UNIVERSIDADE FEDERAL DO MARANHÃO

Programa de Pós-Graduação em Ciência da Computação

Alexandre Ronald de Araujo Oliveira

Processo de Mineração de Dados Orientado ao Usuário Final: O ambiente LAWSMiner

São Luís
2020

Alexandre Ronald de Araujo Oliveira

Processo de Mineração de Dados Orientado ao Usuário Final: O ambiente LAWSMiner

São Luís

2020

Alexandre Ronald de Araujo Oliveira

Processo de Mineração de Dados Orientado ao Usuário Final: O ambiente LAWSMiner

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFMA, como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação

Universidade Federal do Maranhão – UFMA

Centro de Ciências Exatas e Tecnológicas

Programa de Pós-Graduação em Ciência da Computação

Orientador: Mário Antônio Meireles Teixeira

São Luís

2020

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

Oliveira, Alexandre Ronald de Araujo.
Processo para Mineração de Dados Orientado ao Usuário Final : O Ambiente LAWSMiner / Alexandre Ronald de Araujo Oliveira. – 2020.
85 p.

Orientador(a): Mário Antônio Meireles Teixeira.
Dissertação (Mestrado) – Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luís, 2020.

1. Análise Exploratória de Dados. 2. Aprendizagem de Máquina. 3. Mineração de Dados. I. Teixeira, Mário Antônio Meireles. II. Título.

Alexandre Ronald de Araujo Oliveira

Processo de Mineração de Dados Orientado ao Usuário Final: O ambiente LAWSMiner

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFMA, como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação

Mário Antônio Meireles Teixeira
Dr. em Ciência da Computação - UFMA

Tiago Bonini Borchartt
Dr. em Computação - UFMA

Renato Porfirio Ishii
Dr. em Ciência da Computação - UFMS

São Luís
2020

Dedico este trabalho a minha esposa Adryane e aos meus filhos André e Alan, assim como, aos meus pais Reinaldo e Francisca

Agradecimentos

Ao Prof. Mário Antônio Meireles Teixeira pela dedicação, orientação, apoio incondicional e incentivo durante todo o mestrado.

À Profª Alana de Araújo Oliveira Meireles Teixeira pelo apoio, prestatividade e disponibilidade em discutir sobre os pontos essenciais do projeto.

Aos irmãos em Cristo da Igreja Batista Nacional em Vicente Fialho, em especial aos meus pastores Jave Reis e Lana Reis, pelo suporte espiritual e orações.

Aos colegas do Hospital Universitário da Universidade Federal do Maranhão pelo incentivo e apoio a minha pesquisa, em especial ao Magnífico Reitor Prof. Dr. Natalino Salgado Filho, que com seu exemplo e suas palavras, apoiou, contribuiu e inspirou a concretização deste sonho.

A toda minha família que sempre me apoiaram e incentivaram a prosseguir nessa caminhada, não me deixando abater pelas dificuldades.

Ao Programa de Pós-Graduação de Ciência da Computação - PPGCC da Universidade Federal do Maranhão - UFMA do curso de mestrado, pela dedicação e profissionalismo de seus professores e coordenador.

Muito Obrigado!

"Ele fortalece o cansado e dá grande vigor ao que está sem forças. Até os jovens se cansam e ficam exaustos, e os moços tropeçam e caem; mas aqueles que esperam no Senhor renovam as suas forças. Voam alto como águias; correm e não ficam exaustos, andam e não se cansam."

(Isaías 40:29-31)

Resumo

A complexidade de extrair conhecimento da imensa quantidade de dados gerados atualmente, cria a necessidade e traz a oportunidade de desenvolver mecanismos automatizados para acelerar o processo de descoberta de conhecimento, sem esquecer a precisão dos resultados e o incremento da produtividade dos analistas. O processo de descoberta de conhecimento é composto por várias fases sequenciais, bem definidas e relacionadas, desde a seleção de dados, pré-processamento, mineração e avaliação, até a descoberta de fato. Este trabalho apresenta uma proposta de automação da etapa de mineração de dados, com base em uma revisão teórica, levantamento de requisitos, estudo de arquiteturas e aplicações. Como contribuição, um ambiente de mineração de dados orientado ao usuário final foi desenvolvido e disponibilizado em nuvem. Este foi avaliado positivamente por usuários especialistas e também empregado em cenários de ensino-aprendizagem e autoestudo de Ciência de Dados.

Palavras-chave: Aprendizagem de Máquina, Análise Exploratória de Dados, Mineração de Dados, Descoberta de Conhecimento

Abstract

The complexity of extracting knowledge from the immense amount of data currently generated creates the need and brings the opportunity to develop automated mechanisms to accelerate the knowledge discovery process, without forgetting the accuracy of the results and the increase in the productivity of analysts. The knowledge discovery process consists of several sequential, well-defined and related phases, from data selection, pre-processing, mining and evaluation, until the discovery step. This work presents a proposal for automation of the data mining stage, based on a theoretical review, requirements gathering, architecture and applications study. As a contribution, an end-user-oriented data mining environment was developed and made available in the cloud. This was positively evaluated by expert users and also used in teaching-learning and self-study scenarios of Data Science.

Keywords: Machine Learning, Exploratory Data Analysis, Data Mining, Knowledge Discovery.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AED	Análise Exploratória de Dados
arff	Attribute-Relation File Format
CSS	Cascading Style Sheet (folhas de estilo em cascata)
DIKW	Data-Information-Knowledge-Wisdom (Dados-Informação-Conhecimento-Sabedoria)
GNU	General Public License (Licença Pública Geral)
HTML	HyperText Markup Language (Linguagem de Marcação de Hipertexto)
HTTP	Hypertext Transfer Protocol (Protocolo de Transferência de Hipertexto)
HTTP	Integrated Development Environment (Ambientes de desenvolvimentos integrados) - IDE
LAWS	Laboratory of Advanced Web Systems (Laboratório de Sistemas Avançados da Web)
KDD	Knowledge Discovery in Databases (Descoberta de Conhecimento de Bases de Dados)
kNN	k-Nearest-Neighbours (k-vizinhos-mais-próximos)
MVC	Model-view-controller (Modelo-Visão-Controlador)
PCA	Principal Component Analysis (Análise de Componentes Principais)
SVM	Support Vector Machine (Máquina de vetores de suporte)
TCLE	Termo de Conhecimento Livre e Esclarecido
UFMA	Universidade Federal do Maranhão
WEKA	Waikato Environment for Knowledge Analysis (Ambiente para Análise de Conhecimento Waikato).

