

Glécio de Oliveira Santos

Benchmarking em algoritmos de classificação na mineração de dados

São Luis - Maranhão

2018

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

Glécio de Oliveira Santos Benchmarking em algoritmos de classificação na mineração de dados/ Glécio de Oliveira Santos. – São Luis - Maranhão, 2018- 40 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Ivo José da Cunha Serra

Monografia – UNIVERSIDADE FEDERAL DO MARANHÃO - UFMA

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO, 2018.

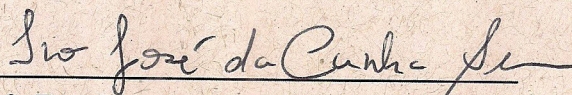
1. Benchmarking. 2. Data Mining. 2. Weka. I. Orientador. II. Universidade Federal do Maranhão. III. Curso de Graduação em Ciência da Computação. IV. Benchmarking em algoritmos de classificação na mineração de dados.

Glécio de Oliveira Santos

Benchmarking em algoritmos de classificação na mineração de dados

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

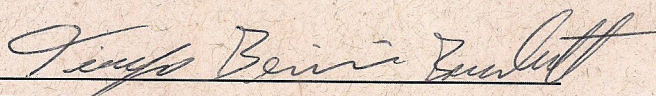
Trabalho aprovado. São Luis - Maranhão, 12 de julho de 2018:



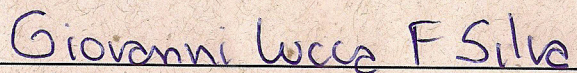
Prof. Dr. Ivo José da Cunha Serra
Orientador



Prof. Msc. Carlos Eduardo Portela
Serra de Castro
Banca examinadora



Prof. Dr. Tiago Bonini Borchardt
Banca examinadora



Prof. Msc. Giovanni Lucca França da
Silva
Banca examinadora

São Luis - Maranhão
2018

Resumo

A tomada de decisões baseada na análise de grandes quantidades de dados é tarefa essencial nas mais diversas áreas. Com o desenvolvimento de novas tecnologias, permitindo assim um maior fluxo de dados entre diferentes dispositivos, uma crescente quantidade de informação agora se apresenta. O uso de mineração de dados permite manipular esses dados para extração de informações, que podem resultar em conhecimento. A mineração de dados constantemente tem sido alvo de estudos e pesquisas para aprimorar seus resultados. Dentre as etapas fundamentais no desempenho da mineração de dados está a de classificação. Diferentes algoritmos podem ser utilizados para essa etapa, obtendo-se resultados mais ou menos eficientes dependendo do cenário em questão. O trabalho apresentado visa utilizar a técnica de *benchmarking* como forma de comparar o desempenho de cada um dos algoritmos de classificação para uma mesma base de dados. Os resultados serão obtidos por meio de um experimento utilizando a ferramenta WEKA, e irão servir como base na escolha da melhor solução.

Palavras-chave: *benchmarking*, mineração de dados, weka, big data, classificação.

Abstract

Decision making based on the analysis of large amounts of data is an essential task in many different areas. With the development of new technologies, thus allowing a greater flow of data between different devices, an increasing amount of information now presents itself. The use of data mining allows manipulating this data for extracting information, which can result in knowledge. Data mining has consistently been the subject of studies and research to improve its results. Among the key steps in the performance of data mining is classification. Different algorithms can be used for this step, obtaining more or less efficient results depending on the scenario in question. The work presented aims to use the benchmarking technique as a way to compare the performance of each of the classification algorithms for the same database. The results will be obtained through an experiment using the WEKA tool, and will serve as the basis for choosing the best solution.

Keywords: *benchmarking*, data mining, weka, big data, classifiers.

Lista de ilustrações

Figura 1 – Geração de <i>benchhmarking</i>	15
Figura 2 – Diretivas do arquivo ARFF da base: diabetes.arff	21
Figura 3 – Tela Inicial do ambiente WEKA	24
Figura 4 – WEE - <i>Setup</i>	25
Figura 5 – WEE - <i>Run</i>	25
Figura 6 – WEE - <i>Analyse</i>	26
Figura 7 – Matriz de Confusão para duas classes	27
Figura 8 – Num. Corretos x Incorretos	28
Figura 9 – Exatidão (<i>Acurácia</i>)	29
Figura 10 – FPR e FNR	30
Figura 11 – <i>F-Measure</i>	31
Figura 12 – ROC	32

Lista de abreviaturas e siglas

AUC	<i>Area under the ROC curve</i>
ARFF	<i>Attribute-Relation File Format</i>
CSV	<i>Comma Separated Values</i>
DM	<i>Data Mining</i>
TPR	<i>True positive rate</i>
TNR	<i>True negative rate</i>
FPR	<i>False positive rate</i>
FNR	<i>False negative rate</i>
IA	<i>Inteligência Artificial</i>
Ibk	<i>Instance based learning</i>
IoT	<i>Internet of things</i>
KDD	<i>Knowledge discovery in databases</i>
mm	<i>milímetros</i>
mmHG	<i>milímetros de mercúrio</i>
ML	<i>Machine Learning</i>
ROC	<i>Receiver Operator Characteristic</i>
SGBD	<i>Sistema de gerenciamento de bancos de dados</i>
U/ml	<i>Unidades por mililitro</i>
WEE	<i>WEKA Experiment Environment</i>
WEKA	<i>Waikato Environment for Knowledge Analysis</i>

Sumário

1	INTRODUÇÃO	9
1.1	Motivação	10
1.2	Objetivos	10
1.2.1	Objetivo geral	10
1.2.2	Objetivos específicos	10
1.3	Organização do trabalho	11
2	BENCHMARKING	12
2.1	Introdução	12
2.2	Significado e definições	13
2.3	Considerações Históricas	14
2.4	Gerações de <i>benchmarking</i>	15
2.5	Princípios do <i>benchmarking</i>	16
2.6	Tipos de <i>benchmarking</i>	17
2.7	<i>Benchmarking</i> na pesquisa	18
2.8	Benchmarking em Mineração de Dados	19
3	BENCHMARKING EM MINERAÇÃO DE DADOS NO WEKA	20
3.1	A ferramenta WEKA	20
3.2	<i>Benchmarking</i> no ambiente WEKA	21
3.2.1	Formatos de arquivos	21
3.3	Base de dados adotada	22
3.4	Classificadores utilizados	22
3.4.1	Descrição dos algoritmos utilizados	23
3.5	Realizando os testes no WEKA	24
3.6	Resultados	26
3.6.1	Principais Métricas analisadas	26
3.6.1.1	Matriz de Confusão	27
3.6.1.2	Instâncias classificadas corretamente	27
3.6.1.3	Instâncias classificadas incorretamente	28
3.6.1.4	Exatidão(<i>Acurácia</i>)	29
3.6.1.5	Taxa de verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos	29
3.6.1.6	Curvas ROC (<i>Receiver Operating Characteristic Curve</i>)	31
4	CONCLUSÃO	34

APÊNDICES	36
APÊNDICE A – DADOS BRUTOS	38
REFERÊNCIAS	39

1 Introdução

A crescente popularização da internet, somada à diminuição dos custos de armazenamento, trouxe consigo um aumento na quantidade de dados trafegando diariamente na rede. Novas tecnologias como: internet das coisas (IoT), redes sociais, automação residencial entre muitas, surgiram conectando diversos dispositivos e produzindo dados de diferentes tipos em quantidades cada vez maiores (AMARAL, 2016).

Essas mudanças contribuíram diretamente para o surgimento do fenômeno *Big Data*—que são conjuntos de dados sendo produzidos com grande volume, variedade e velocidade—isso fez com que novos modelos para manipular esses dados tivessem que ser criados.

A extração de informação relevante passa por um processo de tratamento desses dados. É preciso que eles sejam transformados em informação, para posteriormente gerar conhecimento. Com a mineração de dados essa tarefa se torna possível. Em conjunto com a inteligência artificial, a mineração de dados pode realizar análises complexas de grandes quantidades de dados, tornando-se uma ferramenta decisiva em muitas áreas, e primordial em tomadas de decisão. Uma das principais tarefas na mineração de dados é a de classificação. Nessa tarefa, dependendo do problema, pode demandar alta complexidade e tempo de processamento.

Diversos algoritmos existem para a execução dessa tarefa. Objetivando colher os melhores resultados, processos comparativos podem ser utilizados. O *benchmarking*—técnica que consiste em comparar resultados de desempenho entre sistemas similares tem sido adotado amplamente em áreas como indústria, administração, pesquisa, entre muitas. Podendo ser utilizada tanto na comparação de técnicas como de ferramentas (QUEIROZ et al., 2013).

Na mineração de dados, o uso do *benchmarking* pode ser adotado para comparar o desempenho de diferentes algoritmos e assim otimizar a solução de um determinado problema. Desse modo, é possível ter bons critérios na tomada de decisões e possíveis mudanças nas abordagens escolhidas. O trabalho apresentado visa utilizar o *benchmarking* como técnica de comparação entre os diferentes algoritmos de classificação, auxiliando assim na discussão sobre quais algoritmos possuem melhor eficácia mediante um determinado problema, além de servir como referência na obtenção de estratégias que gerem melhores resultados. O trabalho irá utilizar o *benchmarking* para a avaliação de algoritmos de classificação, em um processo de mineração de dados, apresentando resultados e analisando as características de cada um deles. O objetivo é esclarecer como podemos mensurar o desempenho de diferentes algoritmos de classificação em um cenário de mineração de dados

através do ambiente WEKA.

1.1 Motivação

As novas áreas como inteligência artificial, mineração de dados e *big data*, estão em plena expansão. Diversas áreas têm se utilizado de seus benefícios para encontrar soluções para problemas que antes seriam muito mais complexos. Essas áreas estão revolucionando a forma como a computação interage com o mundo e está mais presente na vida de todos. A mineração de dados consegue extrair valioso conhecimento onde antes seriam considerados apenas como dados. A análise dos dados de uma pesquisa sobre diabetes é um dos exemplos de como é possível utilizar técnicas de mineração de dados para aprender com os dados coletados e gerar um modelo que possa estimar se um determinado indivíduo está propenso ou não a ter a doença. Realizar essa operação de maneira mais eficaz, permite que esse modelo possa ser aplicado amplamente.

