**UNIVERSIDADE FEDERAL DE SANTA CATARINA**
**DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA**

Rodolfo Lorbieski

# MODELO DE ENSEMBLES MULTINÍVEIS PARA CLASSIFICADORES

Florianópolis

2018

Rodolfo Lorbieski

# MODELO DE ENSEMBLES MULTINÍVEIS PARA CLASSIFICADORES

Dissertação submetida ao Programa de Pós-Graduação em Ciência da Computação para a obtenção do Grau de Mestre em Ciência da Computação.
Orientadora: Prof. Dra. Silvia Modesto Nassar

Florianópolis

2018

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Lorbieski, Rodolfo
Modelo de Ensembles Multiníveis para Classificadores / Rodolfo Lorbieski ; orientadora, Silvia Modesto Nassar, 2018.
89 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós Graduação em Ciência da Computação, Florianópolis, 2018.

Inclui referências.

1. Ciência da Computação. 2. Ensemble. 3. Multi Camadas. 4. Aprendizagem de Máquina. 5. Famílias. I. Nassar, Silvia Modesto. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Ciência da Computação. III. Título.

Rodolfo Lorbieski

**MODELO DE ENSEMBLES MULTINÍVEIS PARA CLASSIFICADORES**

Esta dissertação foi julgada adequada para obtenção do título de mestre e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Florianópolis, 16 de março de 2018.

Prof. José Luís Almada Güntzel, Dr.
Coordenador do Programa

**Banca Examinadora:**

Prof. Silvia Modesto Nassar, Dra.
Universidade Federal de Santa Catarina
Orientadora

Prof. Jonny Carlos da Silva, Dr.
Universidade Federal de Santa Catarina

Prof. Mauro Roisenberg, Dr.
Universidade Federal de Santa Catarina

Prof. Mário Antonio Ribeiro Dantas, Dr.
Universidade Federal de Juiz de Fora (Videoconferência)

JOSÉ LUÍS ALMADA GÜNTZEL
Coordenador do PPGCC/UFSC
Portaria nº 2392/2017/GR, de 27/10/2017

*Dedico essa dissertação aos meus maiores exemplos desde que nasci: meus pais e meus irmãos. Também dedico de um modo especial as pessoas que tornaram esse trabalho possível, seja por seu apoio, companheirismo ou amizade: minha orientadora e minha namorada.*

## AGRADECIMENTOS

O espaço aqui limitado e o curto tempo disponível, seguramente, não me permite agradecer como devia a todos que contribuíram de alguma forma pra que eu concluísse o meu mestrado. Esses anos de pesquisa foram uma árdua, porém gratificante, jornada de desafios. E esses desafios não se tornariam realidade sem os diversos apoios a mim retribuídos.

Primeiro de tudo, gostaria de agradecer a Deus por me guiar e me dar tranquilidade para seguir em frente com os meus objetivos e não desanimar com as diversas dificuldades encontradas no meio do caminho.

Gostaria muito de um agradecimento especial a minha orientadora Prof. Dra. Silvia Nassar, que é um exemplo pra mim de liderança, humildade e solidariedade. Preocupada com seus orientandos, sabe transmitir motivação e esta sempre disposta a ajudar. Mesmo sem trabalhar comigo anteriormente, confiou nas recomendações concedidas a mim e no meu currículo pra entrada na pós-graduação. Também agradeço aos demais membros da banca examinadora da minha dissertação por toda contribuição: Prof. Dr. Jonny Carlos da Silva, Prof. Dr. Mauro Roisenberg e o Prof. Dr. Mário Antonio Ribeiro Dantas.

Agradeço aos meus pais, Adolfo e Maria, que sempre me apoiaram de diversas formas, me deram atenção, sempre estiveram perto de mim mesmo numa cidade distante e mostraram a mim e a meus irmãos o valor da honestidade, humildade e o quanto era importante estudar.

Agradeço aos meus irmãos Robson e Rodrigo pelo companheirismo inigualável, com a melhor definição prática possível do que é fraternidade. Devo agradecimento especial ao Robson por me receber em Florianópolis quando nem tinha ideia onde poderia morar, além dos diversos apoios nesses três anos de mestrado.

Agradeço a minha querida namorada/esposa, Jhenifer, por ser tão importante, não apenas pela incansável boa vontade em me ajudar, sempre me fazendo acreditar no final dessa difícil etapa, mas também pela enorme parceria em casa. Aqueles momentos onde ela não me deixava jogar, descansar demasiadamente, que ela pegava no meu pé, foram essenciais para o resultado final.

Algumas pessoas em particular também devo agradecer, como o prof. Dr. João Candido Bracarense Costa, meu orientador em projetos nos tempos de graduação, pela carta de recomendação que com seu prestígio certamente contribuiu muito para minha entrada. Aos amigos Gustavo Paetzold e Rafael Weingartner pelas ajudas nas revisões dos artigos em inglês.

Como não poderia deixar de ser, aproveito a oportunidade pra agradecer aos meus amigos de infância Bruno Popik, Diego Alves Mariano e Edér José Chagas Machado por toda uma vida de amizade e ótimas conversas até hoje. Aos amigos mais recentes e próximos do laboratório Ju, Fábio e Almir na UFSC.

Agradeço ainda à CAPES e ao SENAI pelo financiamento em todos esses anos de pesquisa. Em particular ao SENAI, agradeço a equipe pela compreensão e liberação nas épocas mais dificeis da dissertação para conclusão da mesma. É muito gratificante poder aplicar os conhecimentos adquiridos na academia para inovação na industria. Em especial, no SENAI, agradeço ao Dr. Márcio da Silva Arantes pela humildade e liderança no ambiente de trabalho, pelos inúmeros aprendizados na área de otimização matemática, de algoritmos genéticos, por toda troca de conhecimento em IA, além das boas conversas.

Agradeço tambem a todos que me ajudaram diretamente ou indiretamente que eu não lembrei de citar aqui.

Muito obrigado a todos!

*"Não confunda derrotas com fracasso nem vitórias com sucesso. Na vida de um campeão sempre haverá algumas derrotas, assim como na vida de um perdedor sempre haverá vitórias. A diferença é que, enquanto os campeões crescem nas derrotas, os perdedores se acomodam nas vitórias."*

Roberto Shinyashiki

## RESUMO

Um comitê de máquinas, ou *ensemble*, é uma combinação de diversos classificadores por meio de uma estratégia pré-estabelecida. Seu uso tem sido comum na literatura para garantir um aumento de generalização nos problemas de classificação. Entretanto, é fundamental o uso de uma boa estratégia de diversidade para assegurar a qualidade dos resultados. Para tanto, a presente pesquisa propõe a construção de um modelo multinível, onde a decisão final é realizada por meio da combinação das saídas de *ensembles*. A esse modelo refere-se aqui como comitê de *ensembles*. O tema da presente dissertação buscou avançar o estado da arte ao propor uma estratégia para a realização do comitê de *ensembles*. Propôs-se ainda a combinação de *ensembles* que tenham em sua formação classificadores com similaridades entre si. Dessa forma, cada *ensemble* do comitê especializa-se em determinado paradigma de aprendizagem (família). Busca-se com isso um aumento ainda maior da diversidade. A aplicação do modelo proposto (nível 2) ocorreu em bases de dados públicas com diferentes características e sua avaliação foi mensurada por meio da acurácia, área sob a curva ROC (AUC) e tempo de execução. Os resultados mostraram semelhanças de desempenho dos níveis 0 e 1. O modelo proposto conseguiu um crescimento médio de até 14% e 10% em relação à, respectivamente, acurácia e área sob a curva ROC dos níveis 0 e 1. A família que apresentou os melhores resultados foi a Bayesiana. Os resultados demonstraram que o desempenho da família bayesiana foi 949 vezes mais rápido no tempo de execução que o comitê de ensembles com os resultados de acurácia e área sob a curva ROC mais estáveis e levemente superior às demais famílias (nível 1). Por fim, a análise estatística, com um nível de significância de 5% ($\alpha = 0,05$), comprovou o bom desempenho do comitê de *ensembles* em quase todas as comparações em relação aos demais níveis tanto em termos de acurácia quanto de área sob a curva ROC, embora com um alto tempo de execução.


**Palavras-chave:** Ensemble, Multi Camadas, Aprendizagem de Máquina, Famílias.

## ABSTRACT

A committee machine, or ensemble, is a combination of several classifiers by means of a pre-established strategy. Its use has been common in the literature to ensure an increase the generalization in classification problems. However, a good diversity strategy is essential to ensure the quality of results. Therefore, the present research proposes the construction of a multi-level model, where the final decision is made through the combination of ensembles outputs. This model is referred to here as an committee ensembles. The theme of this dissertation sought to advance the state of the art by proposing a strategy for the accomplishment of the committee ensembles. It's also proposed the combination of ensembles that have in their formation classifiers with similarities among themselves. Therefore, each committee ensemble specializes in a particular learning paradigm (family). An increase in diversity is thus sought. The validation of the proposed method (level 2) use public databases with different characteristics and its evaluation was measured by means of accuracy, area under the ROC curve (AUC) and processing time. The results showed similarities of performance of levels 0 and 1. The proposed model achieved an average growth of up to 14% and 10% in relation to, respectively, accuracy and area under the ROC curve of levels 0 and 1. The family that presented the best results was Bayesian. The results showed that the performance of the Bayesian family was 949 times faster in the execution time than committee ensembles with the results of accuracy and area under the ROC curve more stable and slightly superior to the other families (level 1). Our results are statistically analyzed with a significance level of 5% ($\alpha = 0.05$), which proved the increased good performance of the ensembles committee in almost all comparisons in relation to other levels both in terms of accuracy and area under the ROC curve, although with a high execution time.


**Keywords:** Ensemble, Multi Layers, Machine Learning, Families.

## LISTA DE FIGURAS

# LISTA DE QUADROS

## LISTA DE TABELAS

## LISTA DE ABREVIATURAS E SIGLAS

| | |
|---|---|
| AIME | *Automated Iterative Multi-tier Ensembles* |
| API | Interface de Programação de Aplicativos |
| *AUC* | Área sob a Curva ROC |
| BN | *Bayes Net* |
| *DT* | *Decision Table* |
| FSG | Fuzzy Stacked Generalization |
| *FURIA* | *Fuzzy Unordered Rule Induction Algorithm* |
| *K-nn* | *K-Nearest Neighbor* |
| LIME | Large Iterative Multitier Ensemble |
| MLR | Regressão Multi-Linear |
| NB | *Naïve Bayes* |
| *RBF* | Função de Base Radial |
| SMC | Sistema Multi-Classificadores |
| *SMO* | Otimização Mínima Sequencial |
| UCI | University of California, Irvine |
| Weka | *Waikato Environment for Knowledge Analysis* |

## SUMÁRIO

# 1 INTRODUÇÃO

Torna-se inviável, para uma grande quantidade de dados a extração de informações sem o uso de ferramentas computacionais (TJADEN; COHEN, 2006). Diante a quantidade de dados existentes na sociedade moderna, técnicas de mineração de dados ganharam destaque. É grande a quantidade de algoritmos de aprendizagem de máquina, também chamado de classificadores, desenvolvidos na literatura. Devido a isso, é comum o agrupamento de classificadores pela forma ou paradigma de aprendizagem dos mesmos, sendo frequentemente denominado esses grupos como famílias (FERNÁNDEZ-DELGADO et al., 2014).

Em comum em todos os classificadores está o objetivo em aumentar o valor da acurácia. No entanto, com o uso de um único classificador, máquina responsável pela atribuição de rótulos aos elementos não classificados, essa tarefa tornou-se cada vez mais difícil (LIMA, 2013). A Figura 1 mostra como pode ser difícil realizar a classificação correta em um problema de apenas duas dimensões com um único classificador. Nota-se pela complexa fronteira de decisão que um único classificador linear ou circular não pode resolver de maneira adequada a separabilidade das classes.

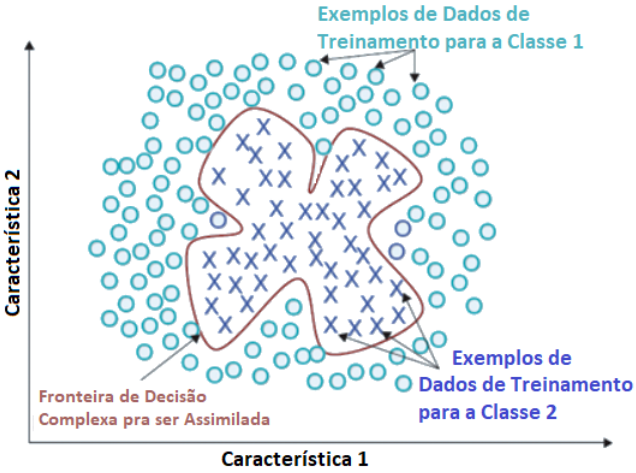


Figura 1 – Fronteira de decisão complexa.

Fonte: Adaptado de Polikar (2006).

O uso de um comitê de máquinas, também conhecido como *ensemble* ou sistema multi-classificadores (SMC) , encontra-se como uma solução para esses problemas. Ela tem como objetivo combinar, por meio de uma estratégia, as saídas de classificadores-base para tomada de decisão. Há um crescente número de pesquisas no tema. Sabe-se hoje que há dois pontos chave que deveriam ser cuidadosamente considerados (CHEN; WONG, 2010). O primeiro diz respeito a diversidade dos componentes do comitê de máquina, onde já ficou provado que para o sucesso do *ensemble* é fundamental "opiniões"diferentes de seus componentes (HANSEN; SALAMON, 1990). Dessa forma, a quantidade dos componentes e o tipo de classificadores usados estão diretamente ligados à diversidade. O segundo ponto chave é selecionar um método de combinação adequado para combinar as saídas dos componentes (CHEN; WONG, 2010).

Diversos métodos *ensembles*, também chamados de meta - classificadores, foram propostos desde então. No geral, os meta - classificadores são categorizados como homogêneos, no caso de os classificadores-base serem formados por um igual mecanismo de aprendizagem; e heterogêneos, no caso da possibilidade de aplicações de diferentes algoritmos de aprendizagem de máquina. Entre os *ensembles* homogêneos destacam-se os métodos *bagging* (BREIMAN, 1996), onde há uma combinação linear simples; e *boosting* (SCHAPIRE, 1990), onde há uma combinação por votação. No caso dos heterogêneos há um destaque para o método *stacking*, que cria um meta - classificador intermediário que usa como dados as saídas dos diferentes classificadores-base que o compõe. Diversos são os algoritmos derivados dos métodos citados, como é o caso do popular *Adaboosting* (FREUND; SCHAPIRE, 1997), uma variação do *boosting*, onde ocorre uma combinação linear ponderada (LIMA, 2012).

Uma ideia para conseguir uma melhora na tarefa de classificação é construir um *ensemble* de *ensembles* ou meta-*ensemble*, onde os componentes do decisor final são formados por *ensembles*. Esse tema tem sido bastante explorado recentemente (ABAWAJY et al., 2016)(OZAY; YARMAN-VURAL, 2016)(JELINEK et al., 2014)(JUREK et al., 2014). Sabe-se que a aprendizagem de múltiplos classificadores é objeto de pesquisa em diferentes comunidades, incluindo reconhecimento de padrões, aprendizagem de máquina, estatística e redes neurais artificiais (MENDES-MOREIRA et al., 2012). Estas comunidades têm diferentes conferências, revistas e muitas vezes usam uma terminologia e notações diferentes. Por vezes, diferentes termos são utilizados para o mesmo conceito. Ocorre também de pesquisadores criarem nomeações diferentes entre

as mesmas etapas do processo. Tudo isso pode gerar confusão entre pesquisadores interessados. A seguir são listados vários sinônimos dos principais vocábulos (MENDES-MOREIRA et al., 2012) citados nesta dissertação.

- Comitê de máquinas: *Ensemble*, Sistemas Multi-Classificadores.
- Classificador: Modelo, Indutor, Hipótese, Preditor.
- Instância: Exemplo, Caso, Objeto.

Tendo em vista o que foi dito, este trabalho aborda a criação de um novo comitê de *ensembles* por meio de uma variação do método *Stacking*, usando como estratégia de diversidade a criação e posterior combinação de *ensembles* especializados em quatro das principais famílias de classificadores existentes: família de classificadores probabilísticos ou bayesianos, classificadores com aprendizagem baseado em distância (*lazy*), baseado em regras e em funções. Foi tomado como referência os paradigmas de aprendizagem (famílias) da ferramenta *Weka*, versão 3.7, ferramenta essa usada como auxiliar durante toda a pesquisa para a execução dos algoritmos de aprendizagem de máquina.