Sumário

1 Introdução

Não há como negar que a informação nos dias de hoje possui uma importância cada vez maior na sociedade. Estar bem informado é uma vantagem competitiva, estratégica. Dados precisos, céleres e organizados asseguram condições primordiais para as tomadas de decisão, investimentos e oportunidades. A informação tornou-se essencial e indispensável para qualquer setor da atividade humana (BRAGA, 2000). A informação sem uma seleção a partir de critérios objetivos acaba por não ter a eficácia necessária e esperada. O excesso de informação por si só não é capaz de gerar o conhecimento necessário.

Segundo Cortella (2008) a informação é cumulativa. Inúmeros são os meios e mecanismos para se armazenar a informação, grandes bancos de dados estão disponíveis ao alcance de um *click* com grande diversidade. Mas o conhecimento é seletivo e é necessário que se busque de forma objetiva e organizada para então se apropriar dela. O conhecimento é a combinação de dados e informações acrescentada de opinião, habilidade e experiência, conforme Chaffey e Wood (2005).

Fayyad et al. (1996) propôs uma forma interativa e iterativa para a descoberta do conhecimento em um banco de dados composto de cinco passos: seleção, pré-processamento, transformação, mineração e avaliação. O processo trata o dado bruto, modelando, aplicando contexto e significado, até que seja exposto o conhecimento encoberto pela multidão de dados.

Fayyad, Piatetsky-Shapiro e Smyth (1996) expõem que o método tradicional de transformar dados e informações em conhecimento depende de análise e interpretação manual, o que se torna completamente impraticável em muitos domínios devido ao aumento exponencial de informação que é gerada atualmente, logo, a necessidade de ampliar os recursos de análise humana para lidar com o grande número de dados que se pode coletar é econômica e científica.

A ciência de dados se popularizou impulsionada pela necessidade de se conhecer, a partir dos dados analisados, seus padrões e comportamentos com o intuito de prever resultados, antecipar diagnósticos e auxiliar nas tomadas de decisão. É uma área multidisciplinar que abrange, por proximidade de interesse, a matemática, estatística, computação e engenharias, tendo como principal objetivo a investigação de dados, utilizando métodos científicos, técnicas e algoritmos computacionais, onde se busca determinar relacionamentos sistemáticos, padrões, regras de associação, conexões escondidas e prevendo e antecipando tendências futuras (OZDEMIR, 2016; HODEGHATTA; NAYAK, 2016).

Conforme Behrens (1997), a análise exploratória de dados (AED) é uma estratégia de análise de dados que busca fornecer ferramentas conceituais e computacionais para a

elicitação de padrões, a fim de promover o desenvolvimento e o refinamento de hipóteses. Uma ferramenta de AED facilita a tarefa de explorar dados, pois abstrai a programação de sistema e de codificação dos algoritmos de análise, assim como, disponibiliza gráficos que ajudam nas descobertas de relacionamentos e correlações entre os registros, de uma forma mais clara e dinâmica, deixando ao especialista a tarefa de interpretação dos gráficos e da análise crítica, próprias da sua área fim.

Por intermédio de Fayyad et al. (1996) o processo de busca do conhecimento foi sistematizado em 5 fases, que vão desde a coleta e seleção de dados, se estendendo pela definição de padrões e modelos, tendo sua finalização na mineração dos dados e na descoberta do conhecimento oculto entre os dados. Todavia, este processo de analisar dados não é uma tarefa fácil. Requer conhecimentos e competências específicas de programação de computadores, além de algoritmos e técnicas de análise e interpretação de dados. Como respostas as dificuldades apresentadas, foram criados mecanismos e ferramentas para auxiliar no processo de exploração e análise dos dados, adaptando-as as necessidades que surgiam.

Sendo assim, a motivação deste trabalho é de possibilitar aos especialistas de análise de dados e pesquisadores, não-programadores, a utilização de um ambiente que simplifica o processo de extração de conhecimento de dados primários, e a possibilidade de aplicação do LAWSMiner no ensino e no autoestudo de ciência de dados.

1.1 Objetivos

1.1.1 Objetivo Principal

Este trabalho tem por objetivo criar e validar um ambiente interativo para automatização do processo de mineração de dados orientado ao usuário final, capaz de auxiliar na exploração, mineração, análise e visualização de dados, garantindo eficiência, agilidade, correção e produtividade.

1.1.2 Objetivos Específicos

Como objetivos específicos, destacam-se:

a) Prototipação do ambiente para mineração e visualização de dados;

b) Implementação de algoritmos de agrupamento, regressão e classificação para predição de dados;

c) Validação do LAWSMiner como ambiente de prática do ensino de aprendizagem de máquina;

c) Validação dos conceitos de mineração de dados estatísticos empregados no ambiente por usuários especialistas.

1.2 Organização do Trabalho

A presente dissertação está dividida em 5 capítulos. No capítulo 1 foi descrita a motivação e os objetivos geral e específicos do trabalho.

No Capítulo 2 são apresentados o referencial teórico e os trabalhos relacionados, o Capítulo 3 descreve a especificação do ambiente, enquanto o Capítulo 4 detalha a validação do ambiente pelos grupos de usuários escolhidos. No Capítulo 5, o trabalho encerra com as considerações finais.

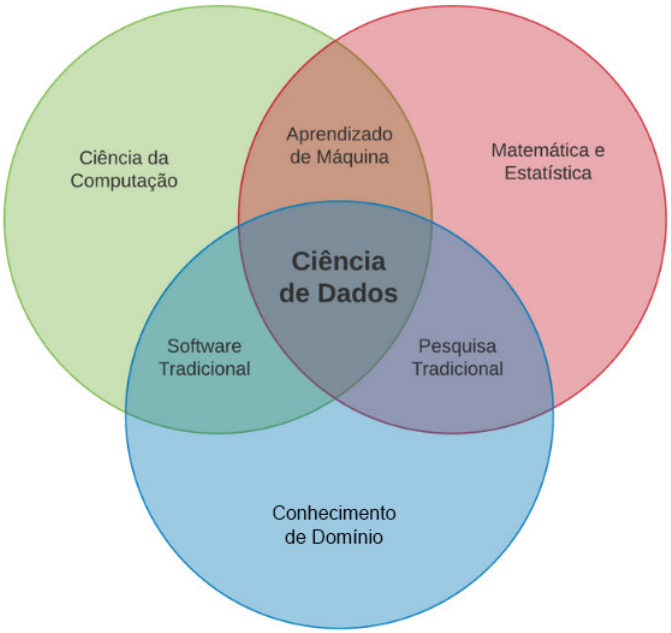
2 Fundamentação Teórica

Este capítulo apresenta alguns conceitos fundamentais para o entendimento da presente dissertação.