1.2 Objetivos

1.2.1 Objetivo geral

O trabalho objetiva esclarecer como podemos mensurar o desempenho de diferentes algoritmos de classificação em um cenário de *data mining* através do ambiente *WEKA*.

1.2.2 Objetivos específicos

- Conceituar o *benchmarking*, suas metodologias e sua aplicabilidade na análise de algoritmos;
- Detalhar a etapa de classificação em mineração de dados, os principais algoritmos que podem ser utilizados assim como melhor desempenho;
- Contribuir com a comunidade científica na divulgação dos resultados de um experimento através da ferramenta *WEKA*;
- Explicar como o *benchmarking* funciona, qual sua utilidade e como pode ser implementado;
- Efetuar o *benchmarking* dos diferentes tipos de algoritmos de classificação, analisar as métricas e avaliar os resultados obtidos, através de um estudo de caso utilizando o ambiente *WEKA*.

1.3 Organização do trabalho

O trabalho se organiza em 4 capítulos, sendo os principais tópicos apresentados nos Capítulos 2 e 3. No Capítulo 2 é abordado o *benchmarking* seus conceitos e aplicações. Logo em seguida, no Capítulo 3 é realizado um experimento tendo como ferramenta base o ambiente WEKA. Ao final do Capítulo 3 são apresentados os resultados obtidos pela ferramenta, com gráficos comparativos entre as principais métricas.

2 *Benchmarking*

Nesse capítulo serão abordados os aspectos sobre o *benchmarking*, sua história e definições, além de mostrar como pode ser aplicado em um cenário cujo objetivo é a otimização dos resultados.

2.1 Introdução

O contínuo crescimento da quantidade de serviços que se utilizam da internet trouxe à tona a necessidade de se analisar os dados gerados pelos mesmos, haja visto a grande importância estratégica que esses mesmos dados possuem em seus segmentos. A separação de conteúdo com relevância pode ser obtida com o uso de técnicas voltadas para essa tarefa. A mineração de dados é ferramenta crucial na análise desses dados e obtenção de informação relevante. Seu processo de funcionamento passa por etapas bem definidas e por vezes complexas ([AMARAL, 2016](#)).

Em um cenário de competitividade mundial cada vez mais delimitado, o uso de ferramentas de maior eficiência se torna primordial na conquista de melhores resultados. A escolha de um algoritmo que apresente o melhor desempenho em um processo de mineração de dados, pode ser decisivo no sucesso ou não da solução de um problema.

Como uma das principais técnicas do modelo experimental de avaliação de desempenho, o *benchmarking* se consolidou como um dos principais métodos de comparação, podendo ser aplicado a diferentes cenários, objetivando atingir a melhor performance em seus processos como um todo ou apenas em partes do mesmo. O uso de *benchmarking* permite comparar os resultados obtidos em um cenário qualquer com outros da mesma classe. Diversas áreas se utilizam de seus princípios, como administração, economia, engenharia, além da computação.

Através de comparações sucessivas entre tarefas similares, dados estatísticos de performance são obtidos e servem como base nas avaliações.

2.2 Significado e definições

Benchmarking é uma palavra da língua inglesa, que, usada como verbo, indica a ação de medir a qualidade de algo com um padrão aceito (CAMBRIDGE, 2018).

Analisada como substantivo, “*benchmark*” refere-se a um nível de qualidade que pode ser usado como referência quando comparado a outros (WEBSTER, 1828).

Segundo Araújo Júnior (2001, p. 241): "A origem do termo benchmarking é oriunda da agrimensura, em que marcações eram utilizadas para definir um marco no terreno, com a finalidade de permitir comparações de altura, direção, distância, entre outros", citado em (MARTINS; SANTOS; CARVALHO, 2010).

Conforme Joo, Nixon, & Stoeberl(2011) definem o termo *benchmarking* a um tipo de abordagem de gerenciamento caracterizada pela implementação de "melhores práticas encontradas em indústrias similares ou mesmo em diferentes indústrias a fim de melhorar a performance de uma organização", citado em (ABBAS, 2014).

Dias (2008) situa as origens do *benchmarking* como: "As raízes linguísticas e metafóricas do benchmarking vêm do termo usado pelos agrimensores, que designavam benchmarking como uma marca ou referência feita sobre uma rocha, muro ou edifício". Portanto um *benchmarking* servia como referência para determinar sua posição ou altitude em medidas topográficas ou registros das marés, citado em (MARTINS; SANTOS; CARVALHO, 2010).

Analisando as diferentes definições, existem um consenso em atribuir ao *benchmarking* papel de método comparativo e de melhoria contínua em um processo ou sistema. Apesar de existirem diferentes definições a respeito do *benchmarking*, a maioria delas é basicamente originária do conceito criado por Robert Camp (CAMP, 1989).

2.3 Considerações Históricas

Ao longo da história, a humanidade vem continuamente buscando aprimorar suas atividades para obter melhores condições de sobrevivência. Mesmo nas antigas civilizações sempre existiu a busca por melhores soluções para os problemas apresentados. Algumas das descobertas mais importantes tiveram origens devido à essa busca pela solução de problemas de modo eficiente. Em várias fases da história, houveram práticas que muito se assemelhavam ao *benchmarking*, mas sem que fossem assim chamadas.

Alguns fatos históricos podem ser caracterizados como marcos no desenvolvimento do *benchmarking*. Um desses fatos ocorreu durante a década de 50, quando indústrias japonesas, no intuito de aprimorar suas técnicas de produção, percorreram várias outras indústrias pelo mundo (a maioria na América e Europa ocidental) buscando melhores práticas e técnicas que pudessem ser aproveitados para suas realidades. Essas indústrias estavam nesse momento praticando o *benchmarking* apesar do termo não ter sido adotado ainda.

Já na década de 60, a IBM iniciou estudos para se tornar mais competitiva. Foram traçadas várias regras para que as operações entre as várias unidades fossem padronizadas, desse modo processos de fabricação mais modernos seriam adotados simultaneamente para todas as unidades de produção.

De acordo com (CAMP, 1989), o *benchmarking* começou a ser adotado no final da década de 70, quando em 1979 a empresa Xerox, após queda no seu desempenho em relação às empresas japonesas, se viu obrigada a mudar de estratégia. Pela comparação de seus métodos com o das concorrentes, a Xerox conseguiu identificar os pontos que poderiam ser trabalhados de modo a permitir uma vantagem competitiva. Como resultado, a Xerox conseguiu corrigir esses pontos deficientes e se sobressair entre a concorrência, decidindo assim, adotar o *benchmarking* como estratégia de crescimento para a empresa desde então.

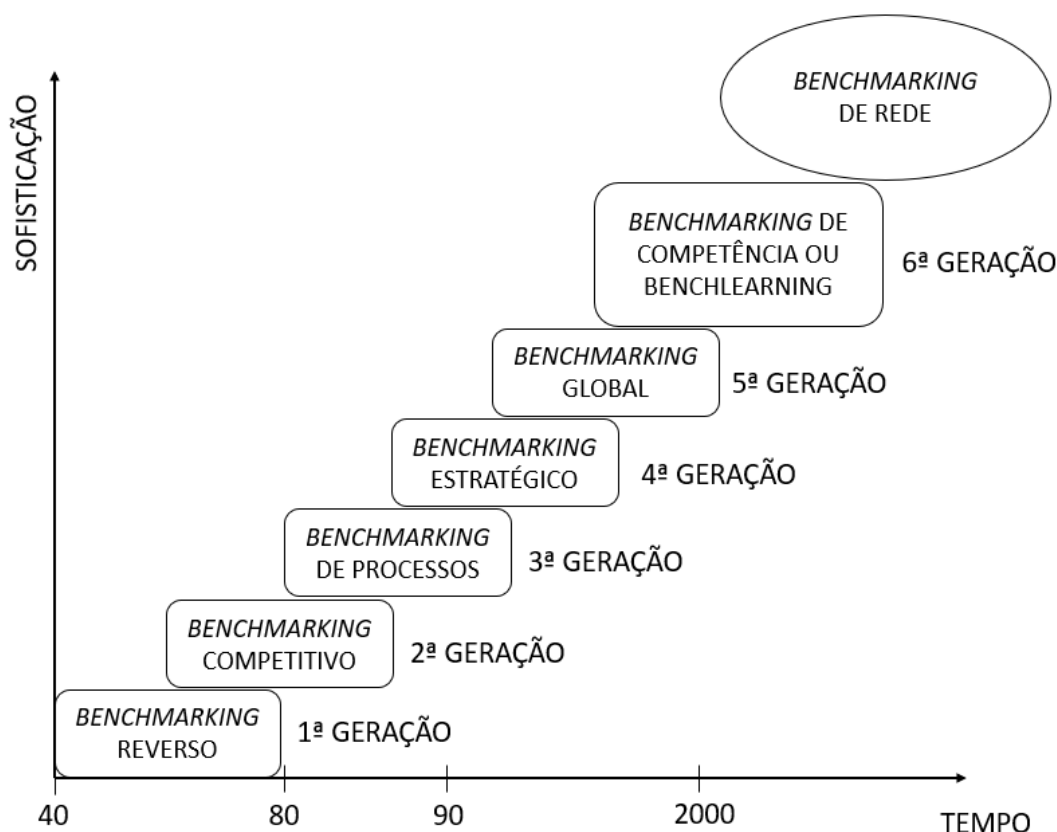
Um dos responsáveis na criação e implantação do *benchmarking* na empresa Xerox, Robert Camp, lançou em 1989 uma das publicações que mais ajudaram a divulgar o *benchmarking*, a obra "*Benchmarking: The Search for Industry Best Practices That Lead to Superior Performance*", onde aborda detalhes da técnica adotada pela empresa, suas etapas e quais os resultados obtidos.

Conforme a técnica ia se popularizando, as empresas que adotaram o *benchmarking* foram modificando o modelo criado pela Xerox e adequando a técnica à suas realidades (WATSON, 1994).

