.D_{CI}. Na medida AUPC-Micro, tanto os métodos G_{RF} quanto o ECC foram melhores em 14 datasets com relação aos métodos HPML.D_{CI}, HPML.D_{CEI} e o L_{RF}.

Na medidas Roc-Auc-Micro e Roc-Auc-Macro, o método ECC obteve melhores resultados em 14 datasets quando comparado com os métodos L_{RF}, HPML.D, HPML.D_{CI} e HPML.D_{CEI}. Quanto às medidas CLP e MLP, o método L_{RF} obteve melhores resultados que os métodos G_{RF}, ECc e HPML.D_{CE} nos 14 datasets, enquanto na WLP, L_{RF} foi melhor que o HPML.D_{CE}. Isto elucida a competitividade entre os métodos, mostrando também que é preciso melhorias nas versões do metódo HPML para melhorar o aprendizado das correlações.

A partir da análise feita com ajuda das Tabela 62, Tabela 62, 64, e Figura 77, fica evidente que o melhor método vai depender de cada caso, já que para as medidas analisadas, a diferença entre eles é pequena. Também é possível concluir aplicar algum tipo de encadeamento no HPML é melhor do que aprender apenas os grupos. Na próxima seção são apresentados os resultados estatísticos.

6.4.3 Testes Estatísticos

Os gráficos de distância crítica são apresentados na Figura 78 onde linhas conectadas indicam que não há diferença estatística entre os métodos. O teste de Friedman foi conduzido com $\alpha = 0,5$ e a distância crítica calculada é de $2,46198$ para todas as medidas. A Tabela 65 apresenta os p-Values para cada medida, assim como os valores de Chi-Square e Hipóteses.

É possível notar pelos gráficos que não há diferença estatística entre os metódos G_{RF}, HPML.D_{CE} e ECC em todas as medidas. Também não há diferença estatística significa

entre os métodos G_{RF} e L_{RF} nas medidas AUPRC-Micro, Roc-Auc-Macro, Roc-Auc-Micro, MLP e WLP. Não há diferença estatística significativa entre os métodos L_{RF}, HPML.D_{CEI}, HPML.D_{CI} e HPML.D nas medidas AUPRC-Micro, Roc-Micro, CLP, MLP e WLP.

Os gráficos comprovam a competitividade apresentada na seção anterior. Ainda que em algumas medidas, diferentes versões do HPML.D estejam no primeiro ou segundo lugar do ranking, não há diferença estatística ou com o método G_{RF} ou com o ECC. Portanto, ainda que a análise de desempenho, com suporte do vitórias-derrotas-empates e comparação pareada, mostre que os resultados são competitivos, que o aprendizado das partições híbridas colaborou, que encadear os grupos ajudou no aprendizado, não é possível afirmar pelos resultados dos testes estatísticos que aprender as partições híbridas de forma encadeada é melhor que os métodos tradicionais.

Tabela 65 – Resultados do Teste de Friedman e Nemenyi

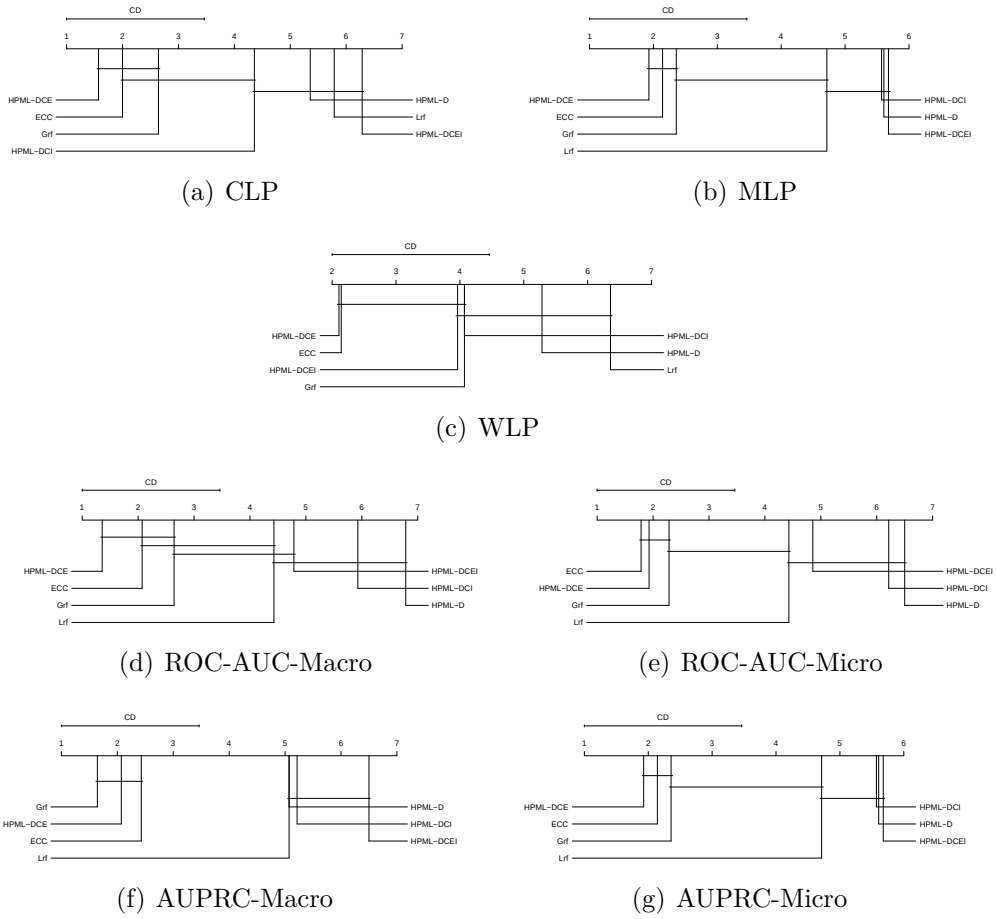
Medidas	ChiSquare	pValue	Distância Crítica	Hipótese
CLP	66,36734694	2,27E-12	2,461987492	Ha: Different
MLP	56,45663265	2,35E-10	2,461987492	Ha: Different
WLP	42,75765306	1,30E-07	2,461987492	Ha: Different
Auprc-Macro	65,29591837	3,75E-12	2,461987492	Ha: Different
Auprc-Micro	56,45663265	2,35E-10	2,461987492	Ha: Different
Roc-Auc-Macro	74,47959184	4,91E-14	2,461987492	Ha: Different
Roc-auc-Micro	72,61224490	1,19E-13	2,461987492	Ha: Different

Feitas as observações com relação ao desempenho preditivo do Experimento 4, as questões levantadas sobre os métodos são então respondidas:

HPML.D$_{PADRAO}$: Se conjuntos de dados grandes podem obter um melhor resultado no HPML.D que o ECC, então poderia se afirmar que aprender grupos disjuntos de rótulos correlacionados sem nenhum encadeamento é melhor que aprender todos os rótulos juntos em um conjunto de classificadores em cadeia. No entanto, o que se verifica é que isso é verdade para alguns conjuntos de dados e, ainda que no ranking o HPML.D_{PADRAO} ocupe uma posição mais baixa que o ECC em algumas medidas de avaliação, os gráficos de distância crítica mostram que há diferença estatística entre eles.