## 1.1 OBJETIVOS

### 1.1.1 Objetivo Geral

Propor um modelo de comitês de máquinas multi-nível baseado em paradigmas de aprendizagem, a fim de promover uma otimização na acurácia e na área sobre a curva ROC nos problemas de classificação.

### 1.1.2 Objetivos Específicos

- Identificar categorias de diferentes classificadores existentes.
- Desenvolver e implementar um modelo de comitê de máquinas hierárquico de três níveis utilizando diferentes técnicas de combinação de classificadores.
- Avaliar o modelo proposto utilizando medidas de tempo, área sob a curva ROC e acurácia sobre cada nível usando bases de dados públicas.

## 1.2 ORGANIZAÇÃO DO TRABALHO

O restante da presente dissertação é composto por mais 5 seções. A seção 2 trata dos fundamentos teóricos e conceituais necessários ao entendimento desse trabalho, contendo técnicas de classificação presentes na literatura, meios de avaliação em classificadores, os diferentes paradigmas de aprendizagem (famílias) e as diferentes técnicas de combinação de classificadores. Na seção 3 é apresentado um estado da arte da presente pesquisa, citando trabalhos atuais que envolvem especialmente comitês de *ensembles*. A seção 4 foca nos procedimentos metodológicos aplicados expondo em alto nível como foi desenvolvido a pesquisa, além de apresentar com mais detalhes os meios utilizados para obtenção dos resultados. Expõe-se aqui as bases de dados utilizadas, os algoritmos de classificação aplicados, configurações dos experimentos e descrição da análise estatística. A seção 5 reservou-se a apresentação da discussão e dos resultados por meio da apresentação de gráficos e tabelas, separando sub-seções para cada medida de avaliação. Por fim, na seção 6 é apresentada as conclusões, limitações e trabalhos futuros.

# 2 FUNDAMENTAÇÃO TEÓRICA

Pessoas muitas vezes são propensas a cometer erros durante uma análise de dados ou mesmo ao tentar estabelecer relações entre os vários atributos de seus dados. Isso torna difícil encontrar manualmente soluções para certos problemas. A aprendizagem de máquina pode muitas vezes ser aplicada com sucesso a estes problemas, melhorando a eficiência dos sistemas e os projetos de máquinas (KOTSIANTIS, 2007).

Classificação é um das mais populares aplicações da mineração de dados (CHEN; WONG, 2010). Classificadores são uma ferramenta inestimável para muitas tarefas, tais como previsões médicas ou genômica, detecção de *spam*, reconhecimento facial e finanças (BOST et al., 2015).

Métodos de aprendizagem *ensemble* treinam combinações de classificadores individuais (podendo ser diferentes tipos de classificadores ou diferentes instâncias do mesmo classificador), chamados classificadores - base (OZA; RUSSELL, 2001). Quando as correlações dos erros feitas pelos classificadores base são baixas, há uma forte tendência de superioridade no desempenho (em termos de acurácia) dos *ensembles*, o que justifica a sua popularidade no meio científico (FREUND; SCHAPIRE, 1996) (TURNER; OZA, 1999).

As próximas subseções apresentam uma fundamentação teórica da aprendizagem supervisionada, características e funcionamento de diferentes modelos de classificadores, estratégias para combinação dos mesmos e como foram realizadas as avaliações nos problemas de classificação.

## 2.1 APRENDIZAGEM SUPERVISIONADA

Na terminologia de aprendizagem de máquina, a classificação é considerado uma instância da aprendizagem supervisionada (ALPAYDIN, 2010), cujo objetivo é criar um mapeamento entre um conjunto de variáveis de entrada X e uma variável de saída Y através de observações dos dados de treinamento (Figura 2) (TAN, 2013).


Figura 2 – Tarefa de classificação.
Fonte: Adaptado de Tan (2013).

O procedimento sem supervisão correspondente é conhecido como *clustering*, que não requer uma saída pré-definida nos dados de treinamento, e envolve agrupamento de dados em categorias com base em alguma medida de similaridade inerente ou distância (OZGUR, 2004). No presente trabalho, considera-se apenas aprendizagem supervisionada.

Conforme Bernardini (2002), a qualidade e a quantidade dos atributos e exemplos à disposição no treinamento são os fatores mais importantes para a qualidade dos classificadores (BERNARDINI, 2002). Como algoritmos de indução, em geral, se baseiam nos dados de treinamento para construção do classificador, é fundamental uma representação adequada dos exemplos, sem inconsistências ou incoerências (BERNARDINI, 2002).

Segundo Pila (2001), o algoritmo de indução trabalha com um conjunto de exemplos de treinamento – que em geral são obtidos através de algum especialista – no qual cada exemplo é constituído de um vetor com os valores dos atributos e as classes, e tem como tarefa induzir um classificador capaz de predizer a qual classe pertence um novo exemplo (PILA, 2001).

Existem diversos tipos de variáveis dependentes na classificação no resultado da aprendizagem supervisionada. Segundo Costa (2014), os principais tipos são destacados no Quadro 1 (COSTA, 2014).

Quadro 1 – Principais tipos de classificação.

| **Tipo de Classificação** | **Variável Dependente** | **Exemplos** |
|---|---|---|
| Nominal | Possui elementos qualitativos. | Determinar espécie de uma planta de acordo com suas características. |
| Binária | Há apenas duas classes possíveis para classificação. | Determinar de acordo com dados se uma informação é verdadeira ou falsa. |
| Multi-classe | Existem múltiplas classes. | Dizer qual o modelo do carro, entre diversas possibilidades, a partir de suas características. |
| Ordinal | Elementos qualitativos que apresentem ideia de ordem. | Definir se um filme é ruim/bom/ótimo de acordo com seu gênero e premiações. |
| Regressão | Valor numérico real. | Estimar altura aproximada de uma criança de acordo com sua idade. |
| *Ranking* | Classificação por ordem. | Classificação de páginas web com maior relevância em motores de busca. |

Fonte: Baseado em Costa (2014)

## 2.2 TÉCNICAS DE CLASSIFICAÇÃO

Diversos são os classificadores de aprendizagem existentes. Os autores Delgado et al. (2014) analisaram 179 classificadores na literatura, nas quais foram divididos em 17 categorias denominadas famílias, sendo exemplos as árvores de decisão, regressão logística, máquinas de vetores de suporte, redes neurais, entre outros (FERNÁNDEZ-DELGADO et al., 2014). A ferramenta de mineração de dados *Weka* (WITTEN et al., 2016), utilizada na presente pesquisa, por sua vez, separa os classificadores em categorias como: Classificadores Bayesianos, funções, *lazy*, regras, árvores e meta. A seguir são detalhadas as famílias utilizadas na presente dissertação.

### 2.2.1 Famílias

O presente trabalho dividiu as famílias da mesma forma que a *API Weka*, utilizando no entanto, apenas as categorias de classificadores bayesianos, funções, *lazy* e regras e meta. A seção a seguir define as técnicas de classificações separadas por famílias utilizadas pela presente pesquisa.

### 2.2.1.a Bayes

Classificadores Bayesianos exploram relações probabilisticas entre as variáveis preditoras e a(s) variável(is) da classe de saída. São exemplos dessa categoria os classificadores *BayesNet (BN)* e *Naïve Bayes (NB)* . Enquanto o classificador *Naïve Bayes* (DOMINGOS; PAZZANI, 1997) estima a probabilidade condicional de classe com base no teorema de Bayes e só pode representar distribuições simples, a rede Bayesiana é um modelo gráfico probabilístico e pode representar independências condicionais entre variáveis (WEISS; KULIKOWSKI, 1991)(PENG et al., 2009), numa forma de causa e efeito.

A Figura 3 ilustra uma rede bayesiana que decide em qual categoria o preço de venda se situa de acordo com as probabilidades da ocorrência de três critérios: número de quartos, número de banheiros e proximidade ao transporte público.

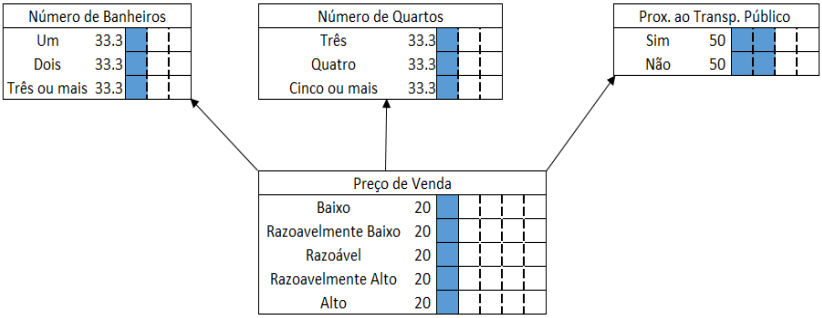


Figura 3 – Rede Bayesiana.
Fonte: Do autor.

Como observado na Figura 3, uma rede Bayesiana consiste de três elementos principais:

- Nodos: Graficamente exibidos como caixas que representam as variáveis do sistema. Um nodo consiste de um conjunto finito de estados mutuamente exclusivos que representam os valores (em probabilidade) que a variável pode assumir.

- Arcos: Graficamente exibidos como flechas que representam relação de causalidade entre nodos.

- Tabelas de probabilidades condicionais: ficam ocultas entre os nodos e os arcos. Cada tabela define o relacionamento quantitativo entre um nó filho e seus nodos principais.

#### 2.2.1.b Funções

Exploram funções matemáticas como aprendizagem para os classificadores. São exemplos dessas categorias o classificador *RBF*, que implementa uma rede neural artificial que usa funções de base radial como funções de ativação (KALYANI; LAKSHMI, 2012) e o classificador *SMO*, que é um algoritmo de otimização mínima sequencial para classificação de vetor de suporte (JAMIL, 2016). A Figura 4 ilustra uma rede neural com Função de Ativação de Base Radial (RBF) que consiste em um modelo neural multicamadas, capaz de aprender padrões complexos e resolver problemas não-linearmente separáveis. A arquitetura de uma rede RBF tem três camadas: camada de entrada, na qual os padrões são apresentados à rede; a camada intermediária (única) que aplica uma transformação não linear do espaço de entrada para o espaço escondido (alta dimensionalidade); e camada de saída que fornece a resposta da rede ao padrão apresentado. A camada escondida possui as funções de base radial, que é caracterizada por apresentar uma resposta que decresce (ou cresce) monotonicamente com a distância a um ponto central.

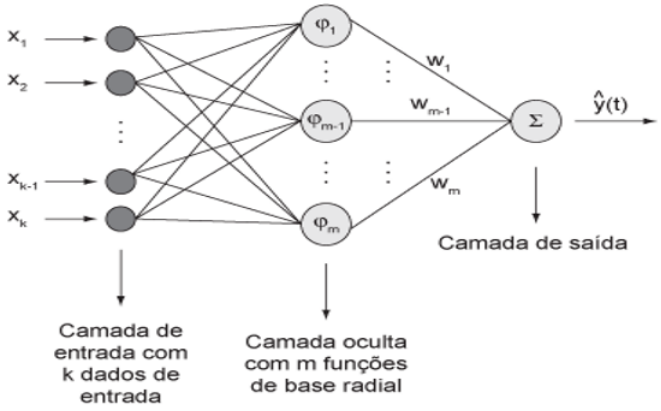


Figura 4 – Rede Neural de Base Radial.
Fonte: Do autor.

#### 2.2.1.c *Lazy*

Paradigma de aprendizagem que acessa os dados de treinamento apenas quando vai classificar um novo objeto (JADHAV; CHANNE, 2016). Dessa forma, não é construído um modelo explicitamente (HAN; PEI; KAMBER, 2011), até que uma nova instância não marcada seja classificada. Os algoritmos de aprendizado *lazy* exigem menos tempo de computação durante a fase de treinamento do que os os demais paradigmas de aprendizagem, mas um tempo maior de computação durante o processo de classificação (SOMAN; DIWAKAR; AJAY, 2006) (KOTSIANTIS, 2007). São exemplos desse paradigma os algoritmos *K-nn* e o *K-Star*. O algoritmo K-vizinho mais próximo (*K-nn*) é o mais simples de todos os algoritmos de aprendizagem de máquina (JADHAV; CHANNE, 2016). Baseia-se no princípio de que as amostras que são semelhantes, geralmente estão próximas (COVER; HART, 1967). O algoritmo *K-Star* (K*) é um classificador *lazy* que classifica uma instância, comparando-a com um banco de dados de exemplos pré-classificados (CLEARY; TRIGG et al., 1995). O pressuposto fundamental é que casos similares terão classificações semelhantes. A Figura 5 ilustra o funcionamento do algoritmo K-nn: O padrão no centro, no caso da utilização dos 5 vizinhos mais próximos, fica classificado como triângulo, por outro lado, o uso dos 10 vizinhos mais próximos classifica o padrão do centro como uma estrela.

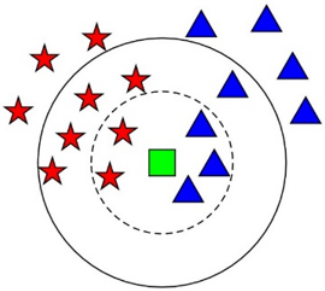

Figura 5 – Classificador K-nn.
Fonte: Do autor.

#### 2.2.1.d Regras

A aprendizagem baseada em regras tem responsabilidade de identificar uma regra que cobre as instâncias de uma certa classe na saída (e excluir outras instâncias que não estejam naquela classe), separa-las, e o procedimento continua sobre aquelas instancias que ainda não foram identificados seguindo uma estratégia de separar-e-conquistar (WITTEN et al., 2016). Existem muitas critérios para a adição de regras a cada estágio que, obviamente, têm um efeito significativo no resultado final (regras produzidas finais) (STEINER et al., 2004). São exemplos dessa família de classificadores o algoritmo *FURIA* e *Decision Table (DT)* . No *Decision Table* procura-se pelos atributos que possam decidir pelo melhor resultado. A idéia é que atributos irrelevantes para a decisão sejam eliminados das regras (STEINER et al., 2004). O *FURIA* por sua vez é um método de classificação baseados em regras difusas (HÜHN; HÜLLERMEIER, 2009), sendo uma melhoria do algoritmo *RIPPER* (COHEN, 1995), uma abordagem direta para gerar regras de classificação que usa poda incremental repetida para causar redução de erros (YANG et al., 1999). A figura 6 ilustra a criação de um classificador por meio de cinco regras pré-definidas. Ao fim, o classificador baseado em regras classifica 3 animais por meio de suas características.

R1: (Ovíparo = sim) ^ (Pode voar = sim) → Pássaros
R2: (Ovíparo = sim) ^ (Vive na água = sim) → Peixes
R3: (Ovíparo = não) ^ (Tipo de sangue = quente) → Mamíferos
R4: (Ovíparo = sim) ^ (Pode voar = não) → Répteis
R5: (Vive na água = Algumas vezes) → Anfíbios

| Nome | Tipo Sanguíneo | Ovíparo | Voa | Vive na água | Classe |
|---|---|---|---|---|---|
| Lêmure | Quente | Não | Não | Não | ? |
| Tartaruga | Frio | Sim | Não | Alg. vezes | ? |
| Gavião | Quente | Sim | Sim | Não | ? |

Figura 6 – Classificadores baseados em regras.
Fonte: Do autor.

### 2.2.2 Meta-Classificadores

Os desenvolvimentos recentes na teoria da aprendizagem computacional levaram a métodos que melhoram o desempenho ou ampliaram as capacidades desses esquemas básicos de aprendizagem. Por operarem na saída de outros classificadores esses métodos de aprendizado são chamados de meta-classificadores (PARENTE, 2012). A razão pela qual os meta classificadores geralmente superam outros métodos reside na natureza da diversidade do erro, sempre herdado em uma classificação (TODOROVSKI; DŽEROSKI, 2003).

Destacam-se entre os meta-classificadores os algoritmos *bagging*, *boosting* e *stacking* e a escolha do método depende de preferências e o tipo de questão em mãos (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

De uma forma geral esses classificadores tem potencial para aumentar a capacidade de generalização na tarefa de classificação. No entanto, um preço a pagar no uso dos meta-classificadores é a "caixa-preta" que existe em seus algoritmos de indução (CORTEZ; EMBRECHTS, 2013). Enquanto classificadores baseados em regras ou árvores de decisão fornecem meios que os seres humanos podem explorar pra entender

por que o modelo classifica dessa forma, meta-classificadores de um modo geral não permitem a compreensão do que está por trás da tomada de decisão melhorada (CORTEZ; EMBRECHTS, 2013).