2.4 Gerações de *benchmarking*

Segundo (WATSON, 1993), o *benchmarking* se originou na década de 40 e sofreu aprimoramentos ao longo do tempo. Ele sugere que o *benchmarking* pode ser dividido em cinco gerações, conforme pode ser visto na Figura.1

Figura 1 – Geração de *benchhmarking*



Fonte: Adaptado de Ahmed and Rafiq (1998, p.288) em (KYRÖ, 2003)

- **1ª geração - Engenharia reversa:** Período inicial da utilização do *benchmarking*. Caracterizado pelo foco no produto e não na técnica. Nesse período eram desmontados produtos para que se analisassem sua constituição e características de fabricação. Desse modo, poderia ser feito uma cópia ou mesmo criar um produto superior com características extraídas do concorrente.
- **2ª geração - *Benchmarking* competitivo:** Influenciado pelos estudos iniciados pela Xerox em 1976, esse período caracteriza-se pelo início na mudança de foco na comparação. Não mais se estava interessado em analisar o produto, agora o objetivo era comparação dos processos.
- **3ª geração - *Benchmarking* de processos:** Iniciado na década de 80, essa fase foi marcada pela busca de metodologias mais eficientes em diferentes áreas.

- **4ª geração - *Benchmarking* estratégico:** Nessa fase, após diferentes evoluções, chegou-se a conclusão que dependendo do resultado entre as comparações, uma mudança estratégica poderia se fazer necessária.
- **5ª geração - *Benchmarking* global:** Em um cenário de cooperação mundial mais amplo, as diferentes culturas e características de outras nações são compreendidas como um fator de influência no desempenho dos negócios.

Estudos recentes sugerem uma sexta geração, denominada por *benchmarking* de competências, ou *benchlearning*. O termo *benchlearning*, refere-se aos esforços de uma organização em se tornar cada vez mais aberta ao aprendizado. Karlöf and Östblom (1995) usaram o termo pela primeira vez se referindo a uma mudança de mentalidade. Eles sugerem que as mudanças nas organizações são influenciadas pela mudança nos comportamentos e competências individuais de sua equipe. Em um cenário de cooperação mundial mais amplo, as diferentes culturas e características de outras nações são compreendidas como um fator de influência no desempenho dos negócios (KYRÖ, 2003).

2.5 Princípios do *benchmarking*

O *benchmarking* é o processo de entender o que é importante para o sucesso da sua organização, compreender seus próprios processos, encontrar e aprender com os outros cujos processos são melhores, e então adaptar esse aprendizado para melhorar seu desempenho. O *benchmarking* é muito mais do que copiar. Requer uma autoavaliação profunda e a capacidade de traduzir práticas que funcionam em outro contexto processo apropriado para sua própria organização.

De acordo com Joo et al.(2011), citado em (ABBAS, 2014), o *benchmarking* possui quatro principais objetivos:

- Identificar medidas chave de performance para cada função em uma operação de negócios;
- Medir o próprio nível de desempenho interno bem como os dos principais concorrentes;
- Comparar níveis de performance e identificar áreas de vantagem comparativa e desvantagens;
- Implementar programas para aproximar o desvio existente entre as operações internas e os principais concorrentes.

2.6 Tipos de *benchmarking*

Podemos distinguir os tipos de *benchmarking* como informal e o formal. O *benchmarking* informal é aquele praticado sem muito critério, quase de modo automático, mesmo que sem notar. Quando se aprende através de uma experiência, ou é feito um aprimoramento de uma atividade através de uma nova abordagem, temos esse tipo de *benchmarking* BPIR (2018).

Dentro do *benchmarking* formal podemos ter dois tipos: *benchmarking* de performance e *benchmarking* de melhores práticas. No *benchmarking* de performance o que se deseja é comparar diferentes níveis obtidos para um determinado processo avaliado. São realizadas comparações com níveis obtidos em processos em outros cenários, cujo desempenho é reconhecido entre os melhores BPIR (2018).

No *benchmarking* de melhores práticas, tem-se uma comparação entre metodologias, abordagens ou técnicas diferentes objetivando a escolha da mais eficiente.

As comparações em um processo de *benchmarking* podem ser realizadas com diferentes objetivos e entre diferentes áreas. Companhias, processos, funções e produtos podem ser mensurados e comparados para estipular metas estratégicas. Conforme o tipo de comparação realizado pelo *benchmarking* podemos ter diferentes classificações, apesar de não haver consenso nas diferentes literaturas, de acordo com Spendolini (1993), Araújo Júnior (2001) e Araújo (2000) em (MARTINS; SANTOS; CARVALHO, 2010) o *benchmarking* pode ser classificado principalmente em três tipos:

- *Benchmarking* interno – Quando diferentes departamentos ou setores são comparados dentro de uma mesma companhia;
- *Benchmarking* competitivo – realiza uma comparação para extrair o resultado de melhor performance entre os avaliados;
- *Benchmarking* funcional ou genérico – Tem como objetivo tornar a companhia em avaliação a melhor em termos de processos e tecnologia. Pode ser aplicado em uma companhia ou área da tecnologia;

Algumas divergências podem existir para diferentes autores, mas os principais tipos são os citados.

2.7 Benchmarking na pesquisa

Apesar de, desde o início, o *benchmarking* se mostrar como ferramenta de melhoria nos processos para as companhias, com o decorrer do tempo outras áreas foram descobrindo as vantagens na utilização de seus princípios para aprimorar suas operações. É o caso da ciência, que frequentemente analisa diferentes métodos ou ferramentas de solução para um determinado problema e os compara, estabelecendo conclusões sobre os pontos positivos e negativos entre cada um.

Conforme (CIFERRI et al., 1995) citado em (QUEIROZ et al., 2013), para analisar o desempenho de um sistema, existem dois modelos que podem ser seguidos, que são: modelo analítico e modelo de simulação ou experimental. As técnicas que constituem o modelo experimental, são: monitoração e *benchmarking*.

A técnica de monitoração não possui padronização para as análises realizadas. Cada ferramenta possui sua própria técnica de avaliação, dificultando que diferentes ferramentas possam ser analisadas com equidade. Diferentemente, o *benchmarking* utiliza tarefas padronizados para realizar testes em diferentes modelos, o que garante uma comparação dos resultados de modo preciso (DEWITT, 1985) citado em (QUEIROZ et al., 2013).

No caso específico da Ciência da Computação, a análise de desempenho por meio de benchmarking é aplicada em diversos segmentos. Conforme (QUEIROZ et al., 2013) para todos esses cenários analisados, as principais soluções que se deseja obter são:

- Avaliar a capacidade máxima do sistema;
- Comparar diferentes tecnologias;
- Avaliar a viabilidade de um sistema mediante um cenário;
- Medir a relação custo benefício.

Na Ciência da Computação, o *benchmarking* é utilizado amplamente na avaliação de sistemas (JR, 1971) citado em (QUEIROZ et al., 2013). O *benchmarking* pode ser utilizado tanto para testes de desempenho de *hardware* quanto em *software*. Muitas áreas da Ciência da Computação utilizam o *benchmarking* para suas comparações, como por exemplo: banco de dados, sistemas operacionais, redes de computadores, sistemas distribuídos e muitas outras. Dentre os principais *benchmarks* para avaliar sistemas de gerenciamento de bancos de dados (SGBD), podemos citar o TPC-B. Criado em 1990 com a finalidade de medir a quantidade de transações por minuto que um SGBD era capaz de processar (COUNCIL, 1990) em (QUEIROZ et al., 2013). No processo, uma base de dados é utilizada com um conjunto bem definido de instruções. Cada sistema executa essa mesma sequência de tarefas e os melhores são aqueles com desempenho superior.

O *benchmarking* aplicado aos algoritmos, programas e similares, possuem características de um *benchmarking* genérico, na medida em que se deseja medir os diferentes resultados obtidos na solução de uma mesma tarefa, apenas mudando a forma de solução.

Atualmente o *benchmarking* é uma técnica amplamente aceita na comunidade científica, e se tornou popular na avaliação de resultados. Sua simplicidade e praticidade, foram os principais fatores que impulsionaram a ampla utilização em diversos trabalhos.

2.8 Benchmarking em Mineração de Dados

Na mineração de dados, diferentes técnicas podem ser utilizadas para solucionar um mesmo problema. Diferentes algoritmos possuem melhor performance mediante um determinado tipo de problema apresentado. O presente trabalho visa por meio de um experimento, aplicar a técnica de *benchmarking* no domínio da classificação dos dados de modo a aumentar a eficiência dos resultados obtidos, na medida que avalia o processo de acordo com critérios bem definidos. O *benchmarking* em mineração é possível ser realizado partindo da escolha de medidas que sejam sensíveis ao comportamento de um algoritmo. A seleção das métricas a serem comparadas permite fazer uma avaliação precisa sobre o comportamento de diferentes algoritmos. Conforme os dados classificados podemos ter diferentes tipos de classificadores. Para o experimento apresentado utilizou-se classificadores binários devido ao fato da classificação só possuir duas classes (0 ou 1).

Conforme SOKOLOVA, LAPALME(2009), as principais métricas que devem ser consideradas para um classificador binário na análise de desempenho se baseiam na matriz de confusão. São elas: exatidão (*accuracy*), precisão (*precision*), taxa de verdadeiros positivos(*TPR*), taxa de falsos positivos(*FPR*), taxa de verdadeiros negativos(*TNR*), taxa de falsos negativos(*FNR*), curva ROC e F-Measure. Todas essas medidas de desempenho estão disponíveis para uso através da ferramenta WEKA e serão utilizadas no experimento detalhado no Capítulo 3.

3 *Benchmarking* em mineração de dados no WEKA

Nesse capítulo será realizado um experimento prático utilizando a ferramenta WEKA, através de sua aplicação *WEKA Experimenter*. Nela serão realizadas avaliações sobre o desempenho de diferentes algoritmos de classificação para mineração de dados, utilizando a mesma base de dados. Os testes serão realizados para a base de dados *Pima Indians Diabetes Database* (SIGILLITO, 1990), distribuída com o WEKA e melhor detalhada na Seção 3.4. Ao final serão avaliadas as métricas com maior relevância e exibição dos dados estatísticos gerados por meio da ferramenta.