HPML.D$_{CI}$: Se o desempenho preditivo do classificador melhora quando é utilizado um Ensemble of Classifier Chains (ECC) em cada um dos grupos disjuntos de rótulos de uma partição, então, seria possível afirmar que usar o ECC com as partições híbridas é melhor que o tradicional ECC. Os resultados mostram que isso é verdade para alguns conjuntos de dados, e em todas as medidas de avaliação pode-se notar uma diferença estatística entre o ECC e o método HPML.D_{CI}. Ainda que o ECC esteja acima no ranking, para alguns casos é possível afirmar que a hipótese é verdadeira.

HPML.D$_{CE}$: Se conjuntos de dados grandes podem obter um melhor resultado a partir da quebra da longa cadeia de classificadores, por meio das partições híbridas, em uma ordem de encadeamento baseada correlações, então seria possível afirmar que encadear grupos disjuntos de rótulos correlacionados é melhor que usar o tradicional ECC. Há



(a) CLP (b) MLP

(c) WLP

(d) ROC-AUC-Macro (e) ROC-AUC-Micro

(f) AUPRC-Macro (g) AUPRC-Micro

Figura 78 – Gráficos de Distância Crítica de Nemenyi.

casos assim reportados nos resultados de desempenho e, apesar de em alguns conjuntos de dados esse método ser melhor que o ECC, não há diferença estatística entre eles nas medidas de avaliação analisadas, portanto, não se pode afirmar que a hipótese é verdadeira para todos os casos.

HPML.D$_{CEI}$: Se aprender vários níveis de correlação leva o classificador a obter melhores resultados, então poderia se dizer que encadear os rótulos dentro dos grupos e também encadear os grupos seria melhor que o tradicional ECC e também melhor que o LOCAL. Os experimentos mostraram que, para os conjuntos de dados selecionados, isso não é totalmente verdade. Ainda que este método tenha tido um desempenho melhor do que os outros métodos nas medidas de problema de rótulo, indicando menor porcentagem de erro nas predições, nas outras medidas o método teve desempenho pior que o ECC e o HPML.D$_{CE}$. Portanto, não se pode dizer que a hipótese é verdadeira para todas as situações.

6.5 Custo Computacional

As três primeiras versões do HPML utilizaram o CLUS-Framework o qual tem um custo computacional muito alto. Isto ocorre pois é necessário criar todos os arquivos físicos em formato ARFF para cada um dos grupos de rótulos disjuntos que compõem cada uma das partições geradas. Este fato consome muita memória e processamento e, portanto, alguns conjuntos de dados não chegaram a terminar a execução, enquanto outros levaram mais de 15 dias. Para os conjuntos de dados menores que 10 rótulos a execução de fato era mais rápida já que menos partições são geradas e, consequentemente, menos arquivos precisam ser construídos.

Também por conta disto, as partições globais e locais executaram mais rapidamente que as partições aleatórias e híbridas, já que para as partições globais apenas um grupo é construídos, e para as locais é feita a separação de todos os rótulos em grupos individuais o que é mais simples do que selecionar rótulos específicos que compõem cada um dos grupos das partições aleatórias e híbridas.

Dado este contexto, optou-se por não mais validar as partições geradas com o CLUS no experimento de comunidades e encadeamento. O coeficiente da silhueta é calculado rapidamente, permitindo obter resultados para análise em menos tempo, e portanto, tornou-se uma alternativa melhor para esses experimentos onde conjuntos de dados com muitos rótulos foram usados. Ainda no caso do encadeamento, havia o agravante do tempo, pois o projeto deveria ser executado em 6 meses e usar o CLUS nesta situação atrasaria por demais a entrega.

O tempo de execução usando as Florestas Aleatórias no experimento de encadeamento foi deveras mais rápido, ainda que usando conjuntos de dados com mais de 100 rótulos. Todos os experimentos foram executados com sucesso no prazo máximo de uma semana. O HPML.D.CEI foi o que mais demorou para ser executado entre todos os particionamentos comparados já que era necessário fazer tanto o encadeamento interno quanto o externo. Já as partições globais foram as mais rápidas.

6.6 Estratégia recomendada

Se for necessária a obtenção de resultados rápidos, o ideal é utilizar as Florestas Aleatórias Multirrótulo no lugar do CLUS e também o coeficiente da silhueta para escolha da melhor partição híbrida, ao invés da validação com o classificador. Além disso, as partições usando os métodos de detecção de comunidades hierárquicos se mostraram melhores do que as partições geradas usando o algoritmo de agrupamento hierárquico aglomerativo e também mapas auto-organizáveis de kohonen.

Vale ressaltar que todas as versões do HPML definidas nesta tese têm um certo custo computacional ao se escolher validar as partições usando um classificador. Se for neces-

sário realizar tal tarefa, então é melhor escolher as Florestas Aleatórias Multirrótulo para a validação, ou ainda algum outro classificador que consiga lidar com os diferentes tipos de grupos de rótulos e que também tenha se provado rápido com relação ao tempo de execução de forma geral.

Neste caso, ainda pode-se utilizar a medida de avaliação Hamming-Loss como critério de seleção de melhor partição híbrida, ao invés da Macro-F1 ou Micro-F1, ou ainda outra medida de interesse conforme o domínio ou contexto do problema multirrótulo que estiver sendo resolvido.

CONCLUSÃO

Nesta tese foi proposta, desenvolvida, implementada e avaliada uma estratégia capaz de particionar o espaço de rótulos que considera as correlações entre rótulos, gera várias partições híbridas, e escolhe uma entre elas que é capaz de otimizar o desempenho dos classificadores. As partições híbridas são compostas por grupos disjuntos de rótulos correlacionados e a estratégia é aqui denominada HPML.

A hipótese desta tese foi a de que é possível encontrar múltiplas partições de dados compostas por grupos disjuntos de rótulos correlacionados, no domínio de problemas de classificação multirrótulo, e uma entre elas ser capaz de melhorar o desempenho preditivo do classificador com relação às tradicionais abordagens global e local. Entende-se por abordagem global os métodos que lidam com todos os rótulos de uma única vez e, como abordagem local, os métodos que realizam a divisão do problema multirrótulo em vários problemas binários. Do ponto de vista de particionamento do espaço de rótulos, a abordagem global pode ser considerada uma partição global, isto é, uma partição de dados composta por um único grupo com todos os rótulos. Já a abordagem local pode ser considerada uma partição local, isto é, uma partição composta por vários grupos e cada grupo é composto por um único rótulo.

Ainda, de acordo com a literatura, a correlação entre rótulos é natural em dados multirrótulo e facilita o aprendizado dos rótulos pelos classificadores. Dessa forma, enquanto na abordagem global os métodos podem não aprender correlações entre rótulos mais específicas que podem ser relevantes para o modelo, na abordagem local essas correlações são totalmente ignoradas. Com as partições híbridas, essa desvantagem no uso das partições (e informações) locais ou globais pode ser superada para alguns conjuntos de dados, conforme mostraram os resultados obtidos durante a condução dos experimentos.

A abordagem HPML foi definida como um fluxo de 7 passos: 1) pré-processamento dos dados; 2) modelagem das correlações; 3) particionamento do espaço de rótulos, 4) construção dos datasets correspondentes à cada partição encontrada; 5) validação das partições híbridas encontradas; 6) escolha da partição híbrida mais adequada; e 7) teste

da partição híbrida selecionada. A partir desta metodologia, quatro diferentes instanciações da estratégia foram elaboradas: 1) HPML.A: modela as correlações usando medidas de similaridade e particiona o espaço de rótulos com um algoritmo de agrupamento hierárquico-aglomerativo; 2) HPML.B: modela e particiona o espaço de rótulos com um SOM; 3) HPML.C: modela e particiona as correlações usando medidas de similaridade e métodos de detecção de comunidades; 4) HPML.D: encadeia o HPML.A em três diferentes formas sendo uma interna (encadeamento dos rótulos dentro de cada grupo), outra externa (encadeamento dos grupos da partição), e uma última combinando encadeamento interno e externo.