Além dos clássicos meta-algoritmos *bagging*, *boosting* e *stacking*, as próximas subseções explicam o funcionamento de cada algoritmo, acrescentando ainda as variações como: *adaboost*, *dagging*, *multiScheme* e *stackingC*.

### 2.2.2.a *Bagging*

O algoritmo *bagging* (*Bootstrap Aggregating*) cria classificadores para o *ensemble* a partir de uma redistribuição do conjunto de treinamento. Quanto à forma de combinação, o modelo utilizado no algoritmo *bagging* é o considerado mais simples e intuitivo (DAISTER, 2007), que é o método por votação majoritária. O conjunto de treinamento de cada classificador é gerado selecionando-se aleatoriamente os exemplos da base de aprendizagem com reposição, conforme Figura 7. Dessa forma, o algoritmo provê a diversidade, lançando-se mão do conceito de redistribuição aleatória dos dados (NASCIMENTO, 2009).

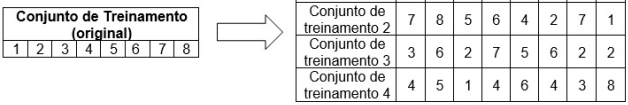


Figura 7 – Redistribuição dos dados via *bagging*
Fonte: Do autor.

### 2.2.2.b *Boosting*

O *boosting* é um meta-algoritmo *ensemble* para reduzir principalmente a polaridade (*bias*), na aprendizagem supervisionada (BREIMAN, 1996). Conforme Figura 8, a diversidade é obtida por aumentar os pesos das amostras mal classificadas de uma forma iterativa (NETO; NASCIMENTO, 2017). O processo iterativo, após distribuição dos dados como no *bagging* (passo 1), consiste em gerar um novo classificador a cada iteração (passo 2). O processo é realizado até que o número desejado de classificadores seja atingido (BRUN, 2017). Por fim, dado a importância de cada classificador do passo 2, ocorre a votação ponderada para a classificação final (passo 3).

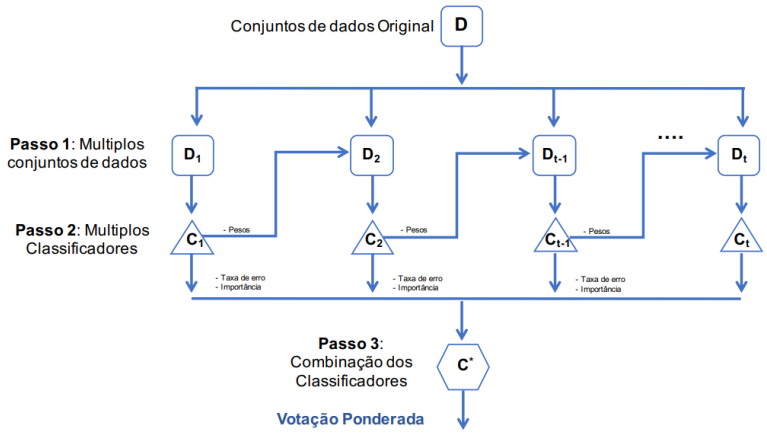


Figura 8 – Funcionamento do Boosting.
Fonte: Retirada de Neto e Nascimento (2017).

Considerado uma versão mais geral do *boosting*, o algoritmo *Adaboost* foi proposto em 1997 por Freund e Schapire (FREUND; SCHAPIRE, 1997). Diversas versões do *AdaBoost* foram desenvolvidas, podendo-se citar o *AdaBoost.*M1 para problemas multi-classes e *AdaBoost.*R específico para problemas de regressão (DAISTER, 2007).

### 2.2.2.c *Stacking*

Wolpert (1992) propôs o quadro de generalização por *Stacking*, que usa uma arquitetura em camadas (WOLPERT, 1992). Classificadores nível-0 recebem como entrada o conjunto de dados original e cada um fornece uma previsão. Essas previsões servirão como entradas em uma tabela de dados intermediária, onde um meta-classificador no nível-1 realiza o treinamento e por fim fornece a predição final. A Figura 9 ilustra o funcionamento do algoritmo *Stacking* com três classificadores – base diferentes.

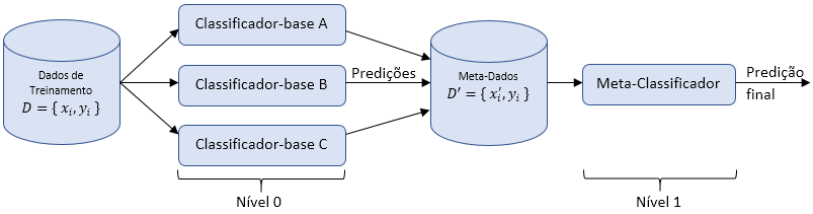


Figura 9 – Funcionamento do *Stacking*.
Fonte: Do autor.

Embora *Stacking* seja menos comum do que *Bagging* e *Boosting* em problemas do mundo real (SESMERO; LEDEZMA; SANCHIS, 2015), o crescimento exponencial dos dados de diferentes formas na era moderna faz do *Stacking* uma interessante alternativa para geração de *ensembles*.

#### 2.2.2.d Variantes

Diversos são os *ensembles* formados por variantes dos três meta - classificadores citados anteriormente. Entre eles, destacam-se aqui os usados na presente dissertação:

- *Dagging*: Um meta - classificador homogêneo semelhante ao *Bagging*, que fornece subconjuntos disjuntos dos dados de treinamento para os classificadores - base (KOTSIANTI; KANELLOPOULOS, 2007). A previsão final é dado pelo voto majoritário (TING; WITTEN, 1997b).

- *MultiScheme*: Meta - classificador heterogêneo que seleciona o melhor dos classificadores - base candidatos utilizando-o com validação cruzada nos dados de treinamento (WITTEN et al., 2016).

- *StackingC*: De acordo com Ting & Witten (1997), para que ocorra sucesso no *Stacking* é necessário usar probabilidades de saída das classes ao invés de predições de classes (TING; WITTEN, 1997a). Para resolver isso, foi criado uma versão mais eficiente do *Stacking* chamada *StackingC*. Entre os objetivos do *StackingC* destaca-se a busca pela otimização de desempenho em conjuntos de dados multiclasse no algoritmo *Stacking* (SEEWALD, 2002).

## 2.3 *BIAS* VERSUS VARIÂNCIA

O dilema *Bias*-Variância é uma ferramenta chave para a compreensão de algoritmos de aprendizagem de máquina, e nos últimos anos o seu uso em estudos empíricos cresceu rapidamente (DOMINGOS, 2000). As noções de *bias* e de variância ajudam a explicar como classificadores muito simples podem superar os mais sofisticados e como comitês de máquinas podem superar classificadores-base (DOMINGOS, 2000).

De uma maneira simples, pode-se dizer que o *bias* mede o quão próximo a suposição média do classificador (em todo o conjunto de treinamento) corresponde ao alvo (quão aproximadamente o indutor erra em média) e a variância mede quanto os erros do indutor vão variar (BAUER; KOHAVI, 1999). Entretanto, como ilustra a Figura 10, ambos conceitos são conflitantes, uma vez que tentativas de redução de *bias* levam a um aumento da variância, e vice-versa (BROWN, 2004). Esse conflito gera a buscada região ótima (ou compromisso ótimo), que é a região onde localiza-se o menor erro de generalização.

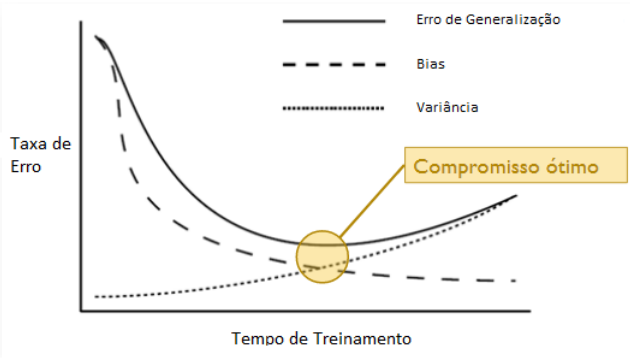


Figura 10 – Dilema *bias*-variância.
Fonte: Adaptada de Brown (2004).

Quando a taxa de *bias* é alta, o classificador não se adapta bem aos dados com os quais foi treinado, causando *underfit* e aumentando o erro de generalização (JOHNSON; TYMMS, 2011). Quando há uma sobrecarga de treinamento, entretanto, há um aumento da variância e ocorre o *overfit*, onde o modelo se adaptou muito bem aos dados de treinamento, porém perde também consigo a capacidade de generalização (JOHNSON; TYMMS, 2011) (poder de classificar corretamente novas

instâncias). A Figura 11 ilustra os conceitos de *underfit* e *overfit* (MARKOWETZ, 2003) em um modelo cujo objetivo é classificar como positivo ou negativo o diagnóstico de um paciente.

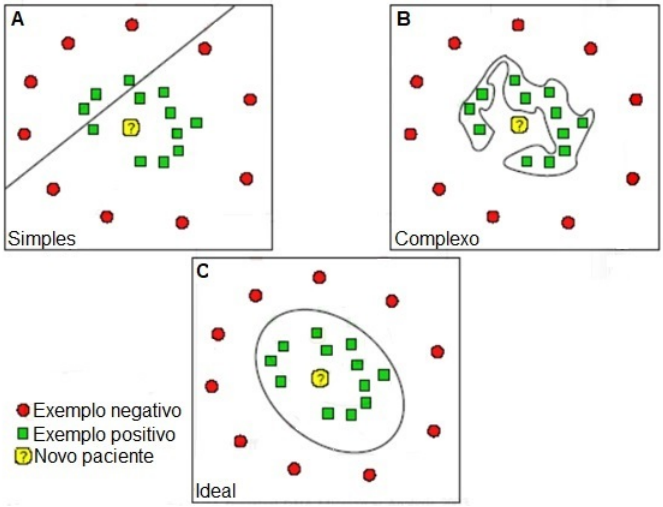


Figura 11 – Exemplos de *underfit* e *overfit*.
Fonte: Adaptada de Markowetz (2003).

Observa-se na Figura 11A um modelo linear, muito simples, onde ocorre um *underfit*, de forma que o modelo não leva em consideração todas as informações presentes nos dados e que não mudará muito diante a classificação de novos pacientes. De um modo contrário, na Figura 11B ocorre um modelo muito complexo que adapta-se muito bem aos dados de treinamento, mas sofre colapso diante classificação de novos pacientes vindos da mesma fonte, perdendo dessa forma a generalização. Um modelo ideal como na Figura 11C fica em um certo limiar entre os modelos demasiadamente simples e o complexo.

Uma solução para abordar este dilema e aumentar capacidade de generalização com o mesmo conjunto de dados é o uso de uma coleção de classificadores ao invés de um. Brown et al. descrevem a noção de um conjunto de preditores e mostram que essa arquitetura pode ser aplicada a qualquer problema de classificação / regressão (BROWN, 2004). Dado que os sistemas multi-classificadores funcionam melhor quando comparados com um único preditor, houve muitas pesquisas procurando melhorar a capacidade de previsão de tais sistemas nos

últimos anos (LIU; YAO; HIGUCHI, 2000)(LIU; YAO, 1999). A próxima seção descreve melhor o funcionamento dos comitês de máquinas.

## 2.4 SISTEMAS MULTI-CLASSIFICADORES

A ideia por trás de sistemas multi-classificadores pode ser descrita como a atividade de construir um modelo preditivo ao integrar múltiplos modelos (ROKACH, 2010). Sabe-se que não existe um classificador dominante para todos as distribuições de dados, conforme Wolpert (1997) cita no teorema *"No free lunch"* (WOLPERT, 1996). Entretanto, espera-se com a integração de modelos uma opinião mais precisa e menos limitada. Embora as vantagens, os conceitos que envolvem os sistemas multi-classificadores são vários. Essa subseção tem como objetivo descrever brevemente os conceitos de topologia e diversidade utilizados em comitês de máquinas.

### 2.4.1 Topologia

O modo como os classificadores se organizam para tomar uma decisão conjunta é chamada de topologia. Existem diversas topologias existentes (ASMITA; SHUKLA, 2014), entre elas, destaca-se a serial e a paralela.

#### 2.4.1.a Topologia Serial

Na topologia serial, classificadores individuais são aplicados um em sequência de outro, implicando em algum tipo de ordenamento entre eles. Esta topologia é adequada quando o custo para realização da classificação é alta, de modo que o classificador primário é o mais barato computacionalmente, e os classificadores secundários têm maior custo de exploração (FUMERA; PILLAI; ROLI, 2004). A Figura 12 ilustra o processo de decisão do comitê de máquinas na topologia horizontal com n classificadores. A ideia é que a saida de um modelo seja usado como entrada no próximo. A predição final é obtida pela classificação do último modelo.


Figura 12 – Topologia Serial
Fonte: Do autor.

#### 2.4.1.b Topologia Paralela

Sendo a mais extensamente utilizada na literatura (KUNCHEVA, 2004), na topologia paralela classificadores são treinados paralelamente diante a um conjunto de dados. A decisão final é tomada pela combinação de decisões de cada classificador-base, que pode ocorrer de diferentes formas, assim como os diversos meta-classificadores citados na subseção 2.2.2. A Figura 13 ilustra o funcionamento do comitê de máquinas na topologia paralela.

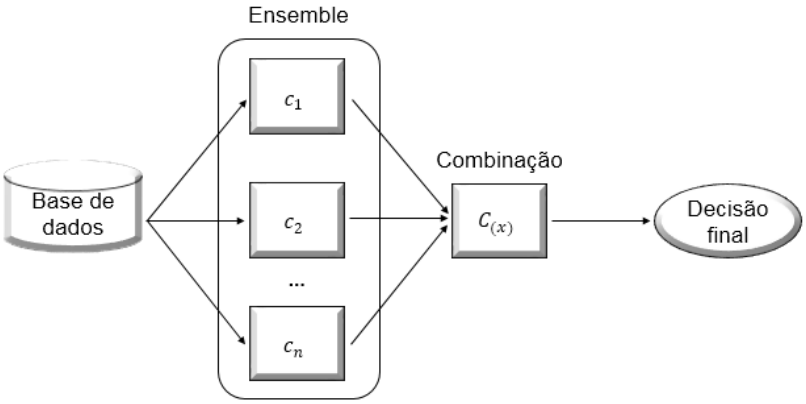


Figura 13 – Topologia Paralela
Fonte: Do autor.

### 2.4.2 Diversidade

Combinar classificadores é interessante apenas se eles discordarem em algumas entradas. Um exemplo disso é quando classificadores combinados via votação, tem baixa diversidade. Se a maioria tem uma

mesma saída e opta por um resultado não adequado o *ensemble* também não terá um bom resultado final. A medida de desacordo entre os membros do *ensemble* é chamada de diversidade. Existem muitas formas de obtê-la, Choi et al (2010) apresenta 76 cálculos de similaridade que podem servir como medida da diversidade (CHOI; CHA; TAPPERT, 2010).

Para ilustrar o conceito e eficácia da diversidade, segue adiante a Figura 14. Conforme ilustração, existe um conjunto de dados com dez instâncias (representado pelas colunas) e três classificadores (representado pelas linhas) que realizam combinações de suas decisões via votação majoritária em 4 situações diferentes. Em uma das situações, os classificadores são independentes, ou seja, possuem diferentes perspectivas sobre os dados, ocasionando uma boa acurácia. Numa segunda situação os classificadores são idênticos (cometem os mesmos erros), assim se um classificador cometer um erro, os classificadores idênticos a esse também cometerão. Na terceira e quarta situação os classificadores são dependentes, ou seja, classificadores que possuem perspectivas muito parecidas em algumas situações. A Figura 14, ilustra como classificadores dependentes podem trazer tanto uma acurácia satisfatória como também insatisfatória, dependendo da distribuição dos dados. Dessa forma, o ideal pra conseguir boa acurácia, sem considerar os dados, é que os classificadores sejam independentes.