3.1 A ferramenta WEKA

Inúmeras ferramentas já foram criadas com o intuito de realizar a mineração de dados. Algumas proprietárias e outras de código aberto. Dentre as ferramentas de código aberto as mais populares são Weka, R e Orange. Para esse trabalho será utilizado a ferramenta WEKA.

Criado em 1993 com a ajuda do governo Neozelandês e continuamente aprimorado até os dias atuais, o ambiente WEKA possui diversas funcionalidades que auxiliam na análise de dados. Pode ser utilizado em linha de comando, pela instanciação de suas bibliotecas ou pela sua interface gráfica. O uso pela interface gráfica é rico em funcionalidades e de fácil adaptação ao usuário.

O ambiente WEKA possui vasta documentação disponível através de seu site oficial além de uma comunidade participativa. Possui interface gráfica bem elaborada e de fácil entendimento. Para o usuário aprender como operar o ambiente, existem diversos recursos para aprendizado recomendados através do site oficial. O site possui um *link* para a WIKI(<http://weka.wikispaces.com/>) onde são detalhados os recursos e funcionalidades da ferramenta, além de permitir que os usuários colaborem uns com os outros.

Para esse trabalho foi utilizado a versão 3.8.2 do WEKA e, como referência de acordo com a recomendação da documentação, utilizou-se o apêndice online: "*Data Mining: Practical Machine Learning Tools and Techniques*", em sua quarta edição (WITTEN et al., 2016).

3.2 Benchmarking no ambiente WEKA

Conforme citado no Capítulo 3, o *benchmarking* pode ser utilizado como técnica de comparação entre diferentes formas de solucionar um problema. Nesse trabalho será utilizado a ferramenta WEKA, através de seu módulo de *benchmarking*, WEKA Experimenter (WITTEN et al., 2016), para comparar diferentes técnicas de classificação para a base de dados escolhida (citada na seção 3.4).

3.2.1 Formatos de arquivos

O formato de arquivo nativo do WEKA é o ARFF, que nada mais é que um arquivo ASCII com parâmetros definidos e dados de instâncias separados por vírgula. Por convenção utiliza-se o último atributo do arquivo como classe, porém esse dado pode ser alterado livremente.

Figura 2 – Diretivas do arquivo ARFF da base: diabetes.arff

```
@relation pima_diabetes
@attribute 'preg' numeric
@attribute 'plas' numeric
@attribute 'pres' numeric
@attribute 'skin' numeric
@attribute 'insu' numeric
@attribute 'mass' numeric
@attribute 'pedi' numeric
@attribute 'age' numeric
@attribute 'class' { tested_negative, tested_positive}
@data
6,148,72,35,0,33.6,0.627,50,tested_positive
1,85,66,29,0,26.6,0.351,31,tested_negative
8,183,64,0,0,23.3,0.672,32,tested_positive
1,89,66,23,94,28.1,0.167,21,tested_negative
0,137,40,35,168,43.1,2.288,33,tested_positive
```

Fonte: elaborado pelo autor

Conforme a Figura 2, um arquivo ARFF possui algumas diretivas que servem como definição para os dados que seguem. No início do arquivo, a diretiva **@relation** define qual o nome do conjunto de dados apresentado. Em seguida, os atributos da base são definidos através das diretivas **@attribute** com o rótulo da coluna entre aspas simples e o tipo do dado. A diretiva **@data** marca o início dos dados efetivamente, com os valores separados por vírgula.

3.3 Base de dados adotada

Para realização dos testes foi escolhido uma das bases de dados presente no pacote do WEKA. A base *PIMA Indians* (SIGILLITO, 1990) foi criada a partir do levantamento de dados de pessoas do sexo feminino, pertencentes a tribo dos índios Pima, uma tribo de nativos americanos que vive no Arizona. Essa tribo viveu por um longo período com uma dieta pobre em carboidratos, porém durante os anos que antecederam a coleta de dados, verificou-se uma mudança em sua dieta, e um significativo aumento da incidência de diabetes tipo 2 nessa população (SIGILLITO, 1990) . A base é constituída de 768 instâncias, composta de oito atributos de entrada e um atributo de saída (classe).

Os atributos da base estão descritos como:

- preg - Número de vezes que ficou grávida;
- plas - Concentração plasmática de glicose de duas horas através de um teste oral de tolerância à glicose;
- pres - Pressão sanguínea diastólica (mm Hg);
- skin - Espessura da pele na região do tríceps (mm);
- insu - Insulina medida com 2 horas (μ U/ml) ;
- mass - Índice de massa corpórea (peso em kg / altura em m);
- pedi - Função que representa a probabilidade de desenvolver a doença baseado no histórico familiar;
- age - Idade(anos);
- class - Classe (0 - não apresenta diabetes, 1 - com diabetes)

Todos os atributos são numéricos e a classe é uma lista de dois valores, 0 para teste negativo e 1 para teste positivo em diabetes tipo 2.

3.4 Classificadores utilizados

O WEKA disponibiliza diversos algoritmos de classificação para testes. Dentre estes, foram escolhidos os 6 algoritmos. Um baseado em árvore de decisão (J48), dois baseados em regra (OneR, Jrip), um baseado em instância (Ibk), um em rede neural (Multilayer perceptron) e um probabilístico (NaiveBayes).

3.4.1 Descrição dos algoritmos utilizados

- **NaiveBayes** - Algoritmo essencialmente estatístico baseado no teorema de Bayes. Utiliza previsões probabilísticas para uma classificação, exemplo: o paciente tem 70% de chance de possuir a doença. A combinação de múltiplas hipóteses ponderadas por suas probabilidades irão compor o modelo de aprendizado. Possui a característica de possuir uma forte independência entre os preditores (JOHN; LANGLEY, 1995).
- **J48** - Reimplementação do algoritmo C4.5. Utiliza árvores de decisão para criar o modelo. Ao contrário do C4.5 o J4.8 tem poda com erro reduzido. Possui opção de ajustar o limiar de confiança da poda através da própria interface (WITTEN et al., 2016).
- **OneR** - Abreviação para *One Rule* ou uma regra. Um algoritmo bem simples e preciso. Em resumo o algoritmo monta uma tabela com as regras mais frequentes e suas previsões, a regra que apresentar menor erro na previsão é selecionada. Após definida a regra, todas as previsões são feitas baseando-se nela (HOLTE, 1993).
- **JRip** - Algoritmo baseado em regras. Baseia-se no algoritmo IREP (COHEN, 1995). Possui a característica de aprendizado de regras proposicional (SHAHZAD; ASAD; KHAN, 2013)
- **Ibk** - Algoritmo de aprendizagem preguiçosa, ou aprendizado baseado em instâncias (IBL-*Instance Based Learning*). Derivado do método de aprendizado K-NN, o mesmo utiliza uma função de similaridade baseada na distância euclidiana dos itens (AHA; KIBLER; ALBERT, 1991).
- **Multilayer Perceptron** - Rede neural que utiliza retropropagação do erro (*back-propagation*) para treinar a rede. Determina a classe de uma instância usando uma combinação linear dos atributos. Funciona melhor com atributos numéricos. Todos os nós são sigmóides (exceto para classes não numéricas), os nós representam os neurônios e são ativados por meio de funções logísticas (HAYKIN, 1994).

3.5 Realizando os testes no WEKA

Na Figura 3 tem-se a tela de apresentação do ambiente WEKA. A mesma constitui basicamente de um menu seletor de aplicações. O ambiente é composto de 5 aplicações: *Explorer*, *Experimenter*, *KnowLedgeFlow*, *Workbench*, *Simple CLI*.