6.7 Resumo dos Resultados

Quatro experimentos foram conduzidos denominados HPML.A.c, Exaustivo-Oráculo, Comunidades e Encadeamento. Em HPML.A$_C$ o Índice Jaccard foi usado para modelar correlações, e o algoritmo de agrupamento hierárquico aglomerativo para particionar. Uma entre três medidas de ligação foi selecionada para construir o dendrograma que em seguida foi usada para encontrar as partições híbridas. O classificador CLUS foi usado para validação e teste das partições, e a medida de avaliação MACRO-F1 foi usada como métrica de seleção de melhor partição. Além disso, essa versão do HPML.A foi comparada com a partição global, local e dois tipos de partições aleatórias baseadas no algoritmo de agrupamento hierárquico aglomerativo.

Uma busca exaustiva e um oráculo foram conduzidos no experimento Exaustivo-Oráculo, com o objetivo de averiguar o quão distante ou próxima uma partição híbrida está da melhor partição escolhida dentre todas as possíveis para um dataset. Datasets com mais de 7 rótulos no espaço de rótulos não foram considerados neste experimento pois o número total de partições possíveis[7] extrapola as capacidades de processamento disponíveis. Neste experimento foi implementada outra versão do HPML.A e a versão do HPML.B, os quais tiveram suas partições comparadas com a global, local e três tipos de partições aleatórias. Novamente o framework CLUS foi usado como classificador, as partições foram validadas com o classificador CLUS e o coeficiente da silhueta, e as partições com maior Macro-F1, Micro-F1 e Coeficiente de Silhueta foram escolhidas para o teste.

O HPML.C foi implementado no experimento Comunidades, em que sete diferentes métodos de detecção de comunidade foram usados para modelar e particionar o espaço de rótulos, assim como duas diferentes medidas de similaridade: Jaccard que considera as frequências com que rótulos ocorrem juntos ou separados; e Rogers-Tanimoto que além dessas também considera as frequências com que dois rótulos nunca ocorrem juntos. Neste experimento foram usados datasets com um número de rótulos maior no espaço de

[7] Sequencia do número de bell: 1 (0), 1 (1), 2 (2), 5 (3), 15 (4), 52 (5), 203 (6), 877 (7), 4140 (8), 21.147 (9), 115.975 (10), 678.570 (11), 4.213.597 (12), 27.644.437 (13)

rótulos, o coeficiente de silhueta para validação e o framework CLUS como classificador. As partições selecionadas foram comparadas com a global, local e também uma versão de partição aleatória baseada em grafo aleatório.

Por fim, o HPML.D foi testado no experimento Encadeamento e, ao invés do CLUS framework, Florestas Aleatórias foram usadas como classificador. Outra versão do HPML.A foi implementada, onde o Índice Jaccard modela as correlações, mas os dendrogramas do algoritmo de agrupamento hierárquico aglomerativo são construídos com a métrica de ligação ward.D2. Os métodos do HPML.D foram comparados entre eles mesmos e também com as partições global, local e o ECC. O Experimento HPML.A)C e Comunidades foram analisados usando as medidas de avaliação CLP, MLP, WLP, Macro-F1, Macro-Precision, Macro-Recall, Micro-F1, Micro-Precision e Micro-Recall. No Experimento de Encadeamento foram analisadas, além da CLP, MLP e WLP, as curvas roc e também de precisão e revocação e, por fim, no Experimento Exaustivo-Oráculo apenas a Macro-F1 e Micro-F1 foram analisadas.

Todos os experimentos conduzidos levam a um resultado comum entre eles: as abordagens global e local não se mostram melhores do que escolhas aleatórias ou direcionadas a grupos de rótulos. Portanto, ainda que diferentes tipos de modelagem de correlação, e particionamentos considerando as correlações, tenham sido utilizados, os classificadores induzidos não obtiveram grandes melhorias no desempenho.

Os resultados das medidas CLP, MLP e WLP elucidaram essa conclusão. A partir dessas medidas de problema de rótulo, que podem também ser entendidas como medidas de erro de predição, foi possível averiguar a proporção de predições erradas e constatar que muitos dos rótulos não estão sendo aprendidos pelos classificadores. Se os rótulos não estão sendo aprendidos pelo métodos tradicionais, então, ainda que se tente melhorar o poder preditivo destes através do aprendizado de correlações, o mesmo poderá não ocorrer de forma adequada.

Com relação ao desempenho, sem considerar as diferenças estatísticas encontradas pelos testes de Friedman/Nemenyi, um comportamento notado entre os experimentos HPML.A$_C$, Exaustivo-Oráculo e Comunidades, foi o de que as partições locais tendem a obter os melhores resultados e as partições globais o pior quando os datasets possuem poucos rótulos. Além disso, as partições híbridas tendem a ser competitivas com as partições aleatórias e também com as locais. De forma semelhante, as partições aleatórias tendem a ser competitivas com as locais e híbridas. Dentro deste contexto, utilizar as partições híbridas é superior a usar as partições globais ou locais. Ao que indica isto é uma tendência e seria necessário mais experimentação para a confirmação exata.

Ainda com relação ao desempenho, os experimentos com o encadeamento mostraram que encadear grupos disjuntos de rótulos correlacionados tem a capacidade para melhorar o desempenho preditivo do classificador, mas ainda é necessário mais investigação, haja visto que não houve diferença estatística significativa com a abordagem global e o ECC.

Com relação ao particionamento, nos experimentos Exaustivo-Oráculo e Comunidades, notou-se que grande parte das partições híbridas escolhidas pelos métodos são mais próximas de uma partição global do que uma partição local. Uma partição com dois grupos de rótulos é mais próxima de uma partição com um grupo do que uma partição com 40 grupos, por exemplo. De forma semelhante, uma partição com $l - 1$ grupos é mais próxima de uma partição local que é composta com l grupos do que uma partição composta por dois grupos. No entanto, o Exaustivo-Oráculo também mostrou que as partições híbridas escolhidas são próximas do oráculo.

Se as partições locais tendem a levar ao melhor desempenho, e as globais ao pior nesse caso, então a dedução a que se chega é de que seria melhor que a partição híbrida escolhida por todos os folds de um dataset fosse uma com uma configuração mais similar possível com a de uma partição local. No entanto, como mostrado pelos experimentos, muitos datasets tiveram melhora no seu desempenho ao escolher uma partição mais próxima da global ao invés da local. Além disso, as características dos datasets também influenciam no processo e resultado final. O mesmo poderia ser dito da situação inversa, onde as partições globais tendem a levar ao melhor desempenho, e muitos datasets tiveram melhora no seu desempenho ao escolher uma partição híbrida mais próxima da local ao invés da global.

Ainda assim, não é possível afirmar que uma partição híbrida mais próxima da local, ou da global, é a melhor para melhorar o poder preditivo pois os testes estatísticos mostraram que, para as várias medidas de avaliação, métodos e conjuntos de dados utilizados, não há diferença estatística entre as partições comparadas em muitos casos.