2.1 Ciência de Dados

Esta era, definitivamente, é a era de dados. Um grande volume de dados é gerado diariamente, o que torna, literalmente, impossível para um ser humano analisá-los em um tempo razoável (OZDEMIR, 2016). Os dados são coletados de várias formas e de diversas fontes diferentes, sendo algumas vezes sem um padrão preestabelecido de geração. Os dados, portanto, podem ter instâncias ausentes, incompletas ou simplesmente erradas, ou além disso, podem estar sobre escalas diferentes, o que dificulta mais ainda a comparação entre eles. Isso ocorre porque diferentes coletores de informações utilizam seus próprios esquemas ou protocolos para o registro de dados, resultando em diferentes e diversas representações desses dados, representando um enorme desafio a análise, a formatação e a extração de informações importantes a partir deles (Wu et al., 2014). Sendo assim, um dos principais objetivos da ciência de dados é propor práticas e procedimentos para melhor enfrentar esses desafios e extrair informações valiosas para uso na tomada de decisões estratégicas, desenvolvimento de produtos, análise de tendências e previsão.

Figura 1 – Diagrama de Venn da Ciência de dados.



Fonte: Adaptado de Ozdemir (2016).

A Ciência de Dados como disciplina multidisciplinar abrange o uso da matemática, estatística e ciência da computação, além da área especializada, conforme a Figura 1. O ponto importante a ser observado é a composição das três esferas de domínio, a saber: ciência da computação, estatística/matemática e especialização científica ou conhecimento de domínio.

Ciência da Computação: A Ciência da Computação disponibiliza mecanismos para a importação e o tratamento dos dados, as linguagens de programação para a codificação dos algoritmos de aprendizagem, e geração de modelos.

Matemática e Estatística: Basicamente todos os algoritmos aplicados a aprendizagem de máquina são baseados em conceitos matemáticos, e a estatística é característica intrínseca da ciência de dados.

Conhecimento de Domínio: O conhecimento de domínio é relativo à área que gera os dados, que sem seu entendimento do negócio a análise das informações coletadas fica prejudicada.

2.2 Processo de Análise de Dados

A análise exploratória de dados não é um estudo muito novo. A necessidade de se estudar o passado, entender o presente e antecipar o futuro sempre foi um grande desafio da humanidade.

Em 1977, John W. Tukey, com seu livro *Exploratory Data Analysis* (em português, Análise Exploratória de Dados), já demonstrava essa preocupação. Segundo ele, as técnicas de análise exploratória de dados adicionaram uma nova dimensão à maneira como as pessoas abordavam os dados e permitiam que a comunidade descobrisse características ocultas entre massas de números, e assim, retirar deles conhecimento.

Nesta seção serão vistos a hierarquia piramidal dos dados e o processo de descoberta do conhecimento.

2.2.1 Pirâmide do Conhecimento

Em ciência de dados os termos dados, informação, conhecimento e sabedoria são amplamente utilizados e fazem parte do conceito de DIKW (*Data-Information-Knowledge-Wisdom*), conhecida como pirâmide do conhecimento (BAŠKARADA; KORONIOS, 2013).

A pirâmide do conhecimento é estruturada em quatro partes, níveis ou camadas, que são os dados, informações, conhecimentos e por fim, a sabedoria, onde cada camada superior acrescenta atributos à camada inferior. A Figura 2 apresenta como estão dispostos os níveis da pirâmide.

Figura 2 – Pirâmide do Conhecimento - DIKW

Fonte: Elaborada pelo autor

Dados: Dados estão no nível mais básico e é o que dá sustentação à pirâmide. Os dados estão em grande número, mas por não estarem organizados e processados, sem contexto ou interpretação, não apresentam nenhum significado específico, geralmente correspondem a uma descrição elementar de fatos, eventos, coisas, atividades ou observações (AWAD; GHAZIRI, 2004).

Informação: As informações adicionam o contexto e o significado aos dados. As informações são dados processados, manipulados e organizados, submetidos a contextos específicos, relacionamentos ou associações, capazes de construir um significado, entendimento. Segundo Laudon e Laudon (2004), as informações são dados que foram moldados de forma significativa e útil.

Conhecimento: Para Chaffey e Wood (2005), conhecimento é a combinação de dados e informações, à qual se acrescenta opinião, habilidades e experiência de especialistas, para resultar em um ativo valioso que pode ser usado para auxiliar na tomada de decisões.

O conhecimento baseia-se em informações extraídas dos dados. Embora os dados sejam uma propriedade das coisas, o conhecimento é uma propriedade das pessoas que as predispõem a agir de uma maneira específica (BODDY D.; KENNEDY, 2005). O conhecimento adiciona a forma de como usar de modo apropriado a informação.

Sabedoria: A sabedoria acrescenta o entendimento de quando usar o conhecimento e a capacidade de julgar quais aspectos desse conhecimento são verdadeiros, corretos e aplicáveis.

Para Awad e Ghaziri (2004), a sabedoria é um conhecimento acumulado que permite entender como aplicar conceitos de um domínio a novas situações ou problema. É o nível mais alto de abstração, com previsão de visão e capacidade de enxergar além do horizonte

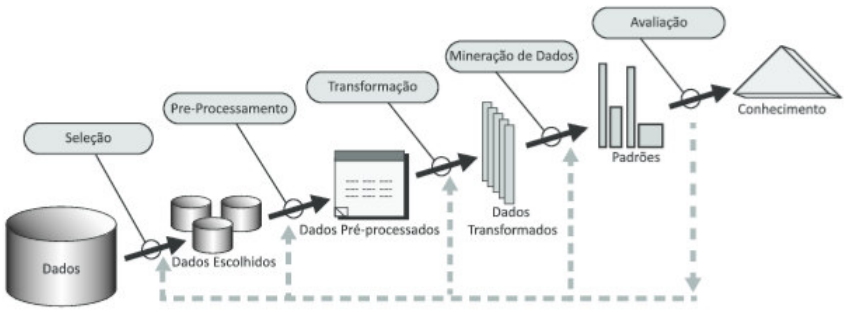
.