Figura 3 – Tela Inicial do ambiente WEKA

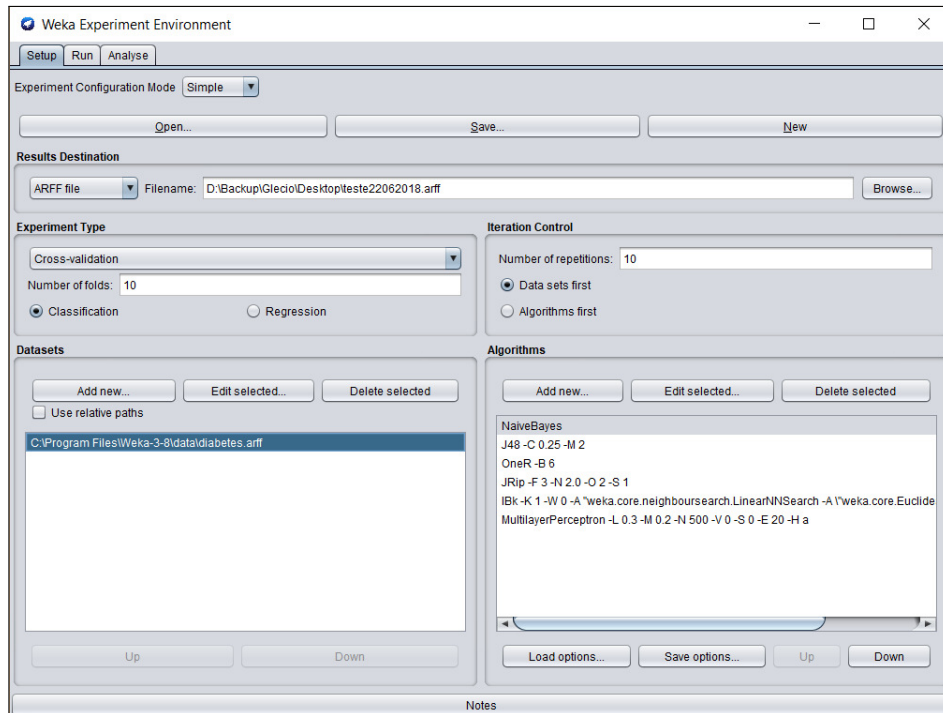


Fonte: elaborado pelo autor

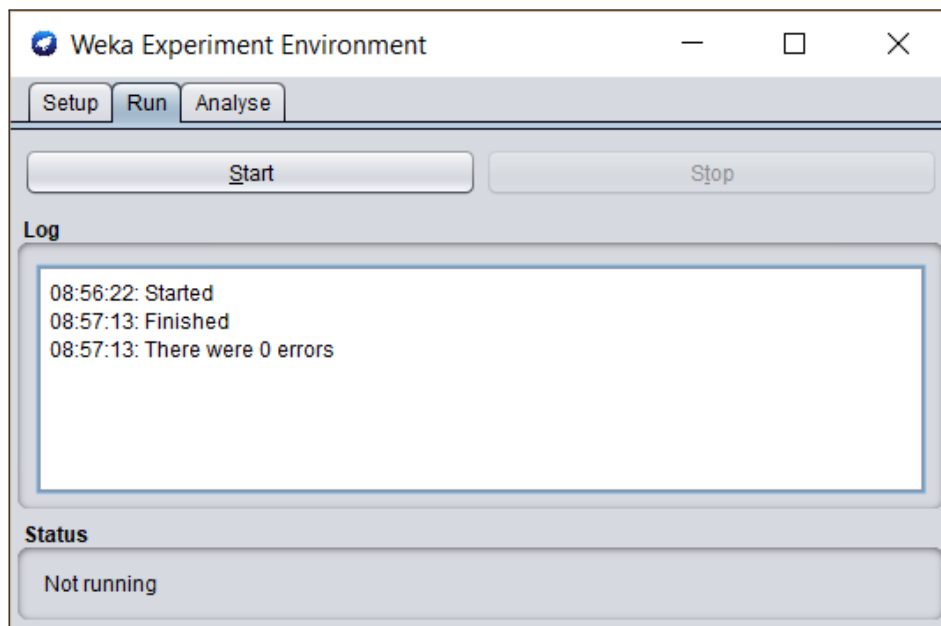
O ambiente WEKA *Experiment Enviroment*(WEE) do WEKA, permite testar múltiplos algoritmos de classificação simultaneamente e gerar estatísticas de desempenho entre os mesmos. O experimenter possui três painéis: *Setup*, *Run* e *Analyse*.

- **Setup:** tela inicial do *Experimenter*, onde é feita a preparação dos testes. Nessa tela pode ser criado um novo experimento ou acessado um experimento realizado anteriormente. Os experimentos salvos possuem a extensão **.exp**, e é nesse arquivos que ficarão armazenadas todas as opções dos testes previamente escolhidas. Para a saída dos resultados podem ser escolhidos três formatos de arquivo: CSV, ARFF ou base de dados JDBC.

Para o teste com a base de dados escolhida (SIGILLITO, 1990) foram utilizados os parâmetros, conforme a Figura 4 . Foram utilizados seis (6) algoritmos de classificação presentes no ambiente: **NaiveBayes**, **J48**, **OneR**, **JRip**, **IBk** e **Multilayer Perceptron**

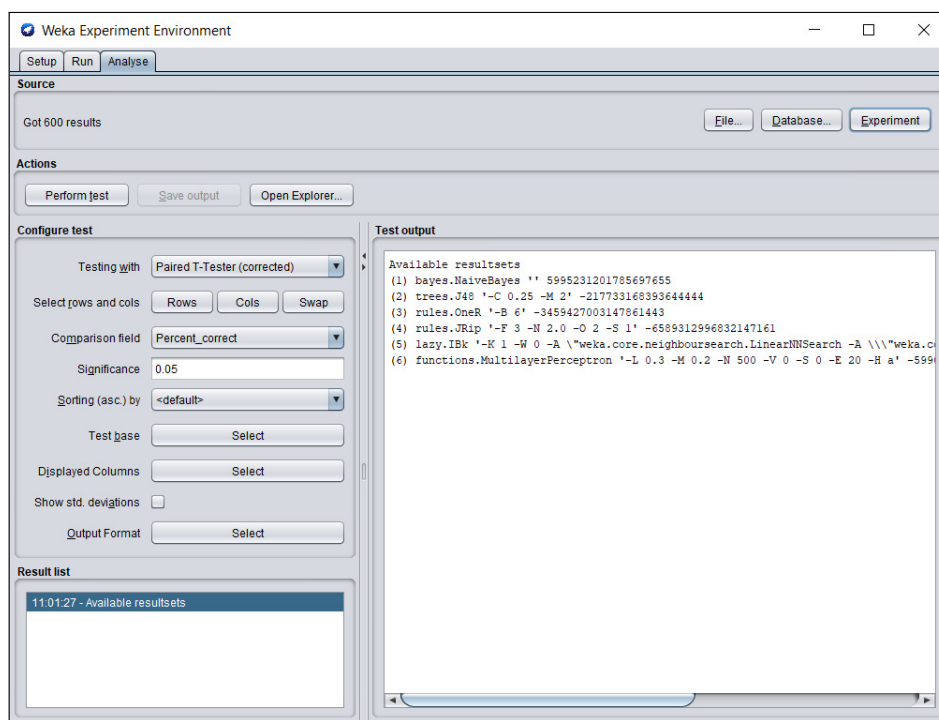
Figura 4 – WEE - *Setup*

- **Run:** local onde é realizado a execução dos testes. Um espaço para exibição de mensagens que é responsável por mostrar detalhes do teste sendo executado, como tempo de início e fim, mensagens de erro e mensagens relacionadas. Na Figura 5 temos a saída após execução de um teste.

Figura 5 – WEE - *Run*

Fonte: elaborado pelo autor

- **Analyse:** os dados gerados após rodar o experimento são carregados nesse módulo. São aceitos arquivos de saída em formato CSV ou ARFF. Um arquivo de saída é composto por um ou mais *resultsets*, sendo um para cada algoritmo selecionado no módulo *Setup*. Na Figura 6 os resultados são apresentados após execução de um teste, além de todas as opções selecionadas para análise.

Figura 6 – WEE - *Analyse*

Fonte: elaborado pelo autor

3.6 Resultados

Para o teste, o WEKA avaliou 700 instâncias, dentre estas 691,2 foram instâncias para treino e 76,8 instâncias de teste, de acordo com as definições padrão do ambiente através da escolha por validação cruzada. Os resultados obtidos em formatos CSV foram exportados para uma planilha para facilitar a criação gráficos dos principais parâmetros de medição, objetivando uma melhor visão dos diferentes desempenhos. A tabela com o resultado completo de todas as métricas para avaliação disponibilizadas no WEKA é listada no apêndice A.

3.6.1 Principais Métricas analisadas

Devemos levar em consideração na análise de *benchmarking* métricas que podem revelar um diagnóstico preciso sobre o desempenho do algoritmo analisado. Para (SOKOLOVA; LAPALME, 2009), as principais medidas de desempenho a serem avaliadas em

algoritmos de classificação são: exatidão (*accuracy*), precisão (*precision*), taxa de verdadeiros positivos (TPR), taxa de falsos positivos (FPR), taxa de verdadeiros negativos (TNR), taxa de falsos negativos (FNR), curvas ROC e F-Measure. Estas métricas são calculadas com base na matriz de confusão e serão definidas a seguir.

3.6.1.1 Matriz de Confusão

A matriz de confusão é uma matriz que auxilia a análise de performance, tipicamente utilizada em aprendizado de máquina. Nela são representadas as quantidade de classificações corretas e incorretas realizadas em um experimento. A diagonal principal da matriz representa as entidades que foram corretamente classificadas. Conforme a Figura 7, temos que os valores que estão fora da diagonal, representam instâncias que foram classificadas incorretamente (falsos positivos ou falsos negativos) (FAWCETT, 2004).

Figura 7 – Matriz de Confusão para duas classes

		Observado	
		VERDADEIRO	FALSO
CLASSIFICADO COMO	VERDADEIRO	VERDADEIROS POSITIVO(TP)	FALSO POSITIVO(FP)
	FALSO	FALSO NEGATIVO(FP)	VERDADEIROS NEGATIVOS(TN)

Fonte: elaborado pelo autor

É uma matriz quadrada de dimensões de acordo com a quantidade de classes. Na Figura 7, temos uma matriz de confusão para duas classes. Os itens fora da diagonal principal representam instâncias classificadas incorretamente.

3.6.1.2 Instâncias classificadas corretamente

As instâncias após classificadas, são divididas em dois conjuntos: classificadas corretamente e classificadas incorretamente. O número de instâncias classificadas corretamente é formado pelo número de verdadeiros positivos (TP - *True Positive*) somado aos verdadeiros negativos (TN - *True negative*), conforme Equação 3.1.