Ao realizar o encadeamento dos rótulos e grupos com os métodos do HPML.D, esperava-se que uma melhoria significativa fosse elucidada nos resultados e que os métodos encadeados fossem superar o ECC tradicional. Apesar dos resultados de desempenho terem mostrado alguma melhora em alguns datasets e medidas de avaliação, de forma geral os testes estatísticos mostraram que não há diferenças significativas entre os métodos.

Pode-se concluir, portanto, que é necessário experimentar o HPML.A, HPML.B e HPML.C com Florestas Aleatórias Multirrótulo para se obter mais informações a respeito do métodos e de como as correlações tem colaborado com a melhoria das predições, assim como também o HPML.D com o Clus-Framework. Dessa forma, uma comparação padronizada entre todos os métodos propostos poderia ser feita e algumas questões levantadas, respondidas, mas também não há garantias de que mesmo fazendo novas experimentações o cenário mude completamente ou respostas sejam encontradas.

Finalizando, em alguns casos é possível encontrar partições de dados compostas por grupos disjuntos de rótulos correlacionados, no domínio de problemas de classificação multirrótulo, e uma entre elas ser capaz de melhorar o desempenho preditivo do classificador com relação às tradicionais abordagens global e local. Mas é preciso melhorar o processo, de forma que os classificadores possam aprender adequadamente as correlações e diminuir a porcentagem de erros de predições, levando a uma melhora mais significativa

nos valores resultantes de desempenho.

6.8 Trabalhos Futuros

Para se chegar nessa melhoria, uma opção seria olhar para medidas de similaridades mais complexas e até mesmo desenvolver uma medida de correlação direcionada ao problema, capaz de capturar as correlações de uma forma diferente da utilizada nos métodos apresentados. Investigar na literatura a forma com que pesquisadores tem tentado modelar e extrair as correlações diretamente entre rótulos, com sucesso, poderia dar algum direcionamento nesse sentido.

Outro aspecto que seria interessante explorar (e formular) é referente à quanto de fato um classificador consegue melhorar as predições ao aprender as correlações, isto é, desenvolver uma medida que quantifique o aprendizado baseado nas correlações.

Os experimentos conduzidos nesta tese poderiam ser reconduzidos para datasets ainda maiores, e considerando também a possibilidade de comparar todas partições híbridas resultantes de todas as métricas de ligação do algoritmo de agrupamento hierárquico aglomerativo, e também de todas as partições dos métodos de detecção de comunidade.

Novos métodos de particionamento também poderiam ser testados, desde que os mesmos produzam partições sem a necessidade de informar um número específico de grupos. Além disso, como novos métodos de detecção de comunidades e algoritmos de agrupamento têm sido propostos na literatura, seria interessante verificar se estes se encaixam melhor na metodologia do HPML.

Quanto ao critério de seleção da melhor partição híbrida, outra medida diferente da Macro-F1 e Micro-F1 poderia ser utilizada, como a Hamming-Loss por exemplo. Mas um novo método para selecionar a partição híbrida mais adequada também poderia ser formulado. Por exemplo, um método que não seja baseado nem nas medidas de avaliação de desempenho, nem em medidas de qualidade de agrupamento, mas um que leve em conta o particionamento baseado nas correlações.

Um dos desafios encontrados durante a execução dos experimentos foi a disponibilidade de recursos computacionais. Este foi um fator que limitou validar todas as partições geradas usando classificadores na validação cruzada de 10-folds para datasets com um espaço de rótulos muito grande, já que é necessário criar os datasets para cada grupo, de cada partição e de cada fold, o que consome bastante memória e processamento. Portanto, para ser capaz de testar a hipótese em datasets ainda maiores, é necessário buscar formas para facilitar a execução em paralelo do HPML com a validação cruzada de 10-folds.

Usar instance hardness ou possibilistic cluster também seria uma opção para melhorar os resultados. Conduzir um experimento controlado também ajudaria a entender porque os métodos propostos não obtiveram diferenças estatísticas significativas. Gerar bases de dados sintéticas também seria uma opção para entender melhor esta e outras questões

como os diferentes tipos de correlações. Conduzir um ensemble de todos os métodos propostos também seria uma possibilidade.

Finalizando, também seria interessante utilizar um classificador multirrótulo diferente de árvores de decisão. Mas para ser compatível com o HPML, é necessário que o classificador tenha disponível versões local e global. Encontrar uma forma de testar todas as partições possíveis de um dataset com mais de 7 rótulos no espaço de rótulos também continua sendo um grande desafio.

Referências

ABDI, H.; WILLIAMS, L. J. Principal component analysis. **WIREs Comput. Stat.**, John Wiley & amp; Sons, Inc., USA, v. 2, n. 4, p. 433–459, jul. 2010. ISSN 1939-5108.

ABEYRATHNA, D. L. B. G. M. **Multi-Label Classification Using Higher-Order Label Clusters**. Dissertação (Mestrado) — Department of Computer Science and the Faculty of the Graduate College University of Nebraska, December 2018.

ALBERS, S. Online algorithms: a survey. p. 3–26, 2003.

ALER, R.; HANDL, J.; KNOWLES, J. D. Comparing multi-objective and threshold-moving roc curve generation for a prototype-based classifier. In: **Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation**. New York, NY, USA: Association for Computing Machinery, 2013. (GECCO '13), p. 1029–1036. ISBN 9781450319638. Disponível em: <https://doi-org.ez31.periodicos.capes.gov.br/10.1145/2463372.2463504>.

ALLAM, Z.; DHUNNY, Z. A. On big data, artificial intelligence and smart cities. **Cities**, v. 89, p. 80 – 91, 2019. ISSN 0264-2751. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0264275118315968>.

ALPAYDIN, E. **Introduction to Machine Learning**. [S.l.]: The MIT Press, 2014. ISBN 0262028182, 9780262028189.

ALTMAN, N. S. An introduction to kernel and nearest-neighbor nonparametric regression. **The American Statistician**, Taylor & Francis, v. 46, n. 3, p. 175–185, 1992.

BAREZI, E. J.; KWOK, J. T.; RABIEE, H. R. Multi-Label learning in the independent label sub-spaces. **Pattern Recognit. Lett.**, Elsevier B.V., v. 97, p. 8–12, 2017. ISSN 01678655.

BASGALUPP, M. et al. Beyond global and local multi-target learning. **Information Sciences**, v. 579, p. 508–524, 2021.

BASTANLAR, Y.; OZUYSAL, M. **Introduction to Machine Learning Second Edition**. [s.n.], 2014. v. 1107. 105–28 p. ISSN 1940-6029. ISBN 9780262012430. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/24272434>.

BHATIA, K. et al. Sparse local embeddings for extreme multi-label classification. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2015. p. 730–738.

BLOCKEEL, H.; RAEDT, L. D.; RAMON, J. Top-down induction of clustering trees. In: **Proceedings of the Fifteenth International Conference on Machine Learning**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998. (ICML '98), p. 55–63. ISBN 1558605568.

BLONDEL, V. D. et al. Fast unfolding of communities in large networks. **Journal of Statistical Mechanics: Theory and Experiment**, IOP Publishing, v. 2008, 2008.

BOGATINOVSKI, J. et al. Comprehensive comparative study of multi-label classification methods. **Expert Systems with Applications**, Elsevier Ltd, v. 203, 2022. ISSN 09574174.

BOUTELL, M. R. et al. Learning multi-label scene classification. **Pattern Recognition**, v. 37, n. 9, p. 1757 – 1771, 2004. ISSN 0031-3203.