2.2.2 Processo de Descoberta de Conhecimento em Base de Dados - KDD (*Knowledge Discovery in Database)*

O processo de Descoberta do Conhecimento em Banco de Dados (KDD) proposto por Fayyad et al. (1996), Figura 3, é definido como "Um processo, de várias etapas, não trivial, interativo e iterativo, para identificação de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados."

A definição KDD traz algumas características importantes sobre o processo. O processo KDD é dito iterativo pois pode ser repetido, em partes ou no todo, quantas vezes for necessário, sendo que uma etapa depende da outra, para encontrar os padrões que sejam úteis e de fácil compreensão através da análise dos dados. Interativo porque o analista de dados pode intervir nas atividades. O processo é não-trivial pois apresentam uma certa complexidade na execução de processos sendo necessário conhecimentos específicos. E em etapas pois é constituído de passos bem definidos e interdependentes entres eles. As saídas de uma etapa são entradas para a etapa seguinte.

Figura 3 – Visão geral das etapas que constituem o processo KDD



Fonte: Fayyad et al. (1996)

As etapas do processo KDD são seleção e preparação de dados, pré-processamento, transformação, mineração de dados e interpretação dos resultados. Portanto, como alerta Fayyad, Piatetsky-Shapiro e Smyth (1996), apesar de comumente se chamar todo o processo de mineração, a mineração de dados é apenas uma etapa dentro do processo de descoberta, essa etapa tem como função a extração de padrões de dados, desempenhada pelos algoritmos de aprendizagem específicos

O processo de extração de conhecimento de uma base de dados consiste em:

Etapa de Seleção: é a primeira etapa do KDD, é uma etapa onde são selecionados os conjuntos de dados, e onde serão agrupados e organizados. A boa execução desta etapa é primordial para se obter resultados relevantes.

Etapa de Pré-processamento: Nesta etapa acontece a separação dos atributos e a limpeza dos dados. As informações ausentes, errôneas ou inconsistentes no conjunto de dados são preenchidas utilizando regras específicas, corrigidas ou retiradas, neste caso, todo o registro é removido para que a qualidade dos dados não seja comprometida.

Etapa de Transformação: A etapa de transformação ou formatação dos dados analisa os dados obtidos da etapa anterior e os reorganiza e armazena de forma a facilitar a interpretação na etapa seguinte.

Etapa de Mineração dos Dados: é a principal etapa no processo de descoberta do conhecimento. Os dados depois de transformados são lidos e interpretados. A mineração faz com que os dados selecionados, tratados e reorganizados sejam transformados em informações.

Etapa de Interpretação e Avaliação: Nesta etapa as regras criadas (modelos de dados) na mineração de dados são interpretadas e avaliadas. Após a interpretação poderão surgir padrões, relacionamentos e descoberta de novos fatos, que serão validados e utilizados.

2.3 Aprendizado de máquina

A aprendizagem é o processo pelo qual as competências, as habilidades, os conhecimentos, os comportamentos ou os valores são adquiridos ou modificados, como resultado de estudo, experiência, formação, raciocínio e observação (NOGARO A.; ECCO, 2014). Este processo pode ser analisado a partir de diferentes perspectivas, de forma que há diferentes teorias de aprendizagem que incluem além da aquisição de novos conhecimentos declarativos, o desenvolvimento de habilidades motoras e cognitivas por meio de instrução ou prática, a organização de novos conhecimentos em representações gerais e eficazes e a descoberta de novos fatos e teorias por meio de observação e experimentação (VELASQUEZ; SAUCEDA, 2001).

A definição de aprendizagem de máquina segundo Mitchell (1997) diz que um computador é capaz de aprender a partir da experiência adquirida por meio da execução de uma classe de tarefas:

> "Diz-se que um programa de computador aprende, a partir da experiência E em relação a uma classe de tarefas T e uma medida de desempenho P, se seu desempenho nas tarefas de T, medido por P, melhora com a experiência E"

FACELI et al. (2015) complementa a definição de Mitchell (1997) especificando que a experiência obtida advém de conclusões encontradas a partir de análises de um conjunto

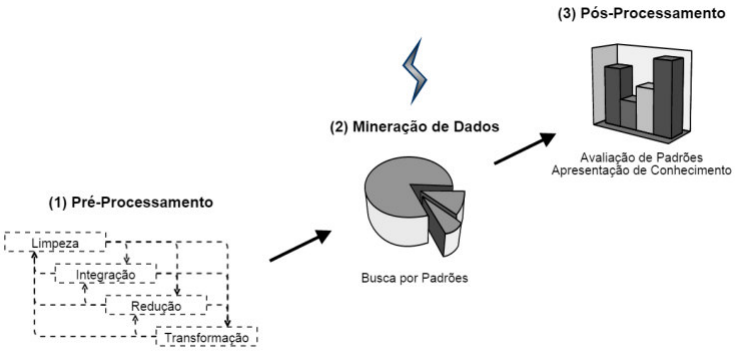
particular de dados, aos quais dão subsídios ao computador para induzir respostas para um conjunto de dados novos:

> Em Aprendizado de Máquina, computadores são programados para aprender com a experiência passada. Para tal, empregam um princípio denominado indução, no qual se obtêm conclusões genéricas a partir de um conjunto particular de exemplos. Assim, algoritmos de Aprendizado de Máquina aprendem a induzir uma função ou hipótese capaz de resolver um problema a partir de dados que representam instâncias do problema a ser resolvido.

O aprendizado de máquina refere-se ao processo pelo qual os computadores desenvolvem o reconhecimento de padrões ou a capacidade de aprender continuamente com dados conhecidos, extraindo informações destes e a partir daí fazer previsões e/ou ajustes sem serem especificamente programados para isso, aplicando o comportamento aprendido para a resolução de novos problemas, por isso, muitas das técnicas utilizadas nessa área são utilizadas em estatística, *business intelligence*, mineração de dados e ciência de dados.