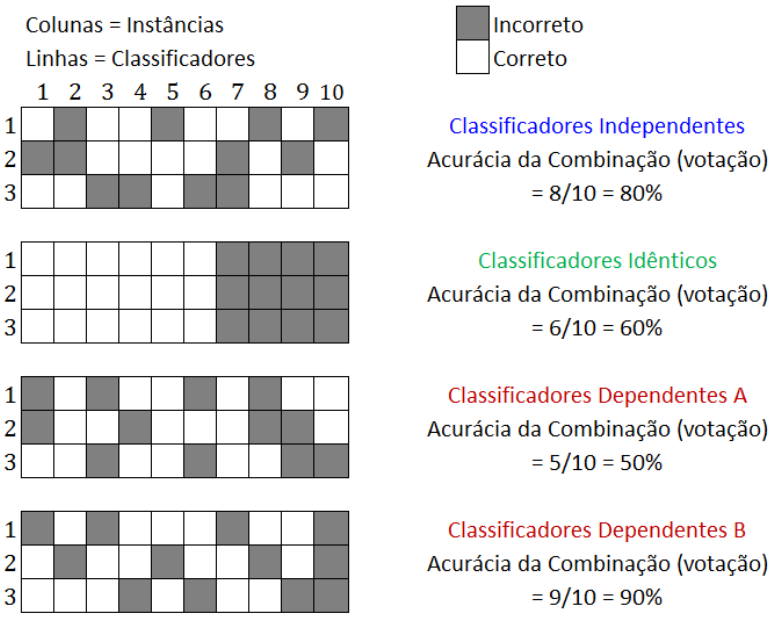


Figura 14 – Ilustração da Diversidade.
Fonte: Adaptado de Ponti (2011).

## 2.5 AVALIAÇÃO DE CLASSIFICADORES

A presente pesquisa utilizou como medidas de avaliação de classificadores a acurácia, área sob a curva ROC e tempo de execução. Tradicionalmente, a métrica usada na avaliação e seleção de modelos de classificação é a acurácia (ou taxa de acerto) estimada. Entende-se como acurácia a razão entre o total de instancias classificadas corretamente e o total de exemplos disponíveis. De um modo geral essa métrica é adequada para boa parte dos casos. No entanto, para problemas desbalanceados (onde há distribuição de amostras desiguais entre as classes) ou quando se deve levar em consideração diferentes custos/benefícios para os diferentes erros/acertos de classificação, a acurácia pode não fornecer informação adequada sobre a capacidade de predição de um classificador em relação a um conjunto de dados (PRATI; BATISTA; MONARD, 2008). Para esses casos, é comum a utilização da área sob a curva ROC (ou *AUC - Area Under ROC*) .

O objetivo das curvas ROC (Receiver Operating Characteristic) é examinar o desempenho de um classificador binário, criando um gráfico dos positivos reais versus falsos positivos para cada limiar de classificação (PRATI; BATISTA; MONARD, 2008). Foi originalmente desenvolvido para avaliar a capacidade de operadores de radar em decidir se uma mancha na tela representava um alvo inimigo (um avião ou um navio) ou uma nave aliada, ou se era um ruído (ALVES, 2014). De fato, ROC é a sigla para "Receiver Operating Characteristic", que pode ser traduzido livremente como "eficiência do operador de recepção de sinais". Trata-se, portanto, de uma medida da capacidade de um observador classificar corretamente um dado dentro de uma chave dicotômica (ALVES, 2014).

Ferramentas de mineração de dados, como *Weka*, fazem automaticamente o cálculo da AUC para os classificadores trabalhados, podendo ser utilizado inclusive para base de dados multi-classe. Diversos são os métodos para o cálculo da AUC, a ferramenta *Weka* em particular calcula a mesma como uma estatística Wilcoxon-Mann-Whitney. Detalhes podem ser encontrados na pesquisa de Waegeman, Baets e Boullart (2006).

As curvas ROC possuem propriedades que as tornam especialmente úteis para domínios com classes desbalanceadas e custos de erros desiguais (ALBERTO; ALMEIDA, 2012). Desse modo, são muitos úteis para se comparar e perceber a relação entre diferentes classificadores, sem se considerar a distribuição das classes ou os erros de má classificação (DRUMMOND; HOLTE, 2006), fornecendo assim informações completas sobre o conjunto de todas as combinações possíveis nas taxas de verdadeiro positivo e de falso positivo.

Por fim, é comum ainda em pesquisas de mineração de dados que o tempo de execução seja também calculado junto com as métricas tradicionais acima destacadas. Geralmente o tempo de execução é influenciado por fatores de treino e pela técnica escolhida. Na presente pesquisa considerou tempo de execução todo o processamento, desde o começo do treinamento até o fim da avaliação.

## 2.6 CONSIDERAÇÕES FINAIS

Nesta seção foi apresentado os principais conceitos necessários para um entendimento do modelo aqui implementado, o comitê de ensembles. Na seção 2.2 apresentamos uma introdução da aprendizagem supervisionada. Após isso, na seção 2.3 apresentamos diferentes técni-

cas de classificação o agrupamento dessas em famílias de classificadores, conforme características de aprendizagem comuns. Destacou-se como um tipo particular de família os meta-classificadores, que são classificadores que atuam com a saída de classificadores-base. É apresentado então na seção 2.4 o dilema bias vs variância e como os ensembles podem resolver em partes esse dilema. Na seção 2.5 são apresentadas as diversas características dos sistemas multi-classificadores, como a topologia, nomenclatura e diversidade. Por fim, na seção 2.6 são apresentadas as medidas de avaliação dos classificadores utilizadas na presente pesquisa. O capítulo a seguir apresentará o estado da arte com os trabalhos correlatos no que envolve comitês de ensembles.

# 3 ESTADO DA ARTE

Alternativas à otimização de classificadores, como *ensembles*, tem sido constantemente desenvolvidas (REN; ZHANG; SUGANTHAN, 2016). Os comitês de máquinas aproveitam-se do conhecimento de vários classificadores para promover uma melhor generalização (LEVESQUE; GAGNÉ; SABOURIN, 2016).

Tem sido mostrado formalmente e empiricamente que comitês de máquinas tem um poder de generalização melhor que classificadores-base (CHANDRA; YAO, 2006). Dietterich apresentou justificativas para eficiência dos *ensembles*, além de destacar que essa sub-área está entre os maiores focos de pesquisa da área de aprendizagem de máquina (DIETTERICH et al., 2000). Tumer e Ghosh apresentam uma prova formal da eficiência dos *ensembles* (TUMER; GHOSH, 1996).

Há diversos *surveys* sobre *ensembles* voltados para aprendizagem supervisionada (DIETTERICH et al., 2000)(POLIKAR, 2006) (BROWN et al., 2005)(ROKACH, 2009)(JUREK et al., 2014) na literatura, onde são resumidos diversos trabalhos, evoluções na área e sugestões de taxonomias para métodos *ensemble*. Entre os diversos métodos *ensemble* propostos, *stacking* tem sido um dos métodos mais representativos e democráticos, pois aceitam classificadores heterogêneos (de diferentes tipos) (TANG et al., 2010).

Na presente dissertação utilizou-se do Stacking para tomada de decisão final em um comitê de máquinas multi-nível heterogêneo baseado em diferentes paradigmas de aprendizagem. Para isso, investigou-se na literatura o que tem sido feito no decorrer dos anos. Dessa forma, é apresentado na seção atual uma revisão do estado da arte com trabalhos correlatos em relação a sistemas multi-classificadores, enfatizando *ensembles* com arquitetura multi-nível.

## 3.1 TRABALHOS CORRELATOS

A idéia do algoritmo *Stacking* foi introduzida por Wolpert (1992) no contexto de redes neurais (WOLPERT, 1992) e então generalizada por Breiman (1996) (BREIMAN, 1996). Ting e Witten (1999) usaram distribuições de probabilidades nas saídas de cada classe dos classificadores-base para entrada do meta-classificador (TING; WITTEN, 1999). Os autores propuseram usar como meta-algoritmo a técnica de Regressão Multi-Linear (MLR) .

Em testes empiricos, *Stacking* mostrou uma significativa perda de performance para bases de dados multi-classe. Para resolver esse problema, Seewald (2002) apresentou um algoritmo alternativo chamado *StackingC*. Baseado no *Stacking* com MLR, esse algoritmo reduz o número de probabilidades retornados pelos classificadores-base para superar a fraqueza do *stacking* em problemas multi-classe (SEEWALD, 2002).

Com a evolução do poder de processamento e memória dos computadores, a combinação de classificadores com um esquema multinível, indo além das duas camadas do *Stacking*, passou a ser utilizado com bastante frequência na literatura recente (ABAWAJY; KELAREV, 2012). Suas aplicações vêm sendo usados desde diagnósticos de doenças (TSIROGIANNIS et al., 2004) (KELAREV et al., 2012) (JELINEK et al., 2014), até segurança de dados em *Big Data* (ABAWAJY JEMAL H; ANDREI; CHOWDHURY, 2014) e e-mails (ABAWAJY et al., 2012). A ideia do seu funcionamento é de que no menor nível fiquem os classificadores-base (classificadores simples), cuja predições servirão como entrada para meta-classificadores do próximo nível. A decisão final é tomada na última camada pela integração das respostas dos classificadores do nível anterior (ABAWAJY; KELAREV, 2012).

Dettling e Bühlmann (2003) aplicaram uma combinação particular do *Bagging* e *Boosting* chamada *BagBoosting* em seus dados. A idéia foi usar o *bagging* como um módulo para o algoritmo de *boosting*, de modo que cada iteração do *boosting* não dependa apenas de um único classificador base, mas sim da agregação da saída de vários deles, gerada a partir de amostras bootstrap (DETTLING; BÜHLMANN, 2003).

A Figura 15 a seguir ilustra as referências dos trabalhos citados até então na forma de linha do tempo.

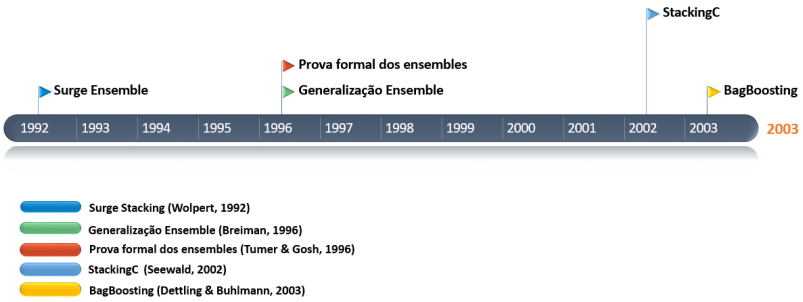


Figura 15 – Estado da arte - 1992 a 2008
Fonte: Do autor.

O trabalho de Su et al (2009) usa o comitê de *ensembles* para a tarefa de reconhecimento facial (SU et al., 2009). Usou-se uma ideia de extração de características locais, onde cada *ensemble* incorpora uma certa evidência facial entre as várias, e da extração de características global representada por um classificador. O fato dos classificadores-base possuírem informações discriminativas diversas entre eles ocasiona a eles certo grau de diversidade no erro. Finalmente, os classificadores locais e global são combinados por meio de uma soma ponderada para formar um classificador hierárquico de *ensembles*.

Kozegar testou diversos classificadores-base e *ensembles* simples para detecção de tumores em mamografias (KOZEGAR et al., 2013). Entretanto, o melhor desempenho adquirido foi utilizando um comitê de 3 *ensembles* : *RandomForest*, *Bagging*, e *AdaBoost*.

O trabalho de ABAWAJY et. al (2014) apresenta o algoritmo LIME (*Large Iterative Multitier Ensemble*) (ABAWAJY JEMAL H; ANDREI; CHOWDHURY, 2014) , especialmente projetado para aplicações relacionados à segurança da informação de *Big Data*. Utilizando uma arquitetura de 4 camadas, os classificadores LIME automatizam o processo de criação dos comitês de *ensembles* de forma iterativa.

O trabalho de Jelinek (2014) criou um *ensemble* com 3 níveis onde investigou-se as aplicações dos classificadores-base baseados em árvore de decisão (JELINEK et al., 2014). O algoritmo avaliou todas as possibilidades de pares dos 4 melhores desempenhos individuais dos meta-classificadores para a combinação do *ensemble* final.

Como a eficiência do algoritmo original do *Stacking* é diretamente dependente do número de classes sendo consideradas no problema (JUREK et al., 2014), um novo método chamada *Troika* foi pro-

posta por Menaheme et. al. (2009) para resolver especialmente problemas multi-classe (MENAHEM; ROKACH; ELOVICI, 2009). É baseada numa arquitetura de quatro camadas, onde a última camada contém somente um "super-classificador", que exibe um vetor de probabilidades como uma decisão final do ensemble (JUREK et al., 2014).

O trabalho de Ozay e Yarman-Vural (2016) propôs uma técnica de *ensemble* de duas camadas, chamada *Fuzzy Stacked Generalization* (FSG) (OZAY; YARMAN-VURAL, 2016), que estabelece uma arquitetura hierárquica baseado em aprendizagem *lazy* (na distância). Na camada base de um FSG, os classificadores Fuzzy k-NN recebem diferentes conjuntos de recursos, cada um dos quais é extraído do mesmo conjunto de dados para obter várias visualizações do conjunto de dados. Na meta-camada um espaço de fusão é construído pela agregação de espaços de decisão de todos os classificadores da camada base, como no *stacking* original.

O trabalho de ABAWAJY (2016) propõe o classificador multi-*ensemble* AIME (ABAWAJY et al., 2016) (automated iterative multi-tier *ensembles* ) . O AIME automatiza o processo de gerar um grande sistema multi-nivel de diferentes *ensembles* para combiná-los facilmente para um único esquema. Segundo autor, o método iterativo de geração automática e treinamento empregado pelos *ensembles* AIME não haviam sido considerados na literatura antes. Foram utilizados diversas técnicas de ensemble, entre elas *bagging*, *boosting*, *dagging* e *stacking*. Considerou como medida de avaliação apenas acurácia.

O trabalho de Yamazaki et al. (2017) relata o uso de uma arquitetura *ensemble* de 3 camadas numa tarefa de classificação de vandalismo na *WSDM Cup*. Diversas dificuldades tiveram de ser vencidas: tamanho do conjunto de dados, dados altamente desbalanceados e classificações próximo a tempo real. Além do pré-processamento, também foram realizadas atividades como extração de características antes do treinamento (Yamazaki et al., 2017).

A Figura 16 ilustra as referências das pesquisas citadas anteriormente sob uma linha do tempo no período de 2009 a 2017.

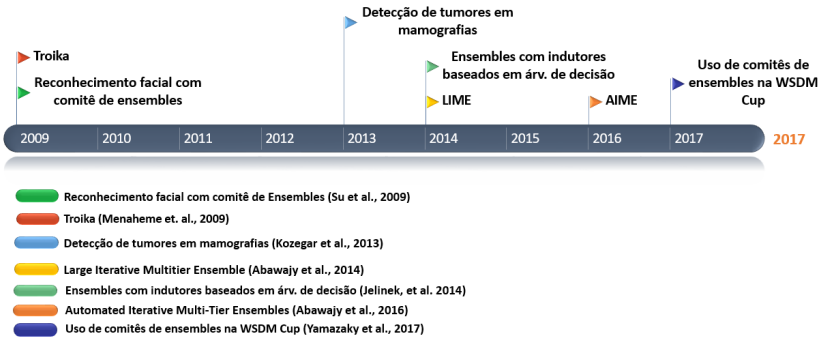


Figura 16 – Estado da arte - 2009 a 2017
Fonte: Do autor.

Cita-se ainda o livro de Dangeti (2017), onde o mesmo descreve modos de como criar comitês de ensemble, explicitando inclusive a implementação de algoritmos utilizando as linguagens R e Python (DANGETI, 2017).

## 3.2 CONSIDERAÇÕES FINAIS

A presente seção apresentou um estudo do estado da arte, realizado sobre os comitês de ensembles e suas atuais aplicações, que contemplou as experiências mais significativas registradas até o presente momento. Com o intuito de apresentar uma visão global dos acontecimentos, foram apresentadas as Figuras 15 e 16. O capítulo a seguir apresenta os procedimentos metodológicos que descrevem os passos e detalhes para a implementação do modelo proposto na presente dissertação.

# 4 PROCEDIMENTOS METODOLÓGICOS

Foi discutido na seção 2.4.2 a importância da diversidade para o sucesso dos comitês de máquina. Ao combinar diferentes algoritmos com erros não correlacionados, há uma grande chance de aumentar a capacidade de generalização (POLIKAR, 2006). De acordo com Chawla (2007), diversidade tem sido uma das principais preocupações no projeto de sistemas *ensemble* (CHAWLA; SYLVESTER, 2007).