BRADLEY, A. P. The use of the area under the roc curve in the evaluation of machine learning algorithms. **Pattern Recognition**, v. 30, n. 7, p. 1145 – 1159, 1997. ISSN 0031-3203. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0031320396001422>.

BREIMAN, L. Bagging predictors. **Machine Learning**, v. 24, n. 2, p. 123–140, ago. 1996. ISSN 1573-0565. Disponível em: <https://doi.org/10.1007/BF00058655>.

CAN, F.; OZKARAHAN, E. A. Concepts and effectiveness of the cover-coefficient-based clustering methodology for text databases. **ACM Trans. Database Syst.**, Association for Computing Machinery, New York, NY, USA, v. 15, n. 4, p. 483–517, dec 1990. ISSN 0362-5915. Disponível em: <https://doi.org/10.1145/99935.99938>.

CARVALHO, A. C. P. L. F. de; FREITAS, A. A. A tutorial on multi-label classification techniques. In: **Studies in Computational Intelligence**. [S.l.]: Springer Berlin Heidelberg, 2009. p. 177–195.

CERRI, R. **Redes Neurais e algoritmos genéticos para problemas de classificação hierárquica multirrótulo**. Tese (Tese de Doutorado) — Instituto de Ciências Matemáticas e Computacionais da Universidade de São Paulo., São Carlos/SP, jan. 2014.

CERVANTES, J. et al. A comprehensive survey on support vector machine classification: Applications, challenges and trends. **Neurocomputing**, v. 408, p. 189 – 215, 2020. ISSN 0925-2312.

CHAN, A.; FREITAS, A. A. A new ant colony algorithm for multi-label classification with applications in bioinfomatics. In: **Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation**. New York, NY, USA: Association for Computing Machinery, 2006. (GECCO '06), p. 27–34. ISBN 1595931864.

CHARTE et al. Tips, guidelines and tools for managing multi-label datasets: The mldr.datasets r package and the cometa data repository. **Neurocomputing**, 2018. ISSN 0925-2312.

CHARTE, F. et al. On the impact of dataset complexity and sampling strategy in multilabel classifiers performance. In: **Hybrid Artificial Intelligent Systems**. [S.l.]: Springer International Publishing, 2016. p. 500–511. ISBN 978-3-319-32034-2.

CHEN, L.; WANG, Y.; LI, H. Enhancement of DNN-based multilabel classification by grouping labels based on data imbalance and label correlation. **Pattern Recognit.**, Elsevier Ltd, v. 132, p. 108964, 2022. ISSN 00313203. Disponível em: <https://doi.org/10.1016/j.patcog.2022.108964>.

CHU, Y. et al. Predicting drug-target interactions using multi-label learning with community detection method (dti-mlcd). **bioRxiv**, Cold Spring Harbor Laboratory, 2020. Disponível em: <https://www.biorxiv.org/content/early/2020/05/12/2020.05.11.087734>.

CLARE, A.; KING, R. D. Knowledge discovery in multi-label phenotype data. In: **Principles of Data Mining and Knowledge Discovery**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001. p. 42–53. ISBN 978-3-540-44794-8.

CLAUSET, A.; NEWMAN, M. E. J.; MOORE, C. Finding community structure in very large networks. **Physical Review E**, American Physical Society (APS), v. 70, 2004.

COMTET, L. **Advanced Combinatorics**. [S.l.]: Reidel, 1974.

COOK, J.; RAMADAS, V. When to consult precision-recall curves. **Stata J.**, v. 20, n. 1, p. 131–148, 2020. ISSN 15368734.

COTTAM, J. A. et al. Evaluation of Alignment: Precision, Recall, Weighting and Limitations. **Proc. - 2020 IEEE Int. Conf. Big Data, Big Data 2020**, p. 2513–2519, 2020.

CRAMMER, K.; SINGER, Y. A family of additive online algorithms for category ranking. **J. Mach. Learn. Res.**, JMLR.org, v. 3, n. null, p. 1025–1058, mar. 2003. ISSN 1532-4435.

DASH, M. et al. Fast hierarchical clustering and its validation. **Data and Knowledge Engineering**, v. 44, n. 1, p. 109–138, 2003. ISSN 0169-023X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0169023X02001386>.

DAVIS, J.; GOADRICH, M. The relationship between precision-recall and roc curves. In: **Proceedings of the 23rd International Conference on Machine Learning**. New York, NY, USA: Association for Computing Machinery, 2006. (ICML '06), p. 233–240. ISBN 1595933832. Disponível em: <https://doi.org/10.1145/1143844.1143874>.

DEMBCZYŃSKI, K. et al. On label dependence in multi-label classification. **Mach. Learn.**, v. 88, n. 1-2, p. 5–45, 2012. ISSN 15730565.

DEMSAR, J. Statistical comparisons of classifiers over multiple data sets. **J. Mach. Learn. Res.**, JMLR.org, v. 7, p. 1–30, dez. 2006. ISSN 1532-4435.

DENG, W. et al. Random Forests. **Machine Learning**, v. 45, p. 5–32, 2001.

DOMANY, E. Superparamagnetic clustering of data — the definitive solution of an ill-posed problem. **Physica A: Statistical Mechanics and its Applications**, v. 263, n. 1, p. 158–169, 1999. ISSN 0378-4371. Proceedings of the 20th IUPAP International Conference on Statistical Physics. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0378437198004944>.

ELISSEEFF, A.; WESTON, J. A kernel method for multi-labelled classification. In: **Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic**. Cambridge, MA, USA: MIT Press, 2001. (NIPS'01), p. 681–687.

ENDUT, N. et al. A Systematic Literature Review on Multi-Label Classification based on Machine Learning Algorithms. **TEM Journal**, UIKTEN - Association for Information Communication Technology Education and Science, v. 11, n. 2, p. 658–666, 2022. ISSN 22178333.

FACELI, K. et al. **Inteligência Artificial. Uma Abordagem de Aprendizado de Máquina**. [S.l.]: LTC, 2011. ISBN 9788521618805.

FAN, R.-E.; LIN, C. A study on threshold selection for multi-label classification. In: . [S.l.: s.n.], 2007.

FAWCETT, T. An introduction to roc analysis. **Pattern Recognition Letters**, v. 27, n. 8, p. 861 – 874, 2006. ISSN 0167-8655. ROC Analysis in Pattern Recognition.

FERRANDIN, M.; CERRI, R. **Multi-label classification via closed frequent labelsets and label taxonomies**. Springer Berlin Heidelberg, 2023. v. 27. 8627–8660 p. ISSN 14337479. ISBN 0050002308048. Disponível em: <https://doi.org/10.1007/s00500-023-08048-5>.

FRIGUI, H. Clustering: Algorithms and applications. In: **2008 First Workshops on Image Processing Theory, Tools and Applications**. [S.l.: s.n.], 2008. p. 1–11.

FÜRNKRANZ, J. et al. Multilabel classification via calibrated label ranking. **Mach. Learn.**, Kluwer Academic Publishers, USA, v. 73, n. 2, p. 133–153, nov. 2008. ISSN 0885-6125.

GANDA, D.; BUCH, R. A survey on multi label classification. **Recent Trends in Programming Languages**, v. 5, 2018. ISSN 2455-1821.

GIBAJA, E. A tutorial on multilabel learning. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 47, n. 3, abr. 2015. ISSN 0360-0300.

GIBAJA, E.; VENTURA, S. Multi-label learning: A review of the state of the art and ongoing research. **Wiley Interdiscip. Rev. Data Min. Knowl. Discov.**, v. 4, n. 6, p. 411–444, 2014. ISSN 19424795.