O aprendizado de máquina é um ramo da inteligência artificial, o qual automatiza efetivamente o processo de construção de modelos analíticos e permite que as máquinas se adaptem a novos cenários de forma independente. Ele possui 3 etapas bem definidas: Pré-processamento, Mineração de Dados e Pós-processamento (Figura 4)

Figura 4 – Etapas da Aprendizagem de Máquina



Fonte: Maciel et al. (2015)

A etapa de Pré-processamento é responsável pela captação, organização e preparação dos dados. É de extrema importância e compreende desde a análise dos dados até sua formatação e normalização. Nessa etapa é realizada a coleta e a seleção dos dados, a análise dos dados coletados, tratamento de valores ausentes e não conformes, e a transformação com incorporação ou mesmo a criação de novos dados a partir dos dados existentes conforme preconiza Fayyad et al. (1996).

A etapa de mineração de dados consiste na aplicação de um algoritmo que busca, efetivamente, por padrões/relações e regularidades em um determinado conjunto de dados, buscando identificar informações ainda desconhecidas pelo usuário.

Na etapa de Pós-processamento é verificada a qualidade do conhecimento descoberto e os padrões encontrados com intuito de trazer novas perspectivas de resolução de problemas que motivaram a realização da análise.

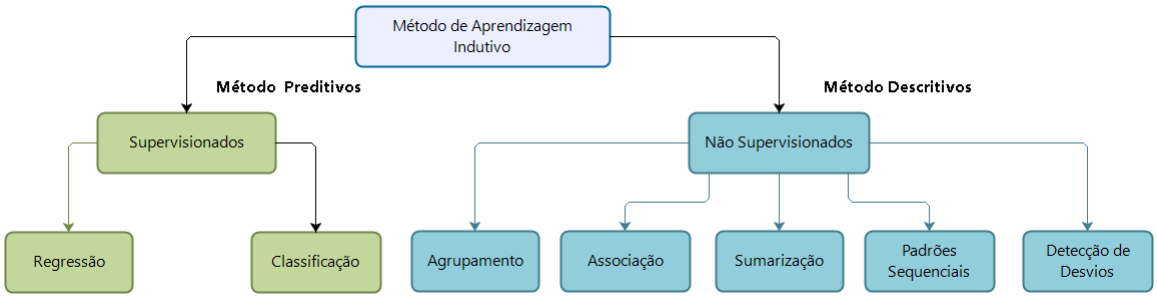
2.3.1 Métodos de Aprendizagem Indutivos

Segundo Monard e Baranauskas (2003), "indução é a forma de inferência lógica que permite obter conclusões genéricas sobre um conjunto particular de exemplos. Ela é caracterizada como o raciocínio que se origina em um conceito específico e o generaliza, ou seja, da parte para o todo". Logo, um novo conceito (hipótese) é aprendido por meio da inferência indutiva de algo já conhecido, sendo que neste processo, a verdade contida na premissa pode ou não ser preservada, o que é um problema, mas, mesmo assim, a inferência indutiva é um dos principais métodos utilizados para derivar conhecimento novo e predizer eventos futuros.

Os métodos de aprendizagem são, resumidamente, funções matemáticas que aplicadas a um conjunto de dados, são capazes de identificar padrões e prever ou antecipar o que pode acontecer com novas instâncias de dados submetidas ao mesmo modelo, ou mesmo descrever o comportamento destes conjuntos de dados entre si.

Portanto, conforme a Figura 5 o aprendizado indutivo, basicamente, se divide em Métodos Preditivos (supervisionado) e Métodos Descritivos (não supervisionado). Os modelos supervisionados são divididos em modelos de regressão e de classificação, já os não supervisionados em agrupamento, associação, sumarização, padrões sequenciais e detecção de desvios.

Figura 5 – Métodos de Aprendizagem Indutivo



Fonte: Elaborada pelo autor.

Aprendizado supervisionado: No método de aprendizado supervisionado, o sistema é programado ou treinado a partir de um conjunto de dados pré-definidos ou rotulados. O programa treinado é capaz de tomar suas próprias decisões quando submetido a um novo conjunto de dados (FOLLOW, 2016). A grande maioria das aplicações de aprendizado de máquina utilizam o método de aprendizado supervisionado. Ele se aplica

em situações em que a análise de dados históricos permite antecipar comportamentos futuros.

Aprendizado não supervisionado: No aprendizado não supervisionado, diferente do aprendizado supervisionado, o programa não passa por uma etapa de treino, pois nesta categoria, o algoritmo, a partir de um conjunto de dados de entrada, é capaz de encontrar padrões, descobrir semelhanças entre esses dados e agrupá-los adequadamente, numa modelagem descritiva. Apresenta dados de entrada semelhantes ao processo de aprendizado supervisionado, no entanto, não há categorias ou rótulos de saída nos quais o algoritmo possa tentar modelar relacionamentos. Esses algoritmos tentam usar técnicas nos dados de entrada para pesquisar regras, detectar padrões, resumir e agrupar os pontos de dados que ajudam a obter informações significativas e descrever melhor os dados para os usuários (FOLLOW, 2016).

Aprendizado por reforço: Neste modelo, o algoritmo deve identificar qual é o caminho a seguir a fim de descobrir qual tem a melhor recompensa por meio de experimentação (tentativas e erros). Conforme Sutton e Barto (1998), nos casos mais desafiadores, as ações podem afetar não apenas a recompensa imediata, mas também a próxima situação e, com isso, todas as subsequentes. Essas duas características, pesquisa por tentativa e erro e recompensa atrasada, são as duas características distintivas mais importantes do aprendizado por reforço.

Seu uso é mais comum em aplicações de robótica, navegação e games.

Regressão: São empregados quando se deseja modelar e analisar a relação entre uma variável dependente (resposta) e uma ou mais variáveis independentes (preditoras).

Classificação: São utilizados para identificar a categoria de novas observações com base em um modelo de classificação construído a partir do conjunto de dados.

Agrupamentos: São algoritmos de aprendizado não supervisionado, são utilizados para agrupar objetos semelhantes (próximos em termos de distância) em um mesmo grupo.