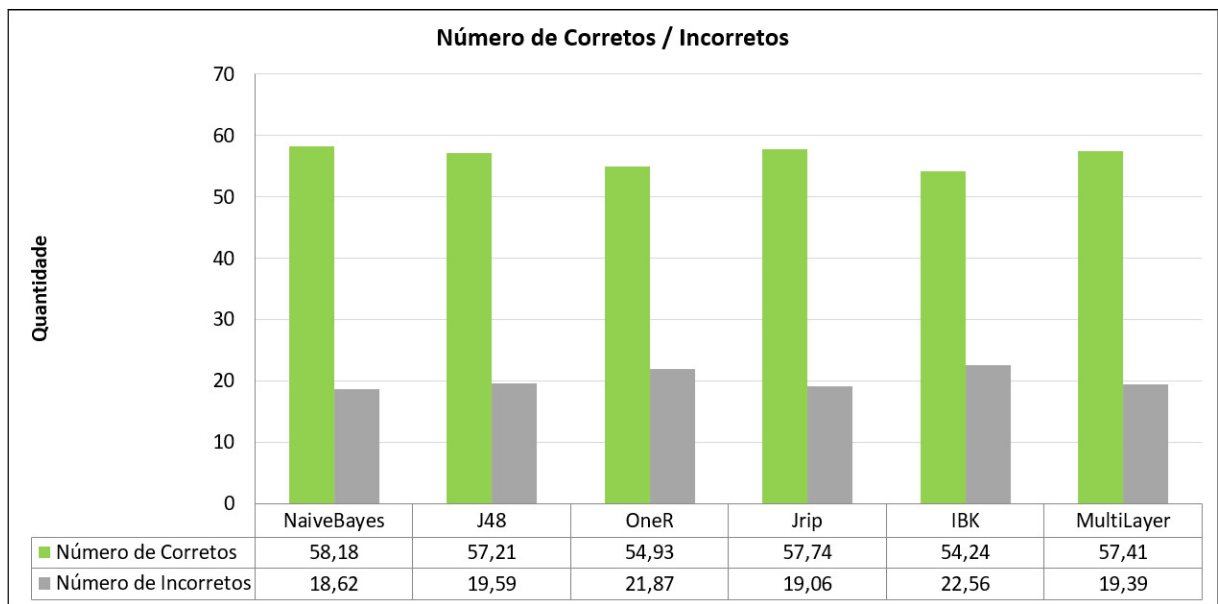
$$Num_Corretos = TP + TN \quad (3.1)$$

3.6.1.3 Instâncias classificadas incorretamente

O total das instâncias que foram classificadas incorretamente é formado pelo número de falsos positivos (FP - *False Positive*) somado aos falsos negativos (FN - *False negative*), conforme Equação 3.2. Na Figura 8 é mostrado o gráfico com o numero de corretos e incorretos.

$$Num_Incorretos = FP + FN \tag{3.2}$$

Figura 8 – Num. Corretos x Incorretos



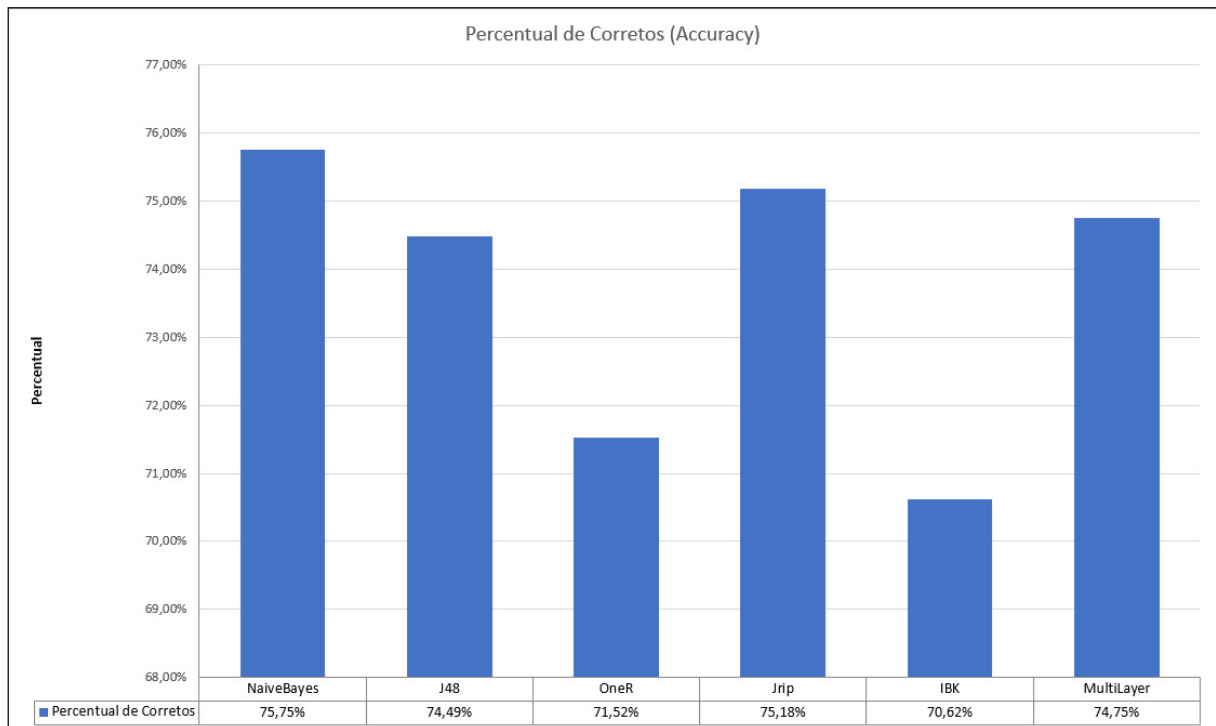
Fonte: elaborado pelo autor

3.6.1.4 Exatidão(Acurácia)

A exatidão é a medida comum entre as métricas. Seu valor define o quão próximo da classificação correta o algoritmo está. Seu cálculo é definido conforme a Equação 3.3 e os valores obtidos são apresentados no gráfico da Figura 9:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.3)$$

Figura 9 – Exatidão (Acurácia)



Fonte: elaborado pelo autor

A exatidão pode ser definida como a razão entre o número de classificações realizadas corretamente ($TP + TN$) pelo numero total de classificações realizadas ($TP + TN + FP + FN$). Na Figura 9 é apresentado um gráfico com os resultados obtidos através do experimento.

3.6.1.5 Taxa de verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos

Taxa de verdadeiros positivos(TPR) ou revocação(*recall*). Equação 3.4:

$$TPR = \frac{TP}{TP + FN} \quad (3.4)$$

Taxa de verdadeiros negativos(TNR) ou precisão(*precision*). Equação 3.5:

$$TNR = \frac{TN}{TN + FP} \quad (3.5)$$

Taxa de Falso Positivos(FPR). Equação 3.6:

$$FPR = \frac{FP}{TN + FP} \quad (3.6)$$

Taxa de Falso Negativos(FNR). Equação 3.7:

$$FNR = \frac{FN}{TP + FN} \quad (3.7)$$

Na Figura 10 são apresentados os resultados obtidos através do experimento.

Figura 10 – FPR e FNR



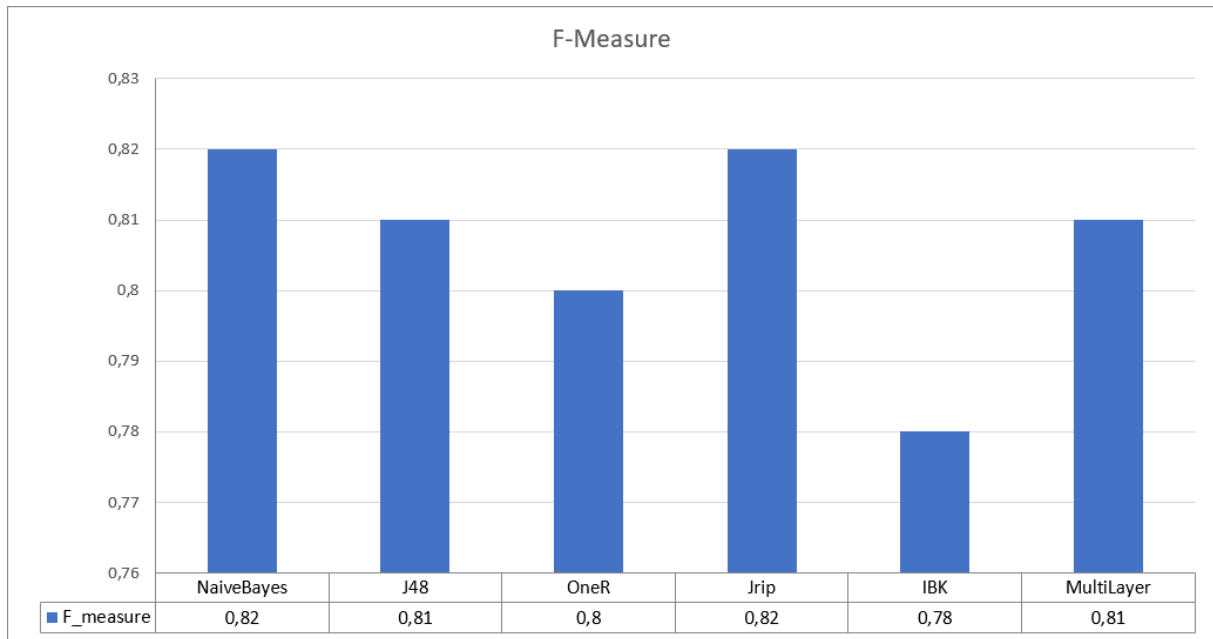
Fonte: elaborado pelo autor

F-Measure:

$$F - Measure = \frac{2}{\frac{1}{TNR} + \frac{1}{TPR}} x' \quad (3.8)$$

F-measure, F1 score ou F score(Equação 3.8), é um medida de precisão, definida como a média harmônica ponderada entre a precisão e a revocação (POWERS, 2011). Na Figura 11 são apresentados os resultados de **F-measure** obtidos através do experimento.

Figura 11 – F-Measure



Fonte: elaborado pelo autor

3.6.1.6 Curvas ROC (*Receiver Operating Characteristic Curve*)

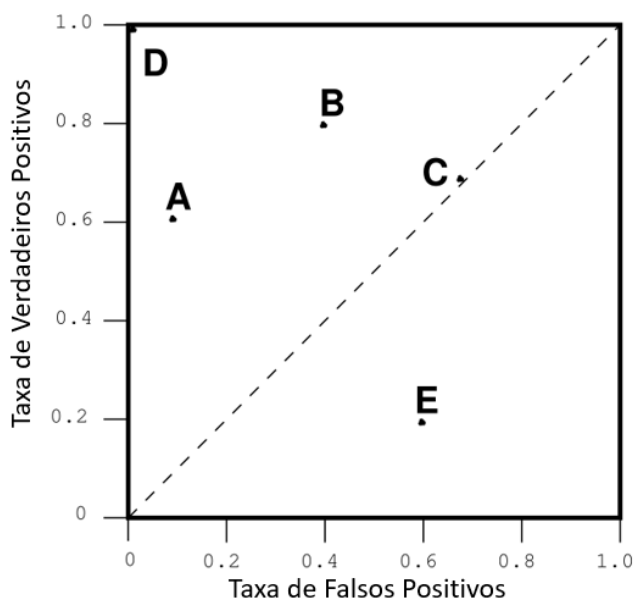
As curvas ROC são plotagens baseadas nos valores extraídos da matriz de confusão. No eixo vertical a taxa de verdadeiros positivos (TPR) e no horizontal a taxa de falsos positivos (FPR). Diferentes curvas delimitam as regiões de corte para as quais, pode-se classificar o comportamento de um algoritmo (FAWCETT, 2004).