GOLALIPOUR, K. et al. From clustering to clustering ensemble selection: A review. **Engineering Applications of Artificial Intelligence**, v. 104, p. 104388, 2021. ISSN 0952-1976. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0952197621002360>.

GONCALVES, E. C.; FREITAS, A. A.; PLASTINO, A. A survey of genetic algorithms for multi-label classification. In: **2018 IEEE Congress on Evolutionary Computation (CEC)**. [S.l.: s.n.], 2018. p. 1–8.

GONCALVES, E. C.; PLASTINO, A.; FREITAS, A. A. A genetic algorithm for optimizing the label ordering in multi-label classifier chains. In: **Proceedings of the 2013 IEEE 25th International Conference on Tools with Artificial Intelligence**. USA: IEEE Computer Society, 2013. (ICTAI '13), p. 469–476. ISBN 9781479929719.

GUEHRIA, S. **A Survey on Ensemble Multi-label Classifiers**. Springer Nature Switzerland, 2023. v. 1. 100–109 p. ISBN 9783031275241. Disponível em: <http://dx.doi.org/10.1007/978-3-031-27524-1_11>.

GUVENIR, H. A.; KURTCEPHE, M. Ranking instances by maximizing the area under roc curve. **IEEE Transactions on Knowledge and Data Engineering**, v. 25, n. 10, p. 2356–2366, 2013.

HAN, J.; KAMBER, M.; PEI, J. **Data Mining: Concepts and Techniques**. [S.l.]: Elsevier LTD, Oxford, 2011. ISBN 0123814790.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning**. New York, NY, USA: Springer New York Inc., 2001. (Springer Series in Statistics).

HAYKIN, S. **Redes Neurais: Princípios e Prática**. [S.l.]: Bookman, 2011. ISBN 978-85-7307-718-6.

HERRERA, F. et al. **Multilabel Classification: Problem Analysis, Metrics and Techniques**. 1st. ed. [S.l.]: Springer Publishing Company, Incorporated, 2016. ISBN 3319411101.

HUANG, J. et al. Group sensitive Classifier Chains for multi-label classification. **Proc. - IEEE Int. Conf. Multimed. Expo**, IEEE, v. 2015-Augus, p. 1–6, 2015. ISSN 1945788X.

HUANG, S. J.; ZHOU, Z. H. Multi-label learning by exploiting label correlations locally. In: **Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence**. [S.l.]: AAAI Press, 2012. (AAAI'12), p. 949–955.

HUANG, X. et al. A survey of community detection methods in multilayer networks. **Data Mining and Knowledge Discovery**, Springer, v. 35, p. 1–45, 1 2021. ISSN 1573756X.

HÜLLERMEIER, E. et al. Label ranking by learning pairwise preferences. **Artificial Intelligence**, v. 172, n. 16, p. 1897 – 1916, 2008. ISSN 0004-3702.

IOANNOU, M. et al. Obtaining bipartitions from score vectors for multi-label classification. In: **2010 22nd IEEE International Conference on Tools with Artificial Intelligence**. [S.l.: s.n.], 2010. v. 1, p. 409–416.

JAVED, M. A. et al. Community detection in networks: A multidisciplinary review. **Journal of Network and Computer Applications**, v. 108, p. 87 – 111, 2018. ISSN 1084-8045. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1084804518300560>.

JOLY, A.; GEURTS, P.; WEHENKEL, L. Random forests with random projections of the output space for high dimensional multi-label classification. In: **Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)**. [S.l.: s.n.], 2014. v. 8724 LNAI, n. PART 1, p. 607–622. ISBN 9783662448472. ISSN 16113349.

JUNIOR, J. D. C. **Detecção de novidade em fluxos contínuos de dados multirrótulo**. Dissertação (Mestrado) — Universidade Federal de São Carlos, 2019.

KASHEF, S.; NEZAMABADI-POUR, H.; NIKPOUR, B. **Multilabel Feature Selection: A Comprehensive Review and Guiding Experiments**. John Wiley & Sons, 2018. Disponível em: <https://books.google.com.br/books?id=a1ddtgEACAAJ>.

KAUFMAN, L.; ROUSSEEUW, P. J. **Finding Groups in Data: An Introduction to Cluster Analysis**. [S.l.]: Wiley-Blackwell, 1990.

KAWAI, K.; TAKAHASHI, Y. Identification of the dual action antihypertensive drugs using tfs-based support vector machines. **Chem-Bio Informatics Journal**, v. 9, p. 41–51, 2009.

KIM, P. **MATLAB Deep Learning - With Machine Learning, Neural Networks and Artificial Intelligence**. [S.l.]: Apress, 2017. 1-151 p. ISBN 978-1-4842-2844-9.

KOCEV, D. et al. Ensembles of multi-objective decision trees,. In: **Machine Learning: ECML 2007**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. p. 624–631. ISBN 978-3-540-74958-5.

LESOT, M. J.; RIFQI, M.; BENHADDA, H. **Similarity measures for binary and numerical data: a survey**. [S.l.], 2009. v. 1, 63-84 p.

LI, Y.; YANG, L. More correlations better performance: Fully associative networks for multi-label image classification. **2020 25th International Conference on Pattern Recognition (ICPR)**, p. 9437–9444, 2021.

LIN, S. C.; CHEN, C. J.; LEE, T. J. A Multi-Label Classification with Hybrid Label-Based Meta-Learning Method in Internet of Things. **IEEE Access**, IEEE, v. 8, p. 42261–42269, 2020. ISSN 21693536.

LUACES, O. et al. Binary relevance efficacy for multilabel classification. **Progress in Artificial Intelligence**, p. 303–313, 2012.

LUKASOV, A. Hierarchical agglomerative clustering procedure. **Pattern Recognition**, v. 11, p. 365–381, 1978.

M, R. The map equation. 2009.

MADJAROV, G.; GJORGJEVIKJ, D.; DŽEROSKI, S. Dual layer voting method for efficient multi-label classification. In: **Pattern Recognition and Image Analysis**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 232–239. ISBN 978-3-642-21257-4.

MADJAROV, G. et al. An extensive experimental comparison of methods for multi-label learning. In: . [S.l.: s.n.], 2012. v. 45, p. 3084–3104. ISSN 00313203.

Mahmud, M. S. et al. A survey of data partitioning and sampling methods to support big data analysis. **Big Data Mining and Analytics**, v. 3, n. 2, p. 85–101, 2020.

MCQUITTY, L. L. Similarity analysis by reciprocal pairs for discrete and continuous data. **Educational and Psychological Measurement**, v. 26, n. 4, p. 825–831, 1966. Disponível em: <https://doi.org/10.1177/001316446602600402>.

MELO, A.; PAULHEIM, H. Local and global feature selection for multilabel classification with binary relevance an empirical comparison on flat and hierarchical problems. In: . [S.l.: s.n.], 2017.

MEZO, I. The r-bell numbers. **Journal of Integer Sequences**, v. 14, 2011.

MISHRA, N. K.; SINGH, P. K. Linear Ordering Problem based Classifier Chain using Genetic Algorithm for multi-label classification. **Appl. Soft Comput.**, Elsevier B.V., v. 117, p. 108395, 2022. ISSN 15684946. Disponível em: <https://doi.org/10.1016/j.asoc.2021.108395>.

MITCHELL, T. M. **Machine Learning**. [S.l.]: McGraw-Hill Education, 1997. ISBN 0070428077.