Muitos sistemas de aprendizagem de máquina assumem que não é de grande relevância classificadores com diferentes meios de aprendizagem e, dessa forma, esses sistemas podem falhar em criar um classificador com uma maior visão global e, portanto, com maior capacidade de generalização. Para resolver isso, a proposta dessa dissertação é criar um modelo de comitês de ensembles com uma estratégia de diversidade baseado em paradigmas de aprendizagem. Por meio desse modelo, cada ensemble torna-se especializado, por meio dos classificadores-base, em uma família de classificadores. Dessa forma, as saídas dos classificadores-base (nível 0) de uma certa família servem como entrada pro nível imediatamente superior (nível 1). O mesmo ocorre do nível 1 para o 2 na combinação dos ensembles.

Após análise preliminar de famílias em diversas bases de dados, optou-se pela utilização de famílias com aprendizagem baseada em regras, famílias *lazy*, funções e bayesianas. O número de níveis utilizados no modelo foi o mínimo necessário para a realização a criação e especialização dos ensembles em famílias, ou seja, foram utilizados 3 níveis (níveis 0, 1 e 2). Acrescentar algum nível além desses tornaria o modelo proposto demasiadamente complexo em termos de tempo de execução pra ser utilizado com todas as famílias propostas.

Apresenta-se nessa seção os procedimentos metodológicos usados para a geração dos resultados obtidos, incluindo as etapas do modelo e configurações dos experimentos.

## 4.1 ETAPAS DO MODELO

O modelo de comitê de ensembles proposto é dividido em etapas. Num primeiro momento, após aquisição de determinada base de dados no repositório da *UCI* (LICHMAN, 2013) , os dados são adaptados para um arquivo .ARFF, exigência para uso da biblioteca *WEKA*. Posteriormente passam por um pré-processamento, onde são removi-

dos dados duplicados, dados faltantes são preenchidos com valores de média e moda, assim como os outliers são revistos pela *API WEKA*, versão 3.7. Após isso, é efetivamente realizado o comitê de *ensembles*, no qual realiza a tarefa de classificação. A Figura 17 ilustra os passos descritos.

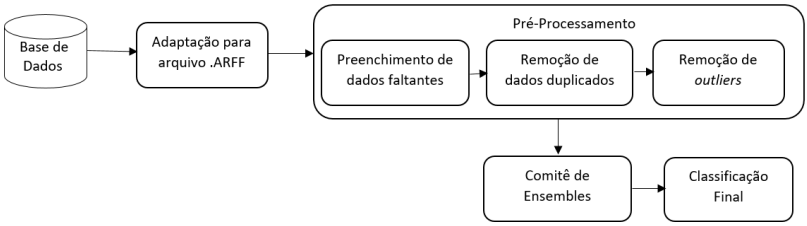


Figura 17 – Etapas do Modelo
Fonte: Do autor.

## 4.2 ALGORITMO DO COMITÊ DE *ENSEMBLES*

Essa subseção tem como objetivo a apresentação do algoritmo de comitê de *ensembles* desenvolvido. Diferente do *stacking*, o algoritmo proposto consiste em adicionar uma camada intermediária de aprendizagem entre os classificadores-base e o decisor final, usando como estratégia de diversidade o treinamento de *ensembles* especializados, cada qual em um diferente paradigma de aprendizagem (famílias), já descritos anteriormente na seção 2. Assim como o *Stacking*, a presente pesquisa usou em sua metodologia uma topologia paralela.

O conceito de "especialização" do *ensemble* se justifica pelo fato de que os classificadores-base - que constituem seus dados - são exclusivamente de uma família específica. A ideia é que os *ensembles* especializados da camada intermediária combinem as saídas dos classificadores base. A partir disso é gerado uma nova saída que sera analisada pelo *ensemble* da camada superior que toma a decisão final por todo o sistema de classificação multi-camada. As possibilidades de meta - classificadores usados no nível 1 são descritos com detalhes na seção 4. Na pesquisa em questão, classificadores homogêneos são formados por um classificador-base principal que também são usados nos classificadores heterogêneos em par com um classificador secundário da mesma família. A Figura 18 ilustra o processo de decisão.

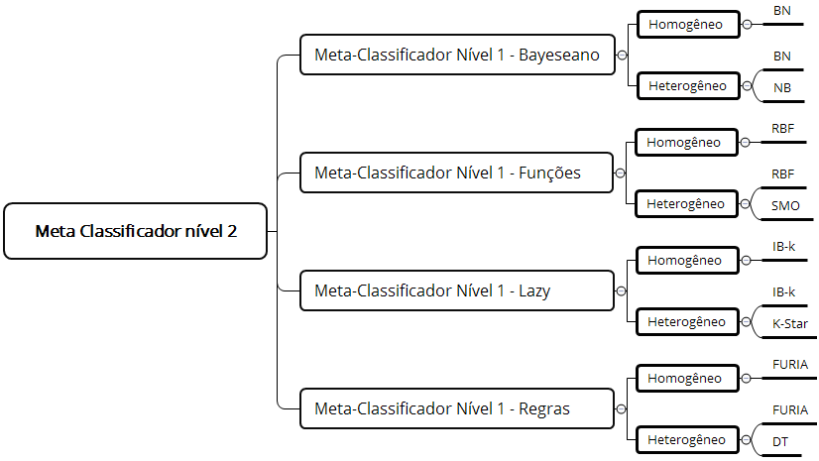


Figura 18 – Modelo proposto
Fonte: Do autor.

Após criação individual de cada classificador, os níveis são treinados e avaliados com determinada base de dados. O desempenho de cada nível é calculado pela média aritmética de todos os classificadores do nível examinado tanto em termos de acurácia, quanto área sob a curva ROC e quanto o tempo de execução.

Para ilustrar de forma detalhada o modelo proposto na presente pesquisa, desenvolveu-se conforme a Figura 19, um diagrama de constituição do comitê de Ensembles.

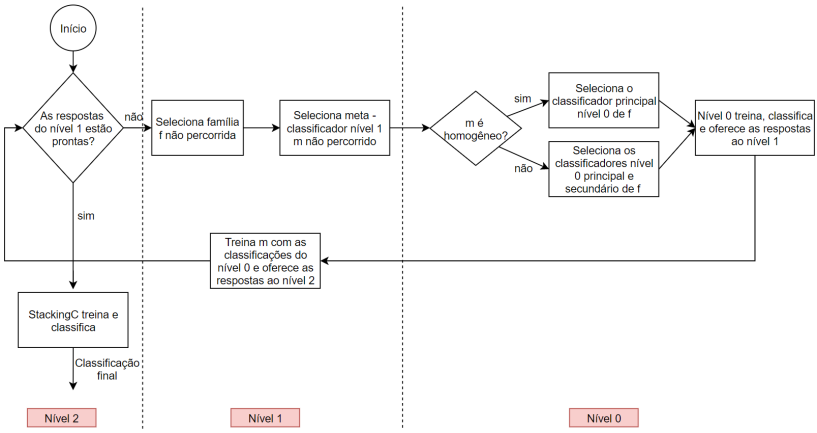


Figura 19 – Diagrama do Modelo Proposto.
Fonte: Do autor.

Inicialmente verifica-se as respostas do nível 1. O ciclo, após o inicío, dura enquanto as saídas de todas as famílias não estiverem prontas. Em caso negativo é selecionada uma das quatro famílias não percorridas f, para em seguida selecionar um dos meta-classificadores m do nível 1. Verifica-se então se m é homogêneo ou heterogêneo. No primeiro caso, escolhe-se o classificador principal nível 0 de f ou no caso de ser heterogêneo seleciona-se os classificadores principais e secundários de f. O nível 0 é então treinado com os classificadores a ele designados, oferece então as respostas ao nível 1, que por sua vez, com essas saídas consegue treinar m que por fim fornece sua resposta ao nível 2. O nível 2 se encerra após todas as famílias serem percorridas e o algoritmo StackingC treinar com as respostas do nível 1.

A avaliação ocorre junto ao treinamento por meio de validação cruzada de 10 pastas. Para que todos os classificadores nível 1 possuam as mesmas condições, são treinados todas as possibilidades de configuração no nível 0 e no nível 1.

## 4.3 ESTUDO EXPERIMENTAL

Esta seção aborda os experimentos realizados. Inicialmente, são apresentadas as bases de dados e as configurações utilizadas nos experimentos, e também como foram obtidos os resultados para análise.

Para realização dos experimentos foi utilizada a *API Weka* (WIT-

TEN et al., 2016). Esta ferramenta serviu de base para as atividades de treinamento, classificação e validação dos classificadores utilizados.

A versão utilizada da *API* foi 3.7, sendo que a presente pesquisa limitou-se em utilizar os valores padrões do *Weka* nos parâmetros não configurados previamente. Considerando que são muitos os parâmetros possíveis em cada classificador/arquitetura, poderiam ser utilizados métodos de otimização para a parametrização, porém esse não foi o foco da presente dissertação.

A implementação foi desenvolvida na plataforma *Java SE*. A opção pela plataforma Java se justifica por motivos de portabilidade (Windows/Linux), gratuidade, familiaridade com a linguagem, boa documentação e fácil comunicação com o *Weka*. As bases de dados, os classificadores usados e outros detalhes são descritos nas subseções a seguir.

### 4.3.1 Bases de dados

Para os experimentos da presente dissertação foram escolhidas sete diferentes bases de dados do repositório público *UCI* (LICHMAN, 2013) que apresentaram entre si variações no número de atributos, instâncias e diferentes números de classes na variável dependente (binária/multi-classe). A Tabela 1 apresenta as propriedades mais importantes das bases de dados selecionadas.

Tabela 1 – Bases de dados utilizadas

| Bases de dados | Abreviação | Número de atributos | Instâncias | Número de classes na saída |
|---|---|---|---|---|
| *Wine* | WI | 13 | 178 | 3 |
| *Glass Identification* | GI | 13 | 214 | 6 |
| *Ecoli* | EC | 8 | 336 | 8 |
| *Breast Cancer* | BC | 9 | 286 | 2 |
| *Diabetes* | DB | 8 | 768 | 2 |
| *Horse Colic* | HC | 27 | 368 | 3 |
| *Ionosphere* | IS | 34 | 351 | 2 |

Fonte: Do autor.

As bases com número de classes na saída maior que dois são consideradas bases de dados multi-classe. A seguir, uma descrição breve de cada uma das base de dados utilizadas.

#### 4.3.1.a Wine

Sendo amplamente usada em problemas de mineração de dados, a base de dados Wine é utilizada para determinar a origem de três tipos de vinhos italianos a partir da análise química (CORTEZ et al., 2009). A amostra de treinamento contêm 178 instâncias com medições de 13 constituintes químicos. É considerada uma base relativamente balanceada, de forma que o tamanho das amostras entre as três classes são distribuídas nos seguintes números: 59, 71 e 48. Apresenta ainda um conjunto de dados com classes facilmente separáveis (CORTEZ et al., 2009), o que pode implicar em alta acurácia diante a maioria dos classificadores.

#### 4.3.1.b Glass Identification

A base de dados Glass Identification tem como objetivo identificar amostras de vidro a partir de suas informações químicas, sendo sua divulgação motivada para investigação criminal (SURESH; DONG; KIM, 2010). Nove características são usadas para identificar sete tipos de vidros (embora que para um desses não haja nenhuma instancia representante). Tem como características dados altamente desbalanceados, com 214 instâncias no total, sendo distribuidas da seguinte forma entre as classes: 35, 38, 9, 7, 5 e 15. O pequeno tamanho da amostra de treinamento junto a sobreposição entre algumas classes aumentam ainda mais a complexidade dessa base (SURESH; DONG; KIM, 2010).

#### 4.3.1.c Ecoli

Para funcionar adequadamente, as proteínas devem ser transportadas para dentro da célula. Por outro lado, a localização celular de uma proteína afeta sua potencial funcionalidade, bem como sua acessibilidade aos tratamentos com drogas. Geralmente a informação necessaria para prever a localização correta da proteína geralmente é encontrado em sua própria sequencia (HORTON; NAKAI, 1997). A base de dados Ecoli possui 336 sequencias de proteínas rotuladas de acordo com 8 classes, distribuídas nos seguintes números por classe: 143, 77, 52, 35, 20, 5, 2, 2.

#### 4.3.1.d Breast Cancer

Segundo a OMS o cancêr de mama é a causa mais frequente de morte por câncer em mulheres (ORGANIZATION et al., 2012). Ao mesmo tempo, ele está entre os tipos de cancer mais curaveis se diagnosticado precocemente. Diante a necessidade do diagnóstico antecipado, trabalhos com mineração de dados pra previsão da doença tem sido muito utilizados. A presente base de dados Breast Cancer trabalha para análise de diagnóstico de recorrência ou não do cancêr de mama (AKAY, 2009). Contém 9 atributos - como idade, tamanho do tumor, localização da mama com suspeita de nodo, entre outras - de 286 pacientes, de forma que entre esses, 201 são casos não recorrentes e 85 são recorrentes.

#### 4.3.1.e Diabetes

Diabetes melito é uma desordem metabólica complexa caracterizada por uma hiperglicemia crônica decorrente de defeitos na secreção e/ou ação da insulina (FERREIRA et al., 2011). Existem dois tipos principais de diabetes identificados: o tipo 1 (dependente de insulina) e tipo 2 (não dependente de insulina) (ASSOCIATION et al., 2010). A base de dados Diabetes consiste de amostras originadas de uma comunidade indígena de Pimas, onde foi reportado maior indice de diabetes no mundo (KNOWLER et al., 1979). Essas amostras referem-se aos diagnósticos de casos positivos (268 instâncias) e negativos (500 instâncias) de diabetes tipo 2, descritas por 8 características, como idade e número de gestações.

#### 4.3.1.f Horse Colic

A cólica é uma das principais causas de morbidade e mortalidade em populações de cavalos (MUÑOZ; GONZÁLEZ, 2014). Diante disso a base de dados Horse Colic possui um quadro clínico de 368 cavalos que passaram por isso. É comumente utilizado na mineração de dados pela alta heterogeneidade dos dados além dos vários valores nulos (30% dos valores) (MUÑOZ; GONZÁLEZ, 2014). A base de dados, bem como seus 28 atributos são bem documentados no repositório da UCI. Há diversas possibilidades de tarefas que podem ser atribuidas para esse conjunto de dados, as mais comuns são predições do atributo 23 (“O que acontece com o cavalo?”) e do 24 (“Era caracterizado uma lesão cirúrgica?”). Na

presente dissertação investiga-se o atributo 23, que prediz o que acontece com o cavalo baseado em suas condições médicas passadas. São três possibilidades de classe nesse atributo: cavalo sobreviveu, morreu ou foi eutanaziado, distribuídos com, respectivamente, 225, 89 e 52 exemplos em cada classe.

#### 4.3.1.g Ionosfera

A presente base de dados consiste de informações coletadas de um radar em relação aos elétrons livres na ionosfera na cidade de Goose Bay, Canadá (SKURICHINA; DUIN, 2005). Os retornos "bons"do radar são aqueles mostrando evidências de algum tipo de estrutura na ionosfera e os retornos "ruins"são aqueles que não retornam nada, ou seja, seus sinais passam pela ionosfera (SKURICHINA; DUIN, 2005). O conjunto de dados são descritos por 34 atributos, com uma amostra de 351 instâncias, divididas entre 225 instância da classe "bom"e 126 da classe "ruim".

### 4.3.2 Classificadores e Configurações dos Experimentos

Para aplicação do modelo das bases comentado na seção 4.3, é necessário estabelecer padrões de configuração entre os classificadores, medidas de avaliação e regras para aplicação do mesmo. A presente subseção detalha as técnicas e configurações gerais do modelo desenvolvido na presente dissertação.

Várias configurações padrão foram utilizadas durante os experimentos. As bases de dados foram avaliadas nos experimentos em termos de acurácia, área sob a curva ROC ($AUC$) e tempo de treinamento. O desempenho foi medido pela técnica de validação cruzada de 10 pastas. Os classificadores, bem como um resumo das configurações aplicadas, podem ser verificados no Quadro 2.

Quadro 2 – Configurações dos experimentos

| | |
|---|---|
| **Avaliação de Desempenho** | *10-fold cross-validation* |
| **Meta-classificadores do nível 1** | *Bagging, Boosting, Dagging, MultiScheme* |
| **Tamanho do pool de classificadores dos *ensembles* homogêneos do nível 1** | 10 |
| **Famílias dos *ensembles* nível 1** | *Regras, Lazy, Funções, Bayes* |
| **Meta-classificador do nível 2** | *StackingC com NaïveBayes como meta-classificador* |
| **Classificadores-Base** | *NB, BN, RBF, SMO, IBk, K-Star, FURIA, DT* |

Fonte: Do autor.