Associação: Uma associação pode ser analisada da seguinte maneira: dado um conjunto de registros e uma coleção de itens, cada um deles identificados com alguns números de itens e de uma coleção, a função de associação é retornar afinidades que existem na coleção de itens deste conjunto de registros. As afinidades podem ser expressas através de regras, como por exemplo, 80% dos registros que contém os itens A e B, também contém os C e D. Em um banco de dados podem ser encontradas várias regras de associação.

Sumarização: Técnicas que permitem a identificação de uma descrição compacta e inteligível para os dados (ou para um subconjunto deles). Frequentemente é possível sumarizar os dados mesmo com alguma imprecisão, e o valor das técnicas é na capacidade de descrever os dados, não necessariamente em sua precisão.

É possível sumarizar os dados de uma base ou coleção através de técnicas de classificação, mas nem toda técnica de classificação cria modelos que descrevem os dados que podem ser facilmente interpretados (SANTOS, 2009).

Padrões Sequenciais: Os padrões sequenciais são obtidos através de análise, contidos em um determinado conjunto de dados. Podem ser aplicados em um conjunto de dados onde constam informações de compras dos consumidores para verificar os conjuntos de produtos comprados pelos mesmos, bem como analisar o perfil dos consumidores.

Detecção de mudança ou desvios (outliers): Técnicas que permitem a descoberta e identificação de dados que não se comportam de acordo com um modelo aceitável dos dados (ou, por exemplo, mudanças em séries temporais ou em dados indexados por tempo). Estas técnicas podem identificar mudanças ou padrões inesperados em todos os dados ou em um subconjunto (SANTOS, 2009).

2.3.2 Classificação

Segundo Kelleher, Namee e D'Arcy (2015) e Hodeghatta e Nayak (2016) a classificação é uma subcategoria de aprendizagem supervisionada cujo objetivo é determinar a qual classe determinado registro pertence. O processo é realizado em duas etapas, treinamento e teste. O conjunto de dados é divido em duas partes, normalmente, 70% dos dados para treinamento e 30% dos dados para teste.

Treinamento: Na primeira etapa, um modelo é construído a partir da análise de dados proveniente de uma amostra de dados de treinamento e o conjunto de atributos que definem a variável de classe. Os dados do treinamento são uma amostra do banco de dados e o atributo de classe já é conhecido;

Teste: Na segunda, o modelo gerado na primeira etapa é aplicado em uma segunda amostra da amostra de dados, dados de teste, onde a variável de classe é predita e comparada com o valor existente no banco de dados, apurando assim, a acurácia do modelo.

Na classificação, determina-se a relação entre a variável classe e as entradas ou variáveis explicativas. Normalmente, os modelos representam as regras de classificação ou fórmulas matemáticas. Depois que essas regras são criadas pelo modelo de aprendizado, esse modelo pode ser usado para prever a classe de outros conjuntos de dados onde a determinada classe é desconhecida (HODEGHATTA; NAYAK, 2016).

2.3.3 Regressão

Assim como a Classificação, a Regressão também é uma subcategoria de aprendizagem supervisionada. Logo, por característica deste tipo de aprendizagem, os algoritmos de regressão também possuem duas etapas, treinamento e teste. No treinamento é gerado

o modelo de regressão baseado na variável alvo e as variáveis explicativas. E no teste, é avaliada a predição da classe predita (variável alvo). A partir de então é calculada, entre outros, a matriz de confusão e acurácia para validação do modelo.

Na Regressão, como na subcategoria anterior, determina a relação entre a variável alvo e as variáveis explicativas, a diferença é que ao invés de variáveis categóricas, a variável alvo é contínua, isso significa que a resposta pode assumir uma gama de valores infinitos (OZDEMIR, 2016).

2.3.4 Agrupamento

Linden (2009) diz que a análise de agrupamento tem como objetivo separar objetos em grupos, baseando-se nas características que estes objetos possuem, colocando em um mesmo grupo objetos que sejam similares de acordo com algum critério pré-determinado.

As técnicas de agrupamento são consideradas como não supervisionadas. Dado um conjunto de registros, são gerados agrupamentos (ou cluster), contendo os registros mais semelhantes. O armazenamento em cluster pode descobrir relacionamentos anteriormente não detectados em um conjunto de dados e assim agrupá-los segundo suas características (HODEGHATTA; NAYAK, 2016).

Normalmente os pontos de similaridade são medidas por meio de distâncias tradicionais (Euclidiana, Manhattan, entre outras). Os elementos de um cluster são considerados similares aos elementos no mesmo cluster e dissimilares aos elementos nos outros clusters. Por trabalhar com o conceito de distância (similaridade) entre os registros, geralmente é necessário realizar a transformação dos diferentes tipos de dados (ordinais, categóricos, binários, intervalos) para uma escala comum, exemplo [0,0, 1,0].

A seguir tem-se alguns dos algoritmos de aprendizado mais utilizados na predição de eventos, sendo eles de aprendizado supervisionado, não supervisionado ou de reforço e ainda de regressão, agrupamento e classificação.

2.4 Algoritmos de Aprendizado de Máquina

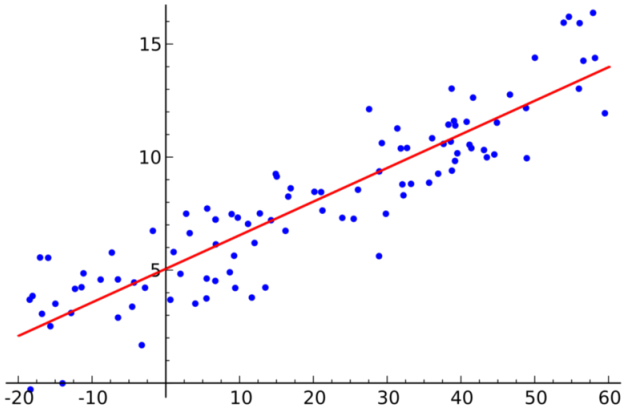
2.4.1 Regressão Linear

A regressão linear é um dos algoritmos mais conhecidos e bem compreendidos em estatística e aprendizado de máquina. Ela é um método de predição numérica que consiste em encontrar uma relação linear, entre preditores e uma variável de resposta, estabelecendo uma relação de causa efeito entre elas Ozdemir (2016).