MITTAL, H. et al. A comprehensive survey of image segmentation: clustering methods, performance parameters, and benchmark datasets. **Multimedia Tools and Applications**, Multimedia Tools and Applications, v. 81, n. 24, p. 35001–35026, 2022. ISSN 15737721.

MITTAL, M. et al. Clustering approaches for high-dimensional databases: A review. **Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery**, Wiley-Blackwell, v. 9, 5 2019. ISSN 19424795.

MITTAL, R.; BHATIA, M. P. S. Classification and comparative evaluation of community detection algorithms. **Archives of Computational Methods in Engineering**, 2020. ISSN 1886-1784.

MORAL-GARCíA, S. et al. A new label ordering method in classifier chains based on imprecise probabilities. **Neurocomputing**, Elsevier B.V., v. 487, p. 34–45, 5 2022. ISSN 18728286.

MOYANO, J. et al. Combining multi-label classifiers based on projections of the output space using evolutionary algorithms. **Knowledge-Based Syst.**, Elsevier BV, p. 105770, mar 2020. ISSN 09507051.

MOYANO, J. M. An evolutionary approach to build ensembles of multi-label classifiers. **Information Fusion**, v. 50, p. 168 – 180, 2019. ISSN 1566-2535.

MOYANO, J. M. et al. Review of ensembles of multi-label classifiers: Models, experimental study and prospects. **Information Fusion**, v. 44, p. 33 – 45, 2018. ISSN 1566-2535.

MURTAGH, F.; CONTRERAS, P. Algorithms for hierarchical clustering: an overview, ii. **Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery**, v. 7, 2017.

MURTAGH, F.; LEGENDRE, P. Ward's Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward's Criterion? **Journal of Classification**, v. 31, n. 3, p. 274–295, October 2014. Disponível em: <https://ideas.repec.org/a/spr/jclass/v31y2014i3p274-295.html>.

NEWMAN, M. E. J. Fast algorithm for detecting community structure in networks. **Physical Review E**, American Physical Society (APS), v. 69, n. 6, jun 2004.

NEWMAN, M. E. J.; GIRVAN, M. Finding and evaluating community structure in networks. **Physical Review E**, American Physical Society (APS), v. 69, 2004.

NIKOLOSKI, S.; KOCEV, D.; DžEROSKI, S. Structuring the output space in multi-label classification by using feature ranking. v. 10785, p. 122–137, 2018. Disponível em: <http://link.springer.com/10.1007/978-3-319-78680-3>.

OZENNE, B.; SUBTIL, F.; MAUCORT-BOULCH, D. The precision-recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases. **J. Clin. Epidemiol.**, Elsevier Inc, v. 68, n. 8, p. 855–859, 2015. ISSN 18785921. Disponível em: <http://dx.doi.org/10.1016/j.jclinepi.2015.02.010>.

PAKRASHI, A.; NAMEE, B. M. Cascademl: An automatic neural network architecture evolution and training algorithm for multi-label classification (best technical paper). In: **Artificial Intelligence XXXVI**. Cham: Springer International Publishing, 2019. p. 3–17. ISBN 978-3-030-34885-4.

PAPANIKOLAOU, Y.; TSOUMAKAS, G.; KATAKIS, I. Hierarchical partitioning of the output space in multi-label data. **Data & Knowledge Engineering**, v. 116, p. 42 – 60, 2018. ISSN 0169-023X.

PEREIRA, R. B. et al. Correlation analysis of performance measures for multi-label classification. **Information Processing & Management**, v. 54, n. 3, p. 359 – 369, 2018. ISSN 0306-4573.

PLIAKOS, K.; VENS, C.; TSOUMAKAS, G. Predicting drug-target interactions with multi-label classification and label partitioning. **IEEE/ACM Transactions on Computational Biology and Bioinformatics**, v. 18, n. 4, p. 1596–1607, 2021.

PONS, P.; LATAPY, M. **Computing communities in large networks using random walks (long version)**. [S.l.]: arXiv, 2005.

RAGHAVAN, U. N.; ALBERT, R.; KUMARA, S. Near linear time algorithm to detect community structures in large-scale networks. **Physical Review E**, American Physical Society (APS), v. 76, 2007.

READ, J. A pruned problem transformation method for multi-label classification. In: **In: Proc. 2008 New Zealand Computer Science Research Student Conference (NZCSRS**. [S.l.: s.n.], 2008. p. 143–150.

READ, J. **Scalable Multi-label Classification**. Tese (Doutorado) — University of Waikato, 2010.

READ, J.; PFAHRINGER, B. Classifier chains: A review and perspectives. **J. Artif. Int. Res.**, AI Access Foundation, El Segundo, CA, USA, v. 70, p. 683–718, may 2021. ISSN 1076-9757. Disponível em: <https://doi.org/10.1613/jair.1.12376>.

READ, J.; PFAHRINGER, B.; HOLMES, G. Multi-label classification using ensembles of pruned sets. In: **2008 Eighth IEEE International Conference on Data Mining**. [S.l.: s.n.], 2008. p. 995–1000.

READ, J. et al. Classifier chains for multi-label classification. In: **Machine Learning and Knowledge Discovery in Databases**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 254–269. ISBN 978-3-642-04174-7.

REICHARDT, J. Statistical mechanics of community detection. **Physical Review E**, American Physical Society (APS), v. 74, 2006.

REICHARDT, J.; BORNHOLDT, S. Detecting fuzzy community structures in complex networks with a potts model. **Physical Review Letters**, American Physical Society (APS), v. 93, n. 21, nov 2004.

RIVOLLI, A. et al. An empirical analysis of binary transformation strategies and base algorithms for multi-label learning. **Machine Learning**, 2020.

RIVOLLI, A.; SOARES, C.; CARVALHO, A. C. P. d. L. F. d. Enhancing multilabel classification for food truck recommendation. **Expert Systems**, Wiley-Blackwell, 2018.

ROKACH, L.; SCHCLAR, A.; ITACH, E. Ensemble methods for multi-label classification. **Expert Systems with Applications**, v. 41, n. 16, p. 7507 – 7523, 2014. ISSN 0957-4174.

ROSVALL, M.; AXELSSON, D.; BERGSTROM, C. T. The map equation. **The European Physical Journal Special Topics**, Springer Science and Business Media LLC, v. 178, n. 1, p. 13–23, nov 2009.

ROUSSEEUW, P. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. **J. Comput. Appl. Math.**, Elsevier Science Publishers B. V., v. 20, n. 1, p. 53–65, 1987. ISSN 0377-0427. Disponível em: <http://portal.acm.org/citation.cfm?id=38772>.

SANDEN, C.; ZHANG, J. Z. Enhancing multi-label music genre classification through ensemble techniques. In: **SIGIR'11 - Proc. 34th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.** [S.l.: s.n.], 2011. p. 705–714. ISBN 9781450309349.

SHAO, H. et al. Symptom selection for multi-label data of inquiry diagnosis in traditional chinese medicine. **Science China Information Sciences**, v. 56, n. 5, p. 052118–052118, 2013.

SHI, Z. et al. Drift detection for multi-label data streams based on label grouping and entropy. **IEEE International Conference on Data Mining Workshops, ICDMW**, v. 2015-Janua, n. January, p. 724–731, 2015. ISSN 23759259.

SILLA, C. N.; FREITAS, A. A. A survey of hierarchical classification across different application domains. **Data Mining and Knowledge Discovery**, v. 22, n. 1, p. 31–72, Jan 2011. ISSN 1573-756X.