Para cada família, há dois classificadores no nível 0 a serem utilizados (classificadores-base). O Quadro 3 divide os classificadores usados em cada paradigma como classificador principal ou classificador secundário. O classificador principal é aquele a ser utilizado pelos meta-classificadores homogêneos em determinada família. O classificador secundário é utilizado em meta-classificadores heterogêneos da família em questão e também como parte do desempenho total do nível 0. A razão da utilização de apenas dois classificadores nível 0 em meta-classificadores heterogêneos se deve ao tempo de execução. Como é considerado nos resultados todas as combinações possíveis entre todos os classificadores nível 2, 1 e 0, acrescentar mais classificadores nível 0 seria inviável.

**Quadro 3** – Classificadores-base utilizados

| **Família** | **Classificador Principal** | **Classificador Secundário** |
|---|---|---|
| **Bayesianos** | *BayesNet* | *Naïve Bayes* |
| **Funções** | *RBF* | *SMO* |
| **Lazy** | *IBk* | *K-Star* |
| **Regras** | *FURIA* | *Decision Table* |

Fonte: Do autor.

O nível 1, por sua vez, divide seus meta-classificadores em homogêneos e heterogêneos. Nos homogêneos foram utilizados os algoritmos *Bagging*, *AdaBoosting*, *Dagging* e como heterogêneo o *MultiScheme*. Cada um dos 4 meta-classificadores desse nível ficou responsável por se especializar por uma família. Para a geração dos resultados desse nível foram utilizadas todas as combinações possíveis de meta-classificadores do nível 1 com cada uma das famílias, dessa forma todos os paradigmas foram igualmente distribuídos entre os meta-classificadores do nível 1.

Para o nível 2 foi utilizado o meta-classificador heterogêneo *StackingC* com o classificador Naïve Bayes (escolhido por ser um classificador tipicamente ágil). A Figura 20 resume os classificadores utilizados em cada nível.

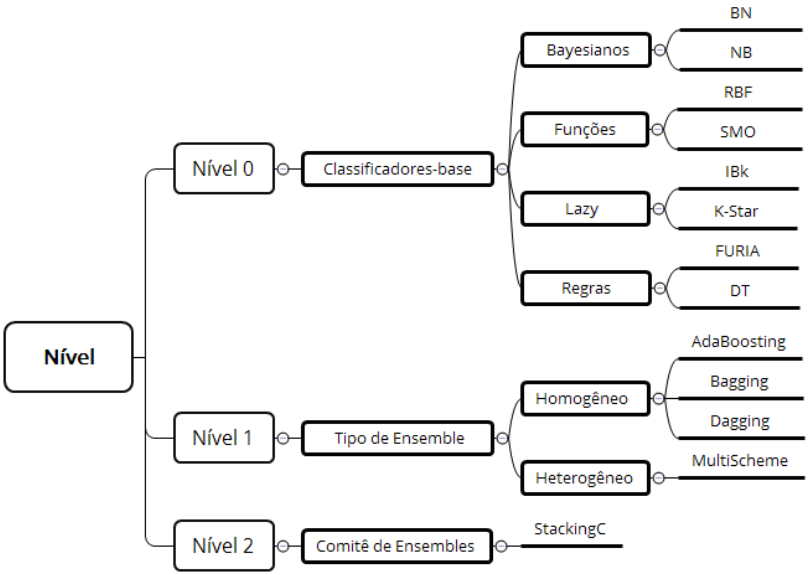


Figura 20 – Classificadores utilizados em cada nível
Fonte: Do autor.

## 4.4 ANÁLISE ESTATÍSTICA

Foi determinado na presente pesquisa que é necessário que os indicadores de avaliação não deveriam depender da distribuição dos dados em qualquer tipo de base de dados. Para isso, indo além da tradicional taxa de acerto (ou acurácia), optou-se pela utilização da área sob a curva ROC (mais detalhes sobre $AUC$, verificar subseção 2.5) e também do tempo gasto para classificação das amostras. Dessa forma, decidiu-se que as medidas de desempenho (variáveis dependentes) que vão determinar a qualidade de um classificador na presente pesquisa são: (1) acurácia, (2) área sobre a curva ROC ($AUC$) e (3) tempo de treinamento. Sendo assim, os experimentos foram conduzidos de maneira que pudessem analisar os seguintes pontos para cada variável dependente:

1. Comparação do nível 2 com os *ensembles* que o formam (nível 1).

2. Comparação geral entre os níveis.

A partir disso, os dados foram analisados quanto à sua normalidade pelo teste de Kolmogorov-Smirnov, como houve normalidade dos dados foi utilizado a análise de variância (ANOVA) de um fator para análise de médias com pós-teste de Tukey. Entretanto, como na variável tempo de treinamento não houve normalidade dos dados foi utilizado o teste de Kruskal-Wallis.

Dessa maneira, na análise 1, para cada base de dados foi comparado os resultados de cada *ensemble* (especializado em certa família) com o nível 2. Na análise 2, foram desconsiderados os paradigmas utilizados, e os três níveis foram analisados separadamente por cada base de dados. Sendo assim, os resultados foram expressos em tabelas e gráficos e em todas as análises estatísticas utilizadas foi empregado um nível de significância de 5% ($\alpha = 0,05$).

## 4.5 CONSIDERAÇÕES FINAIS

Nesta seção foi descrito o modelo de comitê de ensembles proposto na presente pesquisa. Na seção 4.2 foi apresentado as etapas realizadas antes da execução do modelo. Em seguida, na seção 4.3, é apresentado o os passos para execução do comitê de ensembles. Na seção 4.4 é descrito os experimentos utilizados, onde é apresentado as bases de dados e as configurações dos classificadores utilizados para geração dos resultados. Por fim na seção 4.5 é apresentado como foi realizado a análise estatística do modelo. A seção a seguir é apresentado os resultados e as discussões da presente pesquisa.

# 5 RESULTADOS E DISCUSSÕES

Os resultados discutidos nesta seção foram obtidos seguindo criteriosamente os procedimentos metodológicos, experimentos e análise estatística descritas na seção 4.

Grande parte dos trabalhos apresentados no estado da arte usaram em suas pesquisas bases particulares ou, quando públicas, bases diferentes daquelas aqui utilizadas. Por essa razão, a presente seção busca apresentar a comparação de desempenho especialmente entre o modelo proposto e os ensembles tradicionais (nível 1) e os classificadores comuns (nível 0).

A presente seção foi organizadas três sub-seções, uma para cada medida de avaliação. Ambas as sub-seções ainda contam com dois tópicos principais: um que analisa a medida de desempenho entre os níveis e outro que analisa comparação do nível 2 com os *ensembles* do nível 1.

## 5.1 ACURÁCIA

A Tabela 2 apresenta dados médios de acurácia de todos os níveis por paradigma em cada base de dados. Os valores mostrados para os níveis 0 e 1 são uma média da acurácia dos classificadores-base/ensembles de determinado paradigma em cada base. A última coluna da tabela refere-se a significância estatística que envolve comparações das médias dos níveis 1 e 2. Como no nível 0 existem apenas dois classificadores por família, este foi desconsiderado na análise estatística.

No geral o nível 2 apresentou um desempenho médio melhor comparados aos níveis 0 e 1. Estatisticamente, o nível 2 foi superior em todos os paradigmas do nível 1 de todas as bases de dados, com exceção da base WINE que apresentou superioridade estatística apenas no paradigma Regras.

Tabela 2 – Média da acurácia de todos os níveis por paradigma em cada base de dados

| | Média de acurácia dos níveis, segundo as famílias dos classificadores | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Bayes | | Funções | | Lazy | | Regras | | Sign. Estatística (*) |
| **Bases de Dados** | **nível 2** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | |
| *WI* | 0,9756 | 0,9829 | 0,9700 | 0,9716 | 0,9732 | 0,9630 | 0,9716 | 0,9062 | 0,9432 | Superior somente a regras |
| *GI* | 0,7973 | 0,6399 | 0,6617 | 0,6658 | 0,6565 | 0,8083 | 0,7565 | 0,7124 | 0,7062 | Superior a todos |
| *EC* | 0,9334 | 0,9334 | 0,8973 | 0,8387 | 0,8548 | 0,9193 | 0,9 | 0,8887 | 0,8816 | Superior a todos |
| *BC* | 0,9987 | 0,9684 | 0,975 | 0,9903 | 0,9681 | 0,9660 | 0,9715 | 1,0000 | 0,984 | Superior a todos |
| *DB* | 0,9203 | 0,8939 | 0,8941 | 0,8924 | 0,8795 | 0,8475 | 0,8558 | 0,9249 | 0,9084 | Superior a todos |
| *HC* | 0,723 | 0,703 | 0,6979 | 0,6886 | 0,6877 | 0,6708 | 0,6772 | 0,664 | 0,6717 | Superior a todos |
| *IS* | 0,9361 | 0,8689 | 0,888 | 0,9164 | 0,8987 | 0,9838 | 0,858 | 0,8977 | 0,8975 | Superior a todos |
| **Média Aritmética** | **0,8978** | **0,8558** | **0,8549** | **0,852** | **0,8455** | **0,8798** | **0,8558** | **0,8563** | **0,8561** | |
| *Nível 2 significativamente melhor que nível 1 ($\alpha = 0,05$) | | | | | | | | | | |

Fonte: Do autor.

### 5.1.1 Desempenho entre os níveis

Com intuito de mostrar a eficiência do nível 2 em relação aos demais níveis, em termos de acurácia, é apresentado o gráfico da Figura 21. Enquanto que o nível 2 se distância dos demais níveis, observa-se desempenho semelhantes entre os níveis 0 e 1, embora o nível 0 tenha certa prevalência. O desempenho ligeiramente pior do nível 1 se deve provavelmente a baixa diversidade entre os classificadores-base semelhantes que o formam.

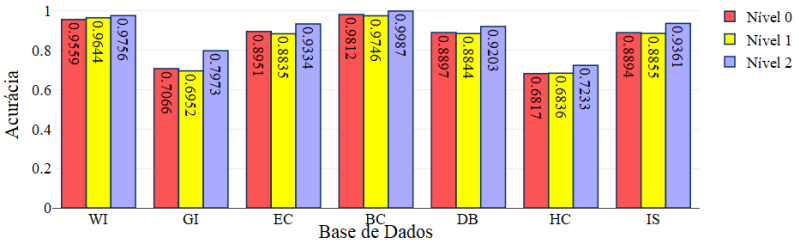


Figura 21 – Desempenho médio da acurácia dos níveis por base de dados.

Fonte: Do autor.

Nota-se, conforme já sugerido em pesquisas prévias (LORBIESKI; NASSAR, 2018b), o nível 2 teve uma tendência de promover uma otimização ainda maior na acurácia para base de dados multi-classe onde não houve grande separabilidade dos dados, como nas bases EC, HC e GI. Em particular para a base de dados GI, a acurácia do nível 2 alcançou

um desempenho 14% superior em relação ao nível 1. A acurácia média geral foi de 86,1%, 85,31% e 89,78% para os respectivos níveis 0, 1 e 2.

### 5.1.2 Comparação dos paradigmas com nível 2

O gráfico da Figura 22 mostra o desempenho do nível 2 e dos *ensembles* que o formam. Em todos os casos, como se confirma na Tabela 2, o nível 2 foi superior em termos de acurácia. No geral houve certa proximidade de desempenho entre as famílias.

Classificadores cujo tamanho da amostra por classe não causam grande impacto para boa aprendizagem (como a família bayesiana (MOTTA et al., 2016)), tiveram um desempenho médio superior aos demais. No geral, famílias de classificadores bayesianos e baseados em regras apresentam uma resistência a ruídos (LETHAM et al., 2015), causando resultados mais estáveis, de forma que não há grandes picos de desempenho no gráfico (HÄMÄLÄINEN; VINNI, 2010).

No que diz respeito a família *lazy* na base de dados HC e IS, a queda de desempenho possivelmente também se deve ao grande número de atributos da base de dados em questão, o que condiz com resultados da literatura (HU et al., 2016), onde a função de distância euclidiana para muitos atributos teve redução de desempenho. A base de dados *Glass Identification* teve o melhor desempenho com a familia *lazy* no nível 1, o que também vai de acordo com os resultados da literatura (ALDAYEL, 2012).

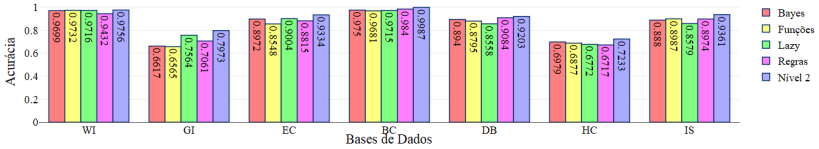


Figura 22 – Desempenho do nível 2 em relação às famílias do nível 1 em termos de acurácia.

Fonte: Do autor.

## 5.2 ÁREA SOB A CURVA ROC

Sendo particularmente útil quando há uma desproporção entre as classes (PRATI; BATISTA; MONARD, 2008), a curva ROC vem sendo muito utilizada como medida de avaliação. A Tabela 3 apresenta dados médios da $AUC$, de todos os níveis por paradigma em cada base de dados. De forma semelhante a Tabela 2, a última coluna refere-se a significância estatística que envolve comparações das médias dos níveis 1 e 2. O nível 0 também foi desconsiderado da análise estatística. A Tabela 3 mostra como o nível 2 foi estatisticamente superior ao nível 1 em todas as bases, com exceção da base Wine - onde houve significativa superioridade do nível 2 apenas em relação à família regras.

Tabela 3 – Média da área sob a curva ROC de todos os níveis por paradigma em cada base de dados

| | Média de AUC dos níveis, segundo as famílias dos classificadores | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | **Bayes** | | **Funções** | | **Lazy** | | **Regras** | | **Sign. Estatística (*)** |
| **Bases de Dados** | **nível 2** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | |
| *WI* | 0,995 | 0,9990 | 0,996 | 0,9910 | 0,993 | 0,9857 | 0,992 | 0,9708 | 0,988 | Superior somente a regras |
| *GI* | 0,937 | 0,8547 | 0,839 | 0,8165 | 0,820 | 0,9036 | 0,897 | 0,8506 | 0,868 | Superior a todos |
| *EC* | 0,993 | 0,9928 | 0,987 | 0,9650 | 0,974 | 0,9715 | 0,979 | 0,9702 | 0,981 | Superior a todos |
| *BC* | 1,000 | 0,9909 | 0,992 | 0,9992 | 0,995 | 0,9880 | 0,995 | 1,0000 | 0,980 | Superior a todos |
| *DB* | 0,968 | 0,9480 | 0,948 | 0,9021 | 0,911 | 0,8781 | 0,899 | 0,9467 | 0,947 | Superior a todos |
| *HC* | 0,843 | 0,8389 | 0,826 | 0,7859 | 0,795 | 0,7316 | 0,762 | 0,7244 | 0,747 | Superior a todos |
| *IS* | 0,978 | 0,9507 | 0,954 | 0,9182 | 0,934 | 0,8883 | 0,914 | 0,9036 | 0,950 | Superior a todos |
| **Média Aritmética** | **0,9591** | **0,9392** | **0,9346** | **0,9111** | **0,9174** | **0,9067** | **0,9197** | **0,9095** | **0,923** | |
| *Nível 2 significativamente melhor que nível 1 ($\alpha = 0,05$) | | | | | | | | | | |

Fonte: Do autor.

### 5.2.1 Desempenho da $AUC$ entre os níveis

O gráfico da Figura 23, apresenta os resultados da $AUC$ das classificações, com relação a eficiência do nível 2 comparado aos níveis 0 e 1. Observa-se que o nível 2 apresenta um desempenho maior em relação aos demais níveis. Além disso, se obteve alternância de desempenho entre os níveis 0 e 1.

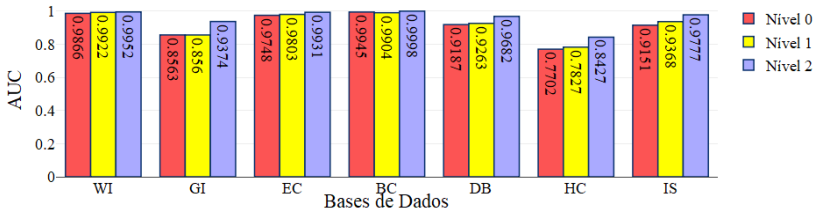


Figura 23 – Desempenho médio da área sob a curva ROC dos níveis por base de dados.

Fonte: Do autor.