As técnicas de regressão modelam o relacionamento de variáveis independentes (preditoras) com uma variável dependente (resposta). As variáveis preditoras são os

atributos dos registros, e a resposta é o que se deseja predizer.

Figura 6 – Gráfico da Regressão Linear



Fonte: Tirzite et al. (2018)

A linha vermelha no gráfico da Figura 6 é referida como a linha reta de melhor ajuste. Com base nos dados fornecidos, traça-se uma linha que modela os pontos melhor.

2.4.2 Regressão Logística

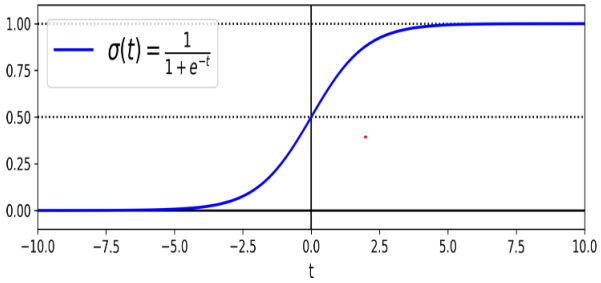
A regressão logística (Figura 7) é outra técnica de predição numérica emprestada pelo aprendizado de máquina do campo da estatística. Segundo Corrar, Paulo e Filho (2007), ela é um recurso que nos permite estimar a probabilidade associada à ocorrência de determinado evento em face de um conjunto de variáveis explanatórias . Seu uso busca estimar a probabilidade de certa variável dependente assumir um determinado valor em função de valores conhecidos de outras variáveis, sendo assim, por se tratar de um evento probabilístico seu resultado compreende o intervalo de zero a um.

De muitas formas, a regressão linear e a regressão logística são semelhantes. Contudo a maior diferença está no propósito que são empregadas. Enquanto algoritmos de regressão linear são usados para prever valores, a regressão logística é usada para tarefas de classificação, como por exemplo, classificar se um e-mail é spam ou não, se um tumor é maligno ou benigno, ou ainda se um site é fraudulento ou não.

2.4.3 K Vizinhos Próximos

O classificador *k-Nearest-Neighbours* (kNN), em português, K vizinhos mais próximos, é um método de classificação simples, geralmente baseado na distância euclidiana entre uma amostra de teste e as amostras de treinamento especificadas. O objetivo desse método é atribuir uma associação em função da distância do vetor dos vizinhos mais próximos de uma amostra e da associação dos grupos de classes possíveis (TSIHRINTZIS et al., 2019).

Figura 7 – Regressão Logística

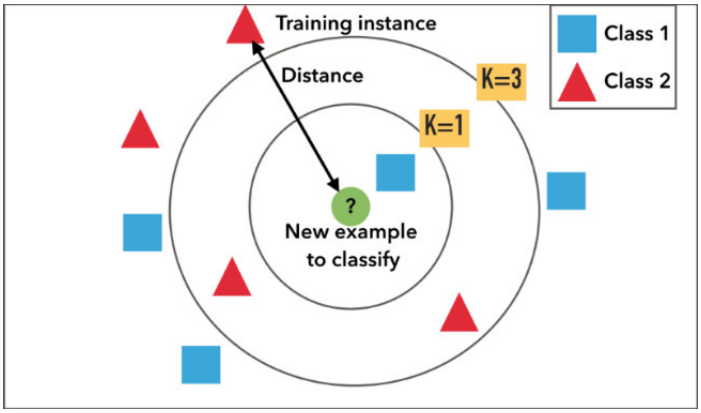


Fonte: Géron (2019)

O algoritmo kNN inicia com um conjunto de dados de treinamento composto por exemplos que são classificados em várias categorias, rotulados por uma variável nominal. O conjunto de dados de teste contendo exemplos não rotulados que, de outra forma, possuem os mesmos recursos que os dados de treinamento. Para cada registro no conjunto de dados de teste, kNN identifica k registros nos dados de treinamento que são os mais próximos em similaridade, onde k é um número inteiro especificado previamente. A instância de teste sem rótulo recebe a classe da maioria dos k vizinhos mais próximo (LANTZ, 2013), conforme a Figura 8.

Devido suas características este algoritmo é fortemente usado quando precisa-se classificar dados que se tem pouco conhecimento prévio a respeito e aplicado amplamente em sistemas de recomendação, pesquisa semântica e detecção de anomalias.

Figura 8 – *k-Nearest-Neighbours*



Fonte: Tolpygo (2016)

2.4.4 Agrupamento K-Means

Agrupamento K-Means é um algoritmo de aprendizado não supervisionado que tem como objetivo agrupar ou particionar pontos de dados em clusters com centroides (OZDEMIR, 2016).

Os algoritmos não supervisionados processam dados de entrada sem rótulos ou treinamentos anteriores, a fim de criar padrões e relacionamentos e agrupar objetos em grupos, de modo que os elementos de um grupo sejam mais semelhantes entre si do que os de outros grupos, conforme Follow (2016).

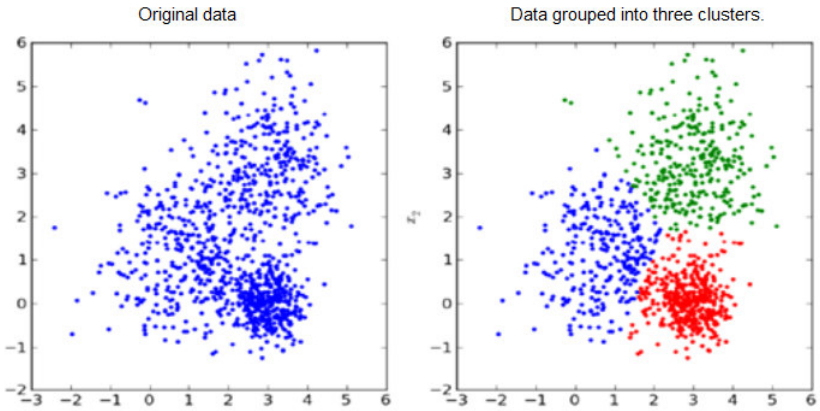
O algoritmo funciona iterativamente para atribuir cada ponto de dados a um dos grupos K com base nos recursos fornecidos. Os pontos de dados são agrupados com base na similaridade do recurso.