Através da plotagem de ROC podemos tirar algumas conclusões:

- Mostra uma diferença entre sensibilidade e especificidade, qualquer aumento na sensibilidade é seguido de um decremento na especificidade e vice-versa;
- Quanto mais próximo um ponto está do canto superior esquerdo melhor é a performance do algoritmo;
- Quanto mais próximo da linha diagonal que divide o gráfico menor é a performance de um algoritmo.

Para algoritmos com valores discretos de classificações, é feita a representação em um gráfico ROC de duas dimensões, onde no eixo X está a taxa de falsos positivos e no eixo Y a taxa de verdadeiros positivos (FAWCETT, 2004). Na Figura 12, temos 5 classificadores, representados pelos pontos A a E. Um ponto na coordenada (0,0) representa um algoritmo cujo número de falsos positivos é zero, porém também não possui nenhum verdadeiro positivo. O ponto D, (1,1) representa a performance ideal, ou seja, sem nenhum

Figura 12 – ROC



Fonte: Adaptado de (FAWCETT, 2004)

falso positivo e todos verdadeiros positivos. Os pontos na linha tracejada representam comportamentos randômicos, uma vez que apresentam igual número de verdadeiros e falsos positivos. Abaixo dessa linha estão os pontos representando comportamentos ineficientes.

A área sob a curva ROC (AUC) é a área delimitada pela curva obtida através da plotagem dos valores gerados para cada instância classificada por um algoritmo. Quanto mais próximo de 1 esse valor, melhor desempenho o algoritmo possui. Valores abaixo de 0.5 representam um algoritmo de desempenho ineficientes.

Os valores de área sob a curva ROC obtidos a partir do experimento estão listados na Tabela 1:

Algoritmo	AUROC
NaiveBayes	0,82
MultiLayer	0,8
J48	0,75
Jrip	0,72
IBK	0,67
OneR	0,65

Tabela 1 – Valores obtidos de AUROC

Na Tabela 2 é apresentado um resumo das métricas obtidas para cada algoritmo ao final da avaliação:

Métrica	NaiveBayes	J48	OneR	Jrip	IBK	MultiLayer	Melhor Desempenho
Número de Incorretos	18,62	19,59	21,87	19,06	22,56	19,39	NaiveBayes
Número de Corretos	58,18	57,21	54,93	57,74	54,24	57,41	NaiveBayes
Taxa de Corretos (ACC)	75,75%	74,49%	71,52%	75,18%	70,62%	74,75%	NaiveBayes
Taxa de Incorretos	24,25%	25,51%	28,48%	24,82%	29,38%	25,25%	NaiveBayes
Taxa de Verdadeiros Positivos (TPR)	0,84	0,82	0,87	0,85	0,8	0,84	OneR
Taxa de Falsos Positivos (FPR)	0,4	0,4	0,57	0,42	0,46	0,42	NaiveBayes / J48
Taxa de Verdadeiros Negativos (TNR)	0,6	0,6	0,43	0,58	0,54	0,58	NaiveBayes / J48
Taxa de Falsos Negativos(FNR)	0,16	0,18	0,13	0,15	0,2	0,16	OneR
Area abaixo da curva ROC (AUROC)	0,82	0,75	0,65	0,72	0,67	0,8	NaiveBayes
F-Measure	0,82	0,81	0,8	0,82	0,78	0,79	NaiveBayes / Jrip

Tabela 2 – Resultados Obtidos

Conforme os dados obtidos fica evidenciado que o algoritmo **Naive Bayes**, cujo critério de classificação é essencialmente estatístico, apresentou melhores resultados na maioria das métricas para a base escolhida. Para o problema em questão, o mesmo seria a melhor escolha para predição baseado nas métricas avaliadas. O algoritmo NaiveBayes apesar de sua simplicidade, possui ótimas performances na maioria dos cenários. Uma das explicações é que o algoritmo tenta alcançar a melhor precisão possível quando os atributos são independentes. Frank, Trigg, Holmes, and Witten citado em (ASHARI; PARYUDI; TJOA, 2013) atribuem a boa performance do algoritmo à sua função de perda zero-um, que define o erro como o número de predições incorretas.

4 Conclusão

A análise de grandes quantidades de dados realizada durante a descoberta de conhecimento passa necessariamente pela mineração de dados. Os classificadores desempenham papel fundamental na mineração, e seu desempenho afeta sensivelmente o resultado final dessa operação. O trabalho apresentou de maneira objetiva, um experimento visando a avaliação de desempenho entre diferentes classificadores para a mineração de dados. A avaliação seguiu as diretrizes de avaliação do *benchmarking*, estabelecendo critérios de avaliação dentro de regras bem definidas.

Os conceitos de *benchmarking* introduzidos no Capítulo 2 serviram como base na elaboração de uma avaliação padronizada e criteriosa, visando um melhor entendimento dos procedimentos de avaliação. A contribuição que o desenvolvimento de tal técnica trouxe, permitiu a ocorrência de diversos avanços em múltiplas áreas. Com o uso do *benchmarking*, torna-se possível a avaliação de um cenário com base em procedimentos comprovadamente eficazes. Esse trabalho utilizou-se dos princípios dessa técnica para avaliar os resultados obtidos para cada algoritmo avaliado, apresentando métricas de grande relevância.

A ferramenta WEKA proporciona vários recursos no trabalho com mineração de dados. É uma ferramenta completa para análise de dados, e tem sido aprimorada constantemente além de poder contar com uma boa documentação, vasta literatura e comunidade ativa. Através desse experimento, foi possível mostrar as funcionalidades existentes na ferramenta WEKA para a análise de dados, com foco nas medições de desempenho e avaliações de algoritmos. O WEE (*WEKA Experimenter Environment*) possibilita configurar testes completos para vários algoritmos. Através dos resultados gerados no WEE, é possível ter uma visão ampla sobre o comportamento dos algoritmos testados, bem como servir de base na escolha de melhores abordagens para um determinado problema.

A mineração de dados está em pleno uso no cotidiano, seu desempenho é meta para muitos estudos e pesquisas. Com este experimento é realizado o *benchmarking* entre seis algoritmos sendo possível mensurar, através das métricas obtidas com o WEE, o desempenho de cada um deles para o problema apresentado. Foram utilizadas as opções por padrão de cada algoritmo, porém, é possível o ajuste de diversos modos de teste possibilitando configurações variadas. Cada algoritmo pode ser configurado individualmente, através de suas opções de configuração. Tal funcionalidade pode ser explorada em trabalho futuro, em que poderia ser feitos testes com diferentes configurações de algoritmos.

As principais métricas estatísticas apresentadas no Capítulo 3 são de fácil entendimento e calculadas de maneira automática com o auxílio da ferramenta. Com a curva ROC é possível ter uma análise gráfica sobre o comportamento de cada algoritmo. As medidas de precisão e recall orientam sobre como se comporta cada classificador mediante um problema. Diversas outras métricas podem ser analisadas dependendo do objetivo, todas elas são apresentadas no Apêndice A. Dados estatísticos apresentados pelo experimento permitiram definir qual melhor estratégia a ser adotada para um problema baseando-se nas métricas de desempenho. Ficou evidenciado que para a base de dados escolhida (SILLITO, 1990) o algoritmo NaiveBayes, que é um algoritmo essencialmente estatístico, obteve melhores resultados de desempenho.

O *benchmarking* em algoritmos, assim como a própria mineração de dados, são áreas com vastas possibilidades de pesquisa. O experimento mostrado abre perspectivas para novos trabalhos completos. Como proposta para trabalhos futuros podemos citar:

- Expandir a abrangência dos testes realizados, uma vez que, é possível o uso das bibliotecas do WEKA programaticamente através de instânciação por uma linguagem de programação. Isso permitiria a criação de novos recursos, como ambientes personalizados pra testes, geração de relatórios e integração com outras linguagens. Esses ambientes podem contar com opções de testes programáveis de acordo com a necessidade. Um bom exemplo seria a aplicação dos testes para vários *seeds* diferentes em cada algoritmo. Desse modo, será possível a obtenção de resultados com valores ainda mais randômicos e em sequências diferentes para análise. Com o uso de bibliotecas externas em Java é possível gerar relatórios e gráficos completos a partir dos resultados obtidos nos testes, possibilitando uma análise completa e bem detalhada dos resultados;
- Comparar algoritmos em bases de dados com características diferentes cada uma. Comparar o desempenho para problemas com diferentes quantidades de classes. Algoritmos estatísticos podem apresentar melhores desempenhos em algumas bases de dados do que em outras, e essa correlação, pode ser estudada por meio de experimentos através da ferramenta WEKA;

Dentre as principais limitações encontradas durante o desenvolvimento do trabalho, a principal foi a deficiência na literatura relacionada à *benchmarking* para computação. Boa parte da literatura existente se aplica ao *benchmarking* como ferramenta de gestão e qualidade nas corporações, devido principalmente ao fato de que sua origem ter ocorrido por nesse ambiente.