O melhor desempenho do nível 2 frente aos demais níveis foi na base GI, com um ganho de performance médio de 10%, em relações aos níveis 0 e 1. A $AUC$ média geral foi 91,66%, 92,36% e 95,91% para os respectivos níveis 0, 1 e 2.

### 5.2.2 Comparação da *AUC* dos *ensembles* com o nível 2

O gráfico da Figura 24 mostra o desempenho do nível 2 e dos *ensembles* que o formam. Os resultados desta análise são semelhantes ao gráfico da acurácia na Figura 22. Dessa forma, o nível 2 teve superioridade em relação aos demais níveis, os *ensembles* também apresentaram comportamento diversificado, embora a família Bayesiana tenha mostrado um desempenho melhor.

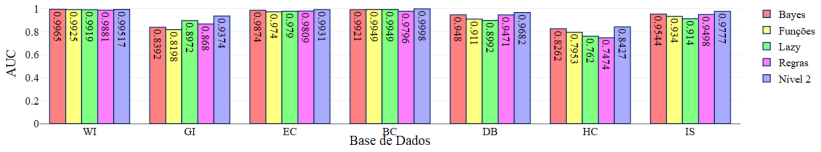


Figura 24 – Desempenho da $AUC$ no nível 2 em relação aos paradigmas do nível 1.

Fonte: Do autor.

## 5.3 TEMPO

Medir o tempo de execução pode ser muito importante, uma vez que uma espera demasiada pode tornar qualquer problema de classificação inviável. A Tabela 4 apresenta os dados de tempo por nível e base de dados. O tempo, aqui mensurado em segundos, foi calculado pelo tempo de execução gasto em cada base de dados. O desempenho de tempo no nível 2, como esperado, foi muito maior que o do nível 1 (ou nível 0), uma vez que o treinamento do nível 1 ou 2 depende dos resultados do nível imediatamente abaixo.

Tabela 4 – Média de tempo, em segundos, de todos os níveis por paradigma em cada base de dados

| | Média de tempo dos níveis, em segundos, de acordo com as famílias dos classificadores | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | **Bayes** | | **Funções** | | **Lazy** | | **Regras** | | **Sign. Estatística (* )** |
| **Bases de Dados** | **nível 2** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | **nível 0** | **nível 1** | |
| *WI* | 66,927 | 0,222 | 0,2593 | 0,2625 | 2,1733 | 0,2625 | 6,3011 | 0,275 | 0,5657 | Superior a todos |
| *GI* | 70,022 | 0,031 | 0,127 | 0,6045 | 2,8186 | 0,167 | 4,6534 | 0,1955 | 0,5677 | Superior a todos |
| *EC* | 93,284 | 0,018 | 0,2301 | 0,6375 | 4,7239 | 0,2405 | 5,8624 | 0,166 | 0,7519 | Superior a todos |
| *BC* | 22,299 | 0,0095 | 0,0774 | 0,295 | 1,31 | 0,0385 | 1,31 | 0,049 | 0,1941 | Superior a todos |
| *DB* | 301,104 | 0,0245 | 0,186 | 0,4275 | 4,491 | 1,3105 | 33,217 | 0,422 | 1,5033 | Superior a todos |
| *HC* | 305,05 | 0,211 | 0,2924 | 69,84 | 373,92 | 0,4175 | 9,3536 | 0,9175 | 3,3321 | Superior a todos, exceto em funções |
| *IS* | 641,498 | 0,0325 | 0,4077 | 0,0468 | 8,627 | 2,2875 | 70,0617 | 0,494 | 2,323 | Superior a todos |
| **Média Aritmética** | **214,31** | **0,0784** | **0,2257** | **10,302** | **56,867** | **0,6749** | **18,6805** | **0,3599** | **1,3197** | |
| *Nível 2 com tempo significativamente superior que nível 1 ($\alpha = 0,05$) | | | | | | | | | | |

Fonte: Do autor.

De uma forma estatística, pode-se dizer que o nível 2 apresentou quase sempre um tempo de execução significativamente superior ao nível 1 (com exceção da família funções na base HC). Observa-se ainda na Tabela 4 que o paradigma *lazy* quase sempre foi pior que todas as demais famílias do mesmo nível. A base de de dados HC causou uma queda de desempenho muito grande em termos de tempo na família baseada em funções. Aparentemente, propriedades da base HC (como a grande quantidade de valores únicos que alguns atributos tem) influenciaram bastante nos resultados de tempo de execução. Resultados gráficos podem ser vistos na Figura 26.

### 5.3.1 Desempenho do tempo entre os níveis

Para efeitos de comparação do tempo de execução médio gasto entre os níveis, pode-se observar o gráfico da Figura 25. Os resultados exibidos na Tabela 4 nos ajudam a entender seu comportamento. Embora a base de dados HC tenha sido a que gastou mais tempo entre as demais bases no nível 1, não foi o que o ocorreu no nível 2. A base de

dados IS ainda foi o que gastou maior tempo, ultrapassando a marca média de 600 segundos.

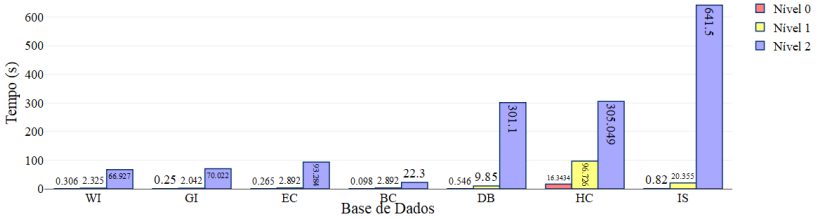


Figura 25 – Desempenho médio do tempo, em segundos, dos níveis por bases de dados.

Fonte: Do autor.

### 5.3.2 Comparação *ensembles* com nível 2

Os gráficos de tempo por família podem ser vistos na Figura 26. Nota-se um tempo de treinamento superior nas famílias funções e *lazy* em relação às demais. Isso porque, embora simples e eficazes, classificadores *lazy* são eficazes durante a fase de treinamento, mas lento em relação aos demais durante processo de classificação (KOTSIANTIS; ZAHARAKIS; PINTELAS, 2007). A grande quantidade de valores únicos que alguns atributos têm na base de dados HC podem ter influenciado nos resultados de tempo de execução na família baseada em funções.

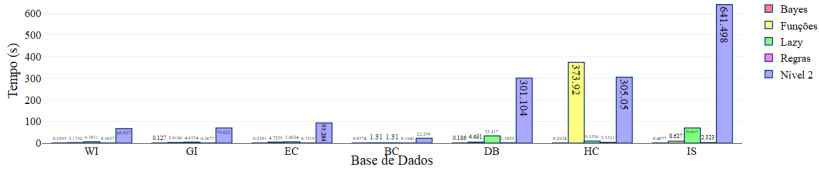


Figura 26 – Desempenho do tempo no nível 2, em segundos, em relação aos paradigmas do nível 1.

Fonte: Do autor.

Os tempos gastos no nível 1 da família bayesiana foram muito baixos, onde o tempo médio de execução dos ensembles dessa família foram aproximadamente 949 vezes mais rápida do que o nível 2, de

forma que o tempo médio geral de execução nessa família foi de apenas 0,2 segundos. Família *lazy* e de funções como dito anteriormente foram os mais lentos. O nível 2 foi pouco mais que 3,7 vezes mais lento que o ensemble baseado em funções e 11 vezes mais lento do que a família *lazy*. Os ensembles baseados em regras tiveram um desempenho 162 vezes mais rápido que o nível 2. O tempo médio geral de execução do nível 2 foi de 214 segundos.

# 6 CONSIDERAÇÕES FINAIS

Os estudos realizados neste trabalho buscaram criar um modelo que combina *ensembles* especializados para uma tomada de decisão final. A ideia fundamental foi criar *ensembles* que que se aperfeiçoaram em diferentes paradigmas de aprendizagem, visando aumentar a diversidade do comitê como um todo, de maneira a se obter ganhos no desempenho preditivo em termos de acurácia e também da área sob a curva ROC. Essa seção tem como objetivo abordar as conclusões, limitações e os trabalhos futuros.

## 6.1 CONCLUSÕES

Diferentes técnicas de combinação foram avaliadas. A família das árvores de decisão, bem como o uso do Stacking no nível 1, já haviam sido testadas anteriormente e não haviam mostrado inicialmente bons resultados (LORBIESKI; NASSAR, 2018a), motivo pelo qual não foi utilizado aqui.

Foi verificado a eficácia do comitê de *ensembles* em termos de acurácia e área sob a curva ROC, além do tempo gasto na execução. Foram utilizados sete base de dados do repositório *UCI* com diferentes configurações.

Em termos de acurácia e *AUC* do nível 1, as famílias *lazy* e funções foram as mais instáveis, com picos de desempenho. Por outro lado, famílias bayesianas e também baseadas em regras foram as mais estáveis. De um modo geral, os desempenhos das famílias em termos de acurácia foram semelhantes e em termos de AUC há um desempenho superior das famílias bayesianas.

Em termos de tempo de execução os paradigmas *lazy* e funções tiveram pior desempenho e o nível 2 necessitou de um tempo maior que os demais níveis. Por outro lado, classificadores bayeseanos e regras foram mais rápidos, com destaque para agilidade dos ensembles de família bayesiana, onde o tempo médio de execução foi 949 vezes mais rápido que o nível 2. As bases de dados IS, DB e HC foram as que consumiram maior tempo no nível 2.

As bases de dados HC e GI foram as que apresentaram comportamento mais peculiar entre todas as bases de dados em termos de classificação, de forma que o desempenho entre as famílias se comportaram de maneira diferente nessas bases.

O modelo proposto conseguiu um crescimento em relação aos níveis 0 e 1 da acurácia e $AUC$ de até, respectivamente, 14% e 10%. Foram utilizadas bases de dados binárias e multi-classe com diversas configurações de atributos e instâncias. Para a realização de maiores correlações entre número de classes, número de atributos ou número de instancias com o desempenho do nível 2, pesquisas com um número maior de bases de dados são necessárias.

A pesquisa desenvolvida limitou-se a trabalhar com problemas de classificação, portanto servindo apenas para problemas de aprendizagem supervisionada que não envolvam regressão.

Por meio dos resultados e da análise estatística, pode-se afirmar que o comitê de *ensembles* separados por famílias trouxe bons resultados em termos de acurácia e área sob a curva ROC, tendo como custo, no entanto, alto tempo de treinamento. De forma que se o objetivo principal é aumento na taxa de generalização, o uso do modelo pode ser vantajoso.

## 6.2 TRABALHOS FUTUROS

Autores como Fernández-Delgado et al. (2014), sugerem uma separação das famílias de uma maneira diferente, mais ampla, totalizando 17 famílias, porém com mais semelhanças entre si e incluindo meta - classificadores entre elas. Pode ser interessante uma análise profunda para comparação dos desempenhos entre as diferentes classificações de famílias.

Um ponto a ser trabalhado é o ajuste dos hiper-parâmetros dos classificadores (conjunto de parâmetros que maximizam determinado critério, como acurácia). Métodos estocásticos e não-determinísticos tem sido empregado para realização desses ajustes (PADIERNA et al., 2017), como algoritmos genéticos, otimização por enxame de partículas e abordagens bayesianas.

Outro ponto a ser analisado em trabalhos futuros é a mensuração da diversidade do comitê de *ensembles*. Analisar o desempenho da diversidade deste modelo frente a *ensembles* tradicionais pode demonstrar por que tem sido comum a utilização desse tipo de modelo.

Sugere-se ainda como trabalhos futuros a exploração de comitês de *ensembles* adicionando na análise fatores como a complexidade inerente à base de dados, tema que tem crescido na área e que possibilitaria outras explorações como uma seleção dinâmica de *ensembles*.

# REFERÊNCIAS

ABAWAJY, J. et al. Performance evaluation of multi-tier ensemble classifiers for phishing websites. In: SCHOOL OF INFORMATION SYSTEMS, DEAKIN UNIVERSITY. *ATIS 2012: Proceedings of the 3rd Applications and Technologies in Information Security Workshop*. [S.l.], 2012. p. 11–16.

ABAWAJY, J. et al. Enhancing predictive accuracy of cardiac autonomic neuropathy using blood biochemistry features and iterative multitier ensembles. *IEEE journal of biomedical and health informatics*, IEEE, v. 20, n. 1, p. 408–415, 2016.

ABAWAJY, J.; KELAREV, A. V. A multi-tier ensemble construction of classifiers for phishing email detection and filtering. In: SPRINGER. *CSS*. [S.l.], 2012. p. 48–56.

ABAWAJY JEMAL H, K.; ANDREI; CHOWDHURY, M. Large iterative multitier ensemble classifiers for security of big data. *IEEE Transactions on Emerging Topics in Computing*, IEEE, v. 2, n. 3, p. 352–363, 2014.

AKAY, M. F. Support vector machines combined with feature selection for breast cancer diagnosis. *Expert systems with applications*, Elsevier, v. 36, n. 2, p. 3240–3247, 2009.

ALBERTO, B.; ALMEIDA, P. *Abordagens de pré-processamento de dados em problemas de classificaçao com classes desbalanceadas*. Tese (Doutorado) — Master's Thesis, Centro Federal de Educaçao Tecnológica de Minas Gerais (Mestrado em Modelagem Matemática e Computacional), 2012.

ALDAYEL, M. S. K-nearest neighbor classification for glass identification problem. In: IEEE. *Computer Systems and Industrial Informatics (ICCSII), 2012 International Conference on*. [S.l.], 2012. p. 1–5.

ALPAYDIN, E. *Introduction to Machine Learning. [Sl]*. [S.l.]: The MIT Press, 2010.

ALVES, T. M. *Formação de indicadores para a psicopatologia do luto*. Tese (Doutorado) — Universidade de São Paulo, 2014.

ASMITA, S.; SHUKLA, K. Review on the architecture, algorithm and fusion strategies in ensemble learning. *International Journal of Computer Applications*, Foundation of Computer Science, v. 108, n. 8, 2014.

ASSOCIATION, A. D. et al. Diagnosis and classification of diabetes mellitus. *Diabetes care*, American Diabetes Association, v. 33, n. Suppl 1, p. S62, 2010.

BAUER, E.; KOHAVI, R. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning*, Springer, v. 36, n. 1, p. 105–139, 1999.

BERNARDINI, F. C. *Combinação de classificadores simbólicos para melhorar o poder preditivo e descritivo de ensembles*. Tese (Doutorado) — Universidade de São Paulo, 2002.

BOST, R. et al. Machine learning classification over encrypted data. In: *NDSS*. [S.l.: s.n.], 2015.

BREIMAN, L. Bagging predictors. *Machine learning*, Springer, v. 24, n. 2, p. 123–140, 1996.

BROWN, G. *Diversity in neural network ensembles*. Tese (Doutorado) — University of Birmingham, 2004.

BROWN, G. et al. Diversity creation methods: a survey and categorisation. *Information Fusion*, Elsevier, v. 6, n. 1, p. 5–20, 2005.

BRUN, A. L. *Geração e Seleção de Classificadores com base na Complexidade do Problema*. Tese (Doutorado) — Pontifícia Universidade Católica do Paraná, 2017.

CHANDRA, A.; YAO, X. Ensemble learning using multi-objective evolutionary algorithms. *Journal of Mathematical Modelling and Algorithms*, Springer, v. 5, n. 4, p. 417–445, 2006.

CHAWLA, N. V.; SYLVESTER, J. Exploiting diversity in ensembles: Improving the performance on unbalanced datasets. In: SPRINGER. *International Workshop on Multiple Classifier Systems*. [S.l.], 2007. p. 397–406.

CHEN, Y.; WONG, M. L. An ant colony optimization approach for stacking ensemble. In: IEEE. *Nature and Biologically Inspired Computing (NaBIC), 2010 Second World Congress on*. [S.l.], 2010. p. 146–151.

CHOI, S.-S.; CHA, S.-H.; TAPPERT, C. C. A survey of binary similarity and distance measures. *Journal of Systemics, Cybernetics and Informatics*, Citeseer, v. 8, n. 1, p. 43–48, 2010.

CLEARY, J. G.; TRIGG, L. E. et al. K*: An instance-based learner using an entropic distance measure. In: *Proceedings of the 12th International Conference on Machine learning*. [S.l.: s.n.], 1995. v. 5, p. 108–114.

COHEN, W. W. Fast effective rule induction. In: *Proceedings of the twelfth international conference on machine learning*. [S.l.: s.n.], 1995. p. 115–123.

CORTEZ, P. et al. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, Elsevier, v. 47, n. 4, p. 547–553, 2009.