Como resultados do algoritmo de agrupamento K-means é encontrado os centroides dos clusters K (Figura 9), que podem ser usados para rotular novos dados e etiquetas para os dados de treinamento (cada ponto de dados é atribuído a um único cluster)

Em vez de definir grupos antes de analisar os dados, o agrupamento permite localizar e analisar os grupos que foram formados organicamente.

Cada centroide de um cluster é uma coleção de valores de recursos que definem os grupos resultantes. Examinar os pesos dos recursos do centroide pode ser usado para interpretar qualitativamente o tipo de grupo que cada cluster representa.

Figura 9 – K-Means Clustering



Fonte: Adaptado de Gattal, Faycel e Laouar (2018)

2.4.5 Árvores de Decisão

Os algoritmos de classificação de árvores de decisão são algoritmos de treinamento supervisionado muito utilizado na inferência indutiva, classificação, regressão e previsão de dados.

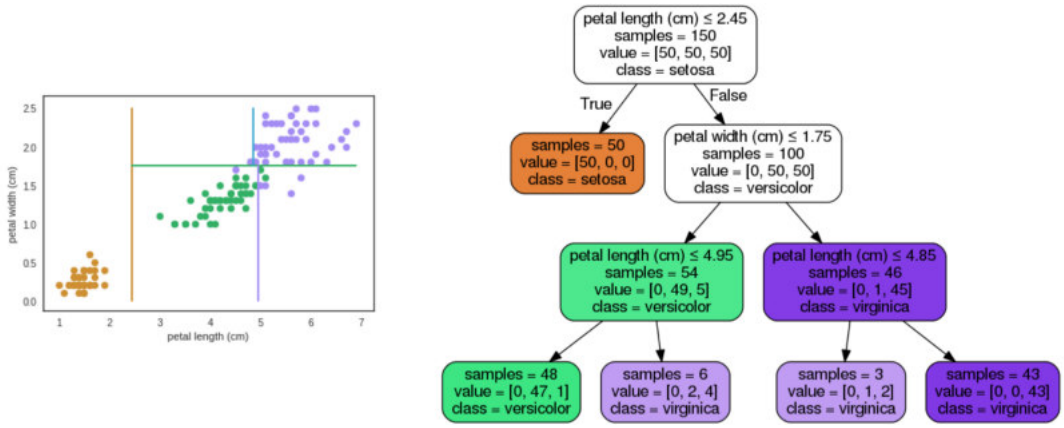
O aprendizado do algoritmo se dá em duas partes. A primeira é a de treinamento que é caracterizado pela construção do modelo, que por sua vez, analisa e descreve um conjunto de dados predeterminado e o teste se refere à análise do desempenho do modelo.

A árvore de decisão é um algoritmo de fácil entendimento, implementação, e um dos mais populares (OLEKSY, 2018), muito usado para tomadas de decisão, mas devido sua

concepção, o algoritmo é fortemente dependente de seus dados de treinamento, possuindo alta variância, prejudicando bastante o seu desempenho, ocasionando o fenômeno de (*over fitting*), termo usado para descrever quando um modelo estatístico se ajusta muito bem ao conjunto de dados anteriormente observado, mas se mostra ineficaz para prever novos resultados.

A figura 10 mostra como se apresenta o gráfico do algoritmo de Árvore de Decisão. Visualmente consegue-se perceber como os dados da amostra estão classificados e quais os critérios foram utilizados para montar cada ramo da árvore.

Figura 10 – Árvore de Decisão - Classificação



Fonte: Géron (2019)

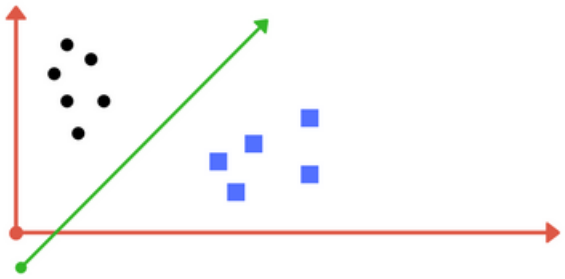
2.4.6 Máquinas de Vetores de Suporte

As máquinas de vetores de suporte pertencem à família de algoritmos de aprendizado de máquina supervisionados usados para analisar dados e reconhecer padrões, usado principalmente para a classificação binária, mas também utilizado na análise de regressão.

Na classificação binária, o algoritmo SVM (do inglês: *support vector machine*) constrói um modelo a partir dos dados de treinamento de forma que os pontos mapeados como de classes diferentes são separados por uma lacuna chamada de hiperplano 11). As amostras nas margens são normalmente chamadas de vetores de suporte. Logo o intuito do algoritmo é encontrar um hiperplano em um espaço n-dimensional que possa classificar os pontos de dados binariamente e que as margens distam o máximo possível Follow (2016).

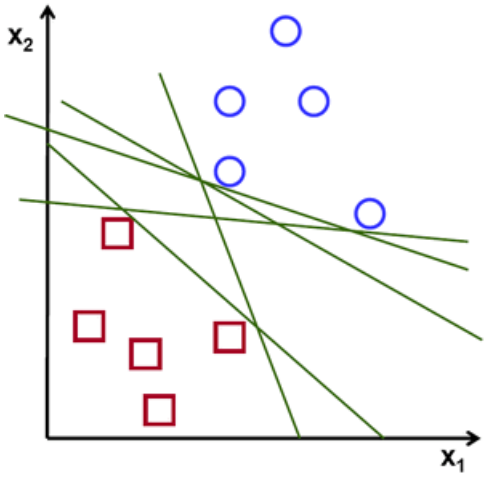
Para separar as duas classes de pontos de dados (Figura 12), existem muitos hiperplanos possíveis que podem ser escolhidos, a melhor classificação é aquela que possuir maiores distâncias entre as margens (Figura 13). Maximizar a distância da margem fornece algum reforço para que os pontos de dados futuros possam ser classificados com maior confiança.

Figura 11 – Máquina de Vetor de Suporte - SVM



Fonte: Patel (2017)

Figura 12 – Hiperplanos possíveis



Fonte: Patel (2017)

2.5 Trabalhos Relacionados

Esta seção aborda brevemente os trabalhos relacionados a ferramentas de visualização e mineração de dados, com o objetivo de contextualizar as soluções e posicionar o ambiente proposto nesse trabalho.

2.5.1 Weka