Apêndices

APÊNDICE A – Dados Brutos

Dado	NaiveBayes	J48	OneR	Jrip	IBK	MultiLayer
Num_of_training_instances	691,2	691,2	691,2	691,2	691,2	691,2
Num_of_testing_instances	76,8	76,8	76,8	76,8	76,8	76,8
Num_incorrect	18,62	19,59	21,87	19,06	22,56	19,39
Num_correct	58,18	57,21	54,93	57,74	54,24	57,41
Num_unclassified	0	0	0	0	0	0
Percent_correct	75,75%	74,49%	71,52%	75,18%	70,62%	74,75%
Percent_incorrect	24,25%	25,51%	28,48%	24,82%	29,38%	25,25%
Percent_unclassified	0	0	0	0	0	0
kappa_statistic	0,45	0,43	0,32	0,43	0,34	0,43
Mean_absolute_error	0,29	0,31	0,28	0,34	0,29	0,3
Root_mean_squared_error	0,42	0,44	0,53	0,43	0,54	0,42
Relative_absolute_error	63,33	67,95	62,65	75,74	64,77	65,39
Root_relative_squared_error	87,99	92,06	111,51	89,67	113,18	88,67
SF_prior_entropy	71,67	71,67	71,67	71,67	71,67	71,67
SF_scheme_entropy	68,41	3488,85	23488,38	61,41	213,01	66,59
SF_entropy_gain	3,26	-3417,19	-23416,71	10,26	-141,35	5,08
SF_mean_prior_entropy	0,93	0,93	0,93	0,93	0,93	0,93
SF_mean_scheme_entropy	0,89	45,45	305,83	0,8	2,77	0,87
SF_mean_entropy_gain	0,04	-44,52	-304,9	0,13	-1,84	0,07
KB_information	26,07	23,24	24,92	19,04	23,38	24,67
KB_mean_information	0,34	0,3	0,32	0,25	0,3	0,32
KB_relative_information	2793,16	2489,41	2669,92	2039,3	2504,96	2643,03
True_positive_rate	0,84	0,82	0,87	0,85	0,8	0,84
Num_true_positives	42,05	41,04	43,38	42,3	39,75	41,78
False_positive_rate	0,4	0,4	0,57	0,42	0,46	0,42
Num_false_positives	10,67	10,63	15,25	11,36	12,31	11,17
True_negative_rate	0,6	0,6	0,43	0,58	0,54	0,58
Num_true_negatives	16,13	16,17	11,55	15,44	14,49	15,63
False_negative_rate	0,16	0,18	0,13	0,15	0,2	0,16
Num_false_negatives	7,95	8,96	6,62	7,7	10,25	8,22
IR_precision	0,8	0,8	0,74	0,79	0,76	0,79
IR_recall	0,84	0,82	0,87	0,85	0,8	0,84
F_measure	0,82	0,81	0,8	0,82	0,78	0,81
Matthews_correlation	0,46	0,44	0,34	0,44	0,34	0,43
Area_under_ROC	0,82	0,75	0,65	0,72	0,67	0,8
Area_under_PRC	0,89	0,82	0,73	0,78	0,74	0,87
Weighted_avg_true_positive_rate	0,76	0,74	0,72	0,75	0,71	0,75
Weighted_avg_false_positive_rate	0,31	0,32	0,42	0,33	0,37	0,33
Weighted_avg_true_negative_rate	0,69	0,68	0,58	0,67	0,63	0,67
Weighted_avg_false_negative_rate	0,24	0,26	0,28	0,25	0,29	0,25
Weighted_avg_IR_precision	0,76	0,75	0,71	0,75	0,7	0,75
Weighted_avg_IR_recall	0,76	0,74	0,72	0,75	0,71	0,75
Weighted_avg_F_measure	0,75	0,74	0,7	0,75	0,7	0,74
Weighted_avg_matthews_correlation	0,46	0,44	0,34	0,44	0,34	0,43
Weighted_avg_area_under_ROC	0,82	0,75	0,65	0,72	0,67	0,8
Weighted_avg_area_under_PRC	0,82	0,74	0,64	0,7	0,65	0,81
Unweighted_macro_avg_F_measure	0,73	0,71	0,65	0,72	0,67	0,71
Unweighted_micro_avg_F_measure	0,76	0,74	0,72	0,75	0,71	0,75
Elapsed_Time_training	0	0	0	0,02	0	0,44
Elapsed_Time_testing	0	0	0	0	0	0
UserCPU_Time_training	0	0	0	0,02	0	0,44
UserCPU_Time_testing	0	0	0	0	0	0
UserCPU_Time_millis_training	0,63	3,75	0,63	17,66	0,31	441,87
UserCPU_Time_millis_testing	0	0	0	0	3,13	0
Serialized_Model_Size	3398	10862,8	1339,2	5729,13	73016,2	13148
Serialized_Train_Set_Size	71267,2	71267,2	71267,2	71267,2	71267,2	71267,2
Serialized_Test_Set_Size	9212,8	9212,8	9212,8	9212,8	9212,8	9212,8
Coverage_of_Test_Cases_By_Regions	96,98	95,22	71,52	100	70,62	96,3
Size_of_Predicted_Regions	83,8	88,9	50	99,99	50	84,82

Referências

- ABBAS, A. *The characteristics of successful benchmarking implementation: guidelines for a national strategy for promoting benchmarking*. Tese (Doutorado) — Massey University, 2014. Citado 2 vezes nas páginas 13 e 16.
- AHA, D. W.; KIBLER, D.; ALBERT, M. K. Instance-based learning algorithms. *Machine learning*, Springer, v. 6, n. 1, p. 37–66, 1991. Citado na página 23.
- AMARAL, F. *Aprenda Mineração de Dados: Teoria e prática*. ALTA BOOKS, 2016. (Autoria Nacional). ISBN 9788576089889. Disponível em: <<https://books.google.com.br/books?id=qZlgDQAAQBAJ>>. Citado 2 vezes nas páginas 9 e 12.
- ASHARI, A.; PARYUDI, I.; TJOA, A. M. Performance comparison between naïve bayes, decision tree and k-nearest neighbor in searching alternative design in an energy simulation tool. *International Journal of Advanced Computer Science and Applications (IJACSA)*, Citeseer, v. 4, n. 11, 2013. Citado na página 33.
- BPIR. *What is Benchmarking?* 2018. Disponível em: <<http://www.bpir.com/benchmarking-what-is-benchmarking-bpir.com/menu-id-69.html>>. Citado na página 17.
- CAMBRIDGE, D. *Cambridge Online Dictionary*. [S.l.]: Pesquisado através de <https://dictionary.cambridge.org/pt/dicionario/ingles/benchmarking>, 2018. Acesso em: 10/06/2018. Citado na página 13.
- CAMP, R. C. Benchmarking: the search for industry best practices that lead to superior performance. In: *Benchmarking: the search for industry best practices that lead to superior performance*. [S.l.]: ASQC/Quality Resources, 1989. Citado 2 vezes nas páginas 13 e 14.
- CIFERRI, R. R. et al. Um benchmark voltado a análise de desempenho de sistemas de informações geográficas. [sn], 1995. Citado na página 18.
- COHEN, W. W. Fast effective rule induction. In: *Machine Learning Proceedings 1995*. [S.l.]: Elsevier, 1995. p. 115–123. Citado na página 23.
- COUNCIL, T. P. P. ‘tpc benchmark b. *Standard Specification, Waterside Associates, Fremont, CA*, 1990. Citado na página 18.
- DEWITT, D. J. Benchmarking database systems: Past efforts and future diretions. *IEEE Database Eng. Bull.*, v. 8, n. 1, p. 2–9, 1985. Citado na página 18.
- FAWCETT, T. Roc graphs: Notes and practical considerations for researchers. *Machine learning*, v. 31, n. 1, p. 1–38, 2004. Citado 3 vezes nas páginas 27, 31 e 32.
- HAYKIN, S. *Neural networks: a comprehensive foundation*. [S.l.]: Prentice Hall PTR, 1994. Citado na página 23.
- HOLTE, R. C. Very simple classification rules perform well on most commonly used datasets. *Machine learning*, Springer, v. 11, n. 1, p. 63–90, 1993. Citado na página 23.

- JOHN, G. H.; LANGLEY, P. Estimating continuous distributions in bayesian classifiers. In: MORGAN KAUFMANN PUBLISHERS INC. *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. [S.l.], 1995. p. 338–345. Citado na página 23.
- JR, H. L. Performance evaluation and monitoring. *ACM Computing Surveys (CSUR)*, ACM, v. 3, n. 3, p. 79–91, 1971. Citado na página 18.
- KYRÖ, P. Revising the concept and forms of benchmarking. *Benchmarking: An International Journal*, MCB UP Ltd, v. 10, n. 3, p. 210–225, 2003. Citado 2 vezes nas páginas 15 e 16.
- MARTINS, S. G.; SANTOS, A. S. d.; CARVALHO, L. M. O benchmarking e sua aplicabilidade em unidades de informação: uma abordagem reflexiva. 2010. Citado 2 vezes nas páginas 13 e 17.
- POWERS, D. M. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *Bioinfo Publications*, 2011. Citado na página 30.
- QUEIROZ, L. T. et al. Um benchmark para avaliação de técnicas de busca no contexto de análise de mutantes sql. Universidade Federal de Goiás, 2013. Citado 2 vezes nas páginas 9 e 18.
- SHAHZAD, W.; ASAD, S.; KHAN, M. A. Feature subset selection using association rule mining and jrip classifier. *International Journal of Physical Sciences*, Academic Journals, v. 8, n. 18, p. 885–896, 2013. Citado na página 23.
- SIGILLITO, V. Pima indians diabetes database. *UCI Machine Learning Repository* [[http://archive.ics.ci.edu/ml/datasets/Pima Indians Diabetes](http://archive.ics.ci.edu/ml/datasets/Pima%20Indians%20Diabetes)]. *Phoenix, AZ: National Institute of Diabetes and Digestive and Kidney Diseases*, 1990. Citado 4 vezes nas páginas 20, 22, 24 e 35.
- SOKOLOVA, M.; LAPALME, G. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, Elsevier, v. 45, n. 4, p. 427–437, 2009. Citado na página 26.
- WATSON, G. A perspective on benchmarking. *organization*, v. 1, p. 1, 1994. Citado na página 14.
- WATSON, G. H. *Strategic benchmarking: How to rate your company's performance against the world's best*. [S.l.]: Wiley, 1993. Citado na página 15.
- WEBSTER, M. *Merriam-Webster*. [S.l.]: Pesquisado através de <http://www.merriam-webster.com>, 1828. Acesso em: 10/06/2018. Citado na página 13.
- WITTEN, I. H. et al. *Data Mining: Practical machine learning tools and techniques*. [S.l.]: Morgan Kaufmann, 2016. Citado 3 vezes nas páginas 20, 21 e 23.