CORTEZ, P.; EMBRECHTS, M. J. Using sensitivity analysis and visualization techniques to open black box data mining models. *Information Sciences*, Elsevier, v. 225, p. 1–17, 2013.

COSTA, J. D. P. *Ensemble methods in ordinal data classification*. Dissertação (Mestrado) — Faculdade de Engenharia da Universidade do Porto, 2014.

COVER, T.; HART, P. Nearest neighbor pattern classification. *IEEE transactions on information theory*, IEEE, v. 13, n. 1, p. 21–27, 1967.

DAISTER, L. P. *Estratégias Para Desenvolvimento de Sistemas de Múltiplos Classificadores em Aprendizado Supervisionado*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO DE JANEIRO, 2007.

DANGETI, P. *Statistics for Machine Learning: Techniques for exploring supervised, unsupervised, and reinforcement learning models with Python and R*. 1. ed. [S.l.]: Packt Publishing, 2017. ISBN 9781788295758.

DETTLING, M.; BÜHLMANN, P. Boosting for tumor classification with gene expression data. *Bioinformatics*, Oxford University Press, v. 19, n. 9, p. 1061–1069, 2003.

DIETTERICH, T. G. et al. Ensemble methods in machine learning. *Multiple classifier systems*, Springer, v. 1857, p. 1–15, 2000.

DOMINGOS, P. A unified bias-variance decomposition. In: *Proceedings of 17th International Conference on Machine Learning*. [S.l.: s.n.], 2000. p. 231–238.

DOMINGOS, P.; PAZZANI, M. On the optimality of the simple bayesian classifier under zero-one loss. *Machine learning*, Springer, v. 29, n. 2, p. 103–130, 1997.

DRUMMOND, C.; HOLTE, R. C. Cost curves: An improved method for visualizing classifier performance. *Machine learning*, Springer, v. 65, n. 1, p. 95–130, 2006.

FERNÁNDEZ-DELGADO, M. et al. Do we need hundreds of classifiers to solve real world classification problems. *Journal of Machine Learning Research*, v. 15, n. 1, p. 3133–3181, 2014.

FERREIRA, L. T. et al. Diabetes melito: hiperglicemia crônica e suas complicações. *Arquivos Brasileiros de Ciências da Saúde*, v. 36, n. 3, 2011.

FREUND, Y.; SCHAPIRE, R. E. Experiments with a new boosting algorithm. In: *In Proceedings of the Thirteenth International Conference on Machine Learning*. [S.l.]: Morgan Kaufmann, 1996. p. 148–156.

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, v. 55, n. 1, p. 119 – 139, 1997. ISSN 0022-0000. <http://www.sciencedirect.com/science/article/pii/S002200009791504X>.

FUMERA, G.; PILLAI, I.; ROLI, F. A two-stage classifier with reject option for text categorisation. In: SPRINGER. *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. [S.l.], 2004. p. 771–779.

HÄMÄLÄINEN, W.; VINNI, M. Classifiers for educational data mining. *Handbook of educational data mining*, Data Mining and Knowledge Discovery Series. CRC Press, Boca Raton, FL, p. 57–74, 2010.

HAN, J.; PEI, J.; KAMBER, M. *Data mining: concepts and techniques*. [S.l.]: Elsevier, 2011.

HANSEN, L. K.; SALAMON, P. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 12, n. 10, p. 993–1001, 1990.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. Overview of supervised learning. In: *The elements of statistical learning*. [S.l.]: Springer, 2009. p. 9–41.

HORTON, P.; NAKAI, K. Better prediction of protein cellular localization sites with the it k nearest neighbors classifier. In: *Ismb*. [S.l.: s.n.], 1997. v. 5, p. 147–152.

HU, L.-Y. et al. The distance function effect on k-nearest neighbor classification for medical datasets. *SpringerPlus*, Nature Publishing Group, v. 5, n. 1, p. 1304, 2016.

HÜHN, J.; HÜLLERMEIER, E. Furia: an algorithm for unordered fuzzy rule induction. *Data Mining and Knowledge Discovery*, Springer, v. 19, n. 3, p. 293–319, 2009.

JADHAV, S. D.; CHANNE, H. Comparative study of k-nn, naive bayes and decision tree classification techniques. *International Journal of Science and Research*, v. 5, n. 1, 2016.

JAMIL, L. S. Data analysis based on data mining algorithms using weka workbench. *International Journal of Engineering Sciences & Research Technology*, 2016.

JELINEK, H. et al. Decision trees and multi-level ensemble classifiers for neurological diagnostics. *Australian Journal of Medical Science*, Australian Institute of Medical Scientists, v. 1, n. 1, p. 1–12, 2014.

JOHNSON, P.; TYMMS, P. The emergence of a learning progression in middle school chemistry. *Journal of Research in Science Teaching*, Wiley Online Library, v. 48, n. 8, p. 849–877, 2011.

JUREK, A. et al. A survey of commonly used ensemble-based classification techniques. *The Knowledge Engineering Review*, Cambridge University Press, v. 29, n. 5, p. 551–581, 2014.

KALYANI, G.; LAKSHMI, A. J. Performance assessment of different classification techniques for intrusion detection. *Learning*, v. 2, n. 1, p. J48, 2012.

KELAREV, A. V. et al. Improving classifications for cardiac autonomic neuropathy using multi-level ensemble classifiers and feature selection based on random forest. In: AUSTRALIAN COMPUTER SOCIETY, INC. *Proceedings of the Tenth Australasian Data Mining Conference-Volume 134*. [S.l.], 2012. p. 93–101.

KNOWLER, W. et al. Islet cell antibodies and diabetes mellitus in pima indians. *Diabetologia*, Springer, v. 17, n. 3, p. 161–164, 1979.

KOTSIANTI, S.; KANELLOPOULOS, D. Combining bagging, boosting and dagging for classification problems. In: SPRINGER. *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. [S.l.], 2007. p. 493–500.

KOTSIANTIS, S. Supervised machine learning: A review of classification techniques. *Informatica*, v. 31, p. 249–268, 2007.

KOTSIANTIS, S. B.; ZAHARAKIS, I.; PINTELAS, P. Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, v. 160, p. 3–24, 2007.

KOZEGAR, E. et al. Assessment of a novel mass detection algorithm in mammograms. *Journal of cancer research and therapeutics*, Medknow Publications & Media Pvt. Ltd., v. 9, n. 4, p. 592, 2013.

KUNCHEVA, L. I. *Combining pattern classifiers: methods and algorithms*. [S.l.]: John Wiley & Sons, 2004.

LETHAM, B. et al. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*, Institute of Mathematical Statistics, v. 9, n. 3, p. 1350–1371, 2015.

LEVESQUE, J.; GAGNÉ, C.; SABOURIN, R. Bayesian hyperparameter optimization for ensemble learning. *CoRR*, abs/1605.06394, 2016. <http://arxiv.org/abs/1605.06394>.

LICHMAN, M. *UCI Machine Learning Repository*. 2013. <http://archive.ics.uci.edu/ml>.

LIMA, N. H. C. *Classificação de padrões através de um comitê de máquinas aprimorado por aprendizagem por reforço*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, 2012.

LIMA, T. P. F. d. An authomatic method for construction of multi-classifier systems based on the combination of selection and fusion. Universidade Federal de Pernambuco, 2013.

LIU, Y.; YAO, X. Ensemble learning via negative correlation. *Neural networks*, Elsevier, v. 12, n. 10, p. 1399–1404, 1999.

LIU, Y.; YAO, X.; HIGUCHI, T. Evolutionary ensembles with negative correlation learning. *IEEE Transactions on Evolutionary Computation*, IEEE, v. 4, n. 4, p. 380–387, 2000.

LORBIESKI, R.; NASSAR, S. Impact of an extra layer on the stacking algorithm for classification problems. *Journal of Computer Sciences*, 2018. <http://thescipub.com/pdf/10.3844/ofsp.11704>.

LORBIESKI, R.; NASSAR, S. M. Performance assessment of multi-level ensemble for multi-class problems. *International Journal of Computer and Information Engineering*, v. 12, n. 3, 2018.

MARKOWETZ, F. Classification by support vector machines. *Practical DNA Microarray Analysis*, 2003.

MENAHEM, E.; ROKACH, L.; ELOVICI, Y. Troika–an improved stacking schema for classification tasks. *Information Sciences*, Elsevier, v. 179, n. 24, p. 4097–4122, 2009.

MENDES-MOREIRA, J. et al. Ensemble approaches for regression: A survey. *ACM Computing Surveys (CSUR)*, ACM, v. 45, n. 1, p. 10, 2012.

MOTTA, P. R. d. A. et al. Estudo exploratório do uso de classificadores para a predição de desempenho e abandono em universidade. Universidade Federal de Goiás, 2016.

MUÑOZ, L. A. B.; GONZÁLEZ, J. H. Similarity networks for classification: a case study in the horse colic problem. 2014.

NASCIMENTO, D. S. C. *Configuração heterogênea de ensembles de classificadores: Investigação em bagging, boosting e multiboosting*. Tese (Doutorado) — Dissertação de mestrado, Universidade de Fortaleza. UNIFOR., Fortaleza, CE, 2009.

NETO, L.; NASCIMENTO, A. do. *Classificação com algoritmo AdaBoost. M1: o mito do limiar de erro de treinamento*. Dissertação (Mestrado) — Pontifícia Universidade Católica do Rio Grande do Sul, 2017.

ORGANIZATION, W. H. et al. *International Agency For Research on Cancer GLOBOCAN 2012: estimated cancer incidence, mortality and prevalence worldwide in 2012*. [S.l.]: Geneva, 2012.

OZA, N. C.; RUSSELL, S. *Online ensemble learning*. [S.l.]: University of California, Berkeley, 2001.

OZAY, M.; YARMAN-VURAL, F. T. Hierarchical distance learning by stacking nearest neighbor classifiers. *Information Fusion*, Elsevier, v. 29, p. 14–31, 2016.

OZGUR, A. Supervised and unsupervised machine learning techniques for text document categorization. *Yüksek Lisans Tezi*, 2004.

PADIERNA, L. C. et al. Hyper-parameter tuning for support vector machines by estimation of distribution algorithms. In: _____. *Nature-Inspired Design of Hybrid Intelligent Systems*. Cham: Springer International Publishing, 2017. p. 787–800. ISBN 978-3-319-47054-2. <"https://doi.org/10.1007/978-3-319-47054-2">.

PARENTE, R. R. *Abordagem de construção de arquitetura homogênea para comitês via meta-aprendizagem*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, 2012.

PENG, Y. et al. Empirical evaluation of classifiers for software risk management. *International Journal of Information Technology & Decision Making*, World Scientific, v. 8, n. 04, p. 749–767, 2009.

PILA, A. D. *Seleção de atributos relevantes para aprendizado de máquina utilizando a abordagem de Rough Sets*. Tese (Doutorado) — Universidade de São Paulo, 2001.

POLIKAR, R. Ensemble based systems in decision making. *IEEE Circuits and systems magazine*, IEEE, v. 6, n. 3, p. 21–45, 2006.

PRATI, R.; BATISTA, G.; MONARD, M. Curvas roc para avaliação de classificadores. *Revista IEEE América Latina*, v. 6, n. 2, p. 215–222, 2008.

REN, Y.; ZHANG, L.; SUGANTHAN, P. N. Ensemble classification and regression-recent developments, applications and future directions. *IEEE Computational Intelligence Magazine*, IEEE, v. 11, n. 1, p. 41–53, 2016.

ROKACH, L. Taxonomy for characterizing ensemble methods in classification tasks: A review and annotated bibliography. *Computational Statistics & Data Analysis*, Elsevier, v. 53, n. 12, p. 4046–4072, 2009.

ROKACH, L. Ensemble-based classifiers. *Artificial Intelligence Review*, Springer, v. 33, n. 1-2, p. 1–39, 2010.

SCHAPIRE, R. E. The strength of weak learnability. *Machine learning*, Springer, v. 5, n. 2, p. 197–227, 1990.

SEEWALD, A. K. How to make stacking better and faster while also taking care of an unknown weakness. In: MORGAN KAUFMANN PUBLISHERS INC. *Proceedings of the nineteenth international conference on machine learning*. [S.l.], 2002. p. 554–561.

SESMERO, M. P.; LEDEZMA, A. I.; SANCHIS, A. Generating ensembles of heterogeneous classifiers using stacked generalization. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Wiley Online Library, v. 5, n. 1, p. 21–34, 2015.

SKURICHINA, M.; DUIN, R. P. Combining feature subsets in feature selection. In: SPRINGER. *International Workshop on Multiple Classifier Systems*. [S.l.], 2005. p. 165–175.

SOMAN, K.; DIWAKAR, S.; AJAY, V. *Data Mining: Theory and Practice [WITH CD]*. [S.l.]: PHI Learning Pvt. Ltd., 2006.

STEINER, M. T. A. et al. Data mining como suporte à tomada de decisões-uma aplicação no diagnóstico médico. *XXXVI SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL,"O IMPACTO DA PESQUISA OPERACIONAL NAS NOVAS TENDÊNCIAS MULTIDISCIPLINARES*, v. 23, p. 96–107, 2004.

SU, Y. et al. Hierarchical ensemble of global and local classifiers for face recognition. *IEEE Transactions on Image Processing*, IEEE, v. 18, n. 8, p. 1885–1896, 2009.

SURESH, S.; DONG, K.; KIM, H. A sequential learning algorithm for self-adaptive resource allocation network classifier. *Neurocomputing*, Elsevier, v. 73, n. 16-18, p. 3012–3019, 2010.

TAN, P.-N. Introduction to data mining. Addison-Wesley, 2013.

TANG, B. et al. Reranking for stacking ensemble learning. *Neural Information Processing. Theory and Algorithms*, Springer, p. 575–584, 2010.

TING, K. M.; WITTEN, I. H. Stacked generalization: when does it work? Department of Computer Science, University of Waik, 1997.

TING, K. M.; WITTEN, I. H. Stacking bagged and dagged models. In: *In Proc. 14th International Conference on Machine Learning*. [S.l.: s.n.], 1997.

TING, K. M.; WITTEN, I. H. Issues in stacked generalization. *J. Artif. Intell. Res.(JAIR)*, v. 10, p. 271–289, 1999.

TJADEN, B.; COHEN, J. A survey of computational methods used in microarray data interpretation. In: *Applied Mycology and Biotechnology*. [S.l.]: Elsevier, 2006. v. 6, p. 161–178.

TODOROVSKI, L.; DŽEROSKI, S. Combining classifiers with meta decision trees. *Machine learning*, Springer, v. 50, n. 3, p. 223–249, 2003.

TSIROGIANNIS, G. et al. Classification of medical data with a robust multi-level combination scheme. In: IEEE. *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*. [S.l.], 2004. v. 3, p. 2483–2487.

TUMER, K.; GHOSH, J. Error correlation and error reduction in ensemble classifiers. *Connection science*, Taylor & Francis, v. 8, n. 3-4, p. 385–404, 1996.

TURNER, K.; OZA, N. C. Decimated input ensembles for improved generalization. In: IEEE. *Neural Networks, 1999. IJCNN'99. International Joint Conference on*. [S.l.], 1999. v. 5, p. 3069–3074.

WAEGEMAN, W.; BAETS, B. D.; BOULLART, L. A comparison of different roc measures for ordinal regression. In: CITESEER. *Proceedings of the 3rd International Workshop on ROC Analysis in Machine Learning., N. Lachiche, C. Ferri, and S. Macskassy, Eds.* [S.l.], 2006. p. 63–69.

WEISS, S. M.; KULIKOWSKI, C. A. *Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning, and expert systems*. [S.l.]: Morgan Kaufmann Publishers Inc., 1991.

WITTEN, I. H. et al. *Data Mining: Practical machine learning tools and techniques*. [S.l.]: Morgan Kaufmann, 2016.

WOLPERT, D. H. Stacked generalization. *Neural networks*, Elsevier, v. 5, n. 2, p. 241–259, 1992.

WOLPERT, D. H. The lack of a priori distinctions between learning algorithms. *Neural computation*, MIT Press, v. 8, n. 7, p. 1341–1390, 1996.

Yamazaki, T. et al. Ensemble Models for Detecting Wikidata Vandalism with Stacking - Team Honeyberry Vandalism Detector at WSDM Cup 2017. *ArXiv e-prints*, dez. 2017.

YANG, J. et al. Feature subset selection for rule induction using ripper. In: *Proceedings of the Genetic and Evolutionary Computation Conference*. [S.l.: s.n.], 1999. v. 2, p. 1800.