SILVA, F. C. da. **Analise ROC**. 2006.

SILVA, T. C.; ZHAO, L. **Machine Learning in Complex Networks**. [S.l.]: Springer Publishing Company, Incorporated, 2016.

SOROWER, M. A literature survey on algorithms for multi-label learning. **Oregon State University, Corvallis**, p. 1–25, 2010.

SPIVEY, M. Z. A generalized recurrence for bell numbers. **Journal of Integer Sequences**, v. 11, 2008.

SZYMAŃSKI, P.; KAJDANOWICZ, T.; KERSTING, K. How is a data-driven approach better than random choice in label space division for multi-label classification? **Entropy**, v. 18, n. 8, p. 1–23, 2016. ISSN 10994300.

TAHIR, M. A. U. H. et al. A Classification Model for Class Imbalance Dataset Using Genetic Programming. **IEEE Access**, Institute of Electrical and Electronics Engineers Inc., v. 7, p. 71013–71037, 2019. ISSN 21693536.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. **Introduction to Data Mining**. Addison Wesley, 2005. ISBN 0321321367. Disponível em: <http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0321321367>.

THABTAH, F. A.; COWLING, P.; PENG, Y. Mmac: a new multi-class, multi-label associative classification approach. In: **Fourth IEEE International Conference on Data Mining (ICDM'04)**. [S.l.: s.n.], 2004. p. 217–224.

THEODORIDIS, S. **Pattern Recognition, Fourth Edition**. [S.l.]: Academic Press, 2009. ISBN 9781597492720.

THEODORIDIS, S.; KOUTROUMBAS, K. Clustering algorithms ii: Hierarchical algorithms. **Pattern Recognition**, p. 653–700, 2009.

TSOUMAKAS, G. **Mining Multi-label Data**. Boston, MA: Springer US, 2010. 667–685 p. ISBN 978-0-387-09823-4.

TSOUMAKAS, G.; KATAKIS, I. Multi-label classification: An overview. **Int J Data Warehousing and Mining**, v. 2007, p. 1–13, 2007.

TSOUMAKAS, G.; KATAKIS, I.; VLAHAVAS, I. Effective and efficient multilabel classification in domains with large number of labels. **Proc. ECML/PKDD 2008 Work. Min. Multidimens. Data**, p. 30–44, 2008.

TSOUMAKAS, G.; VLAHAVAS, I. Random k-labelsets: An ensemble method for multilabel classification. In: **Machine Learning: ECML 2007**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. p. 406–417. ISBN 978-3-540-74958-5.

TYRALIS, H.; PAPACHARALAMPOUS, G. A Brief Review of Random Forests for Water Scientists and Practitioners and Their Recent History in Water Resources. **Water**, 2019.

UEDA, N.; SAITO, K. Parametric mixture models for multi-labeled text. In: **Advances in Neural Information Processing Systems 15**. [S.l.]: MIT Press, 2003. p. 737–744.

VEMBU, S.; GÄRTNER, T. **Label Ranking Algorithms: A Survey**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. 45–64 p. ISBN 978-3-642-14125-6.

VENS, C. et al. Decision trees for hierarchical multi-label classification. **Mach. Learn.**, v. 73, n. 2, p. 185–214, nov 2008. ISSN 08856125.

VILLE, B. de. Decision trees. **WIREs Computational Statistics**, v. 5, n. 6, p. 448–455, 2013. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/wics.1278>.

WANG, T. et al. A multi-label text classification method via dynamic semantic representation model and deep neural network. **Appl. Intell.**, Applied Intelligence, 2020. ISSN 15737497.

WANG, X. et al. Atc-nlsp: Prediction of the classes of anatomical therapeutic chemicals using a network-based label space partition method. **Frontiers in Pharmacology**, v. 10, 2019. Disponível em: <https://www.frontiersin.org/article/10.3389/fphar.2019.00971>.

WANG, X.; ZHU, X.; YE, M. Sts-nlsp: A network-based label space partition method for predicting the specificity of membrane transporter substrates using a hybrid feature of structural and semantic similarity. **Frontiers in Bioengineering and Biotechnology**, v. 7, p. 306, 2019.

WARD, J. H. Hierarchical grouping to optimize an objective function. **Journal of the American Statistical Association**, v. 58, n. 301, p. 236–244, 1963. Disponível em: <http://www.jstor.org/stable/2282967>.

WENG, W. et al. Multi-label learning based on label-specific features and local pairwise label correlation. **Neurocomputing**, v. 273, p. 385 – 394, 2018. ISSN 0925-2312.

WU, F. Y. The potts model. **Rev. Mod. Phys.**, American Physical Society, v. 54, p. 235–268, Jan 1982. Disponível em: <https://link.aps.org/doi/10.1103/RevModPhys.54.235>.

WU, X.-Z.; ZHOU, Z.-H. A unified view of multi-label performance measures. In: **Proceedings of the 34th International Conference on Machine Learning**. [S.l.]: JMLR.org, 2017. (ICML17, v. 70), p. 3780–3788.

XU, D.; TIAN, Y. A comprehensive survey of clustering algorithms. **Annals of Data Science**, v. 2, p. 165–193, 2015.

XU, J. et al. Joint input and output space learning for multi-label image classification. **IEEE Transactions on Multimedia**, v. 23, p. 1696–1707, 2021.

YE, C. et al. Multi-label active learning with label correlation for image classification. In: **2015 IEEE International Conference on Image Processing (ICIP)**. [S.l.: s.n.], 2015. p. 3437–3441.

YIN, X. Canonical correlation analysis based on information theory. **Journal of Multivariate Analysis**, v. 91, n. 2, p. 161 – 176, 2004. ISSN 0047-259X.

ZHANG. Lift: Multi-label learning with label-specific features. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 37, n. 1, p. 107–120, 2015.

ZHANG, M.; ZHOU, Z. Ml-knn: A lazy learning approach to multi-label learning. **Pattern Recognition**, v. 40, n. 7, p. 2038–2048, 2007. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320307000027>.

ZHANG, M. L. et al. **Binary relevance for multi-label learning: an overview**. [S.l.]: Higher Education Press, 2018. 191–202 p.

ZHANG, M.-L.; WU, L. A review on multi-label learning algorithms. **Knowledge and Data Engineering, IEEE Transactions on**, n. 99, p. 1, 2014. ISSN 1041-4347.

ZHANG, M.-L.; ZHOU, Z.-H. Multilabel neural networks with applications to functional genomics and text categorization. **IEEE Transactions on Knowledge and Data Engineering**, v. 18, n. 10, p. 1338–1351, 2006.

ZHENG, X. et al. A Survey on Multi-Label Data Stream Classification. **IEEE Access**, v. 8, p. 1249–1275, 2020.

ZHOU, J. P. et al. Iatc-nrakel: An efficient multi-label classifier for recognizing anatomical therapeutic chemical classes of drugs. **Bioinformatics**, v. 36, n. 5, p. 1391–1396, 2020. ISSN 14602059.

ZHOU, Q. M. et al. A relationship between the incremental values of area under the ROC curve and of area under the precision-recall curve. **Diagnostic Progn. Res.**, Diagnostic and Prognostic Research, v. 5, n. 1, 2021.

ZHU, Y.; KWOK, J. T.; ZHOU, Z. Multi-label learning with global and local label correlation. **IEEE Transactions on Knowledge and Data Engineering**, v. 30, n. 6, p. 1081–1094, 2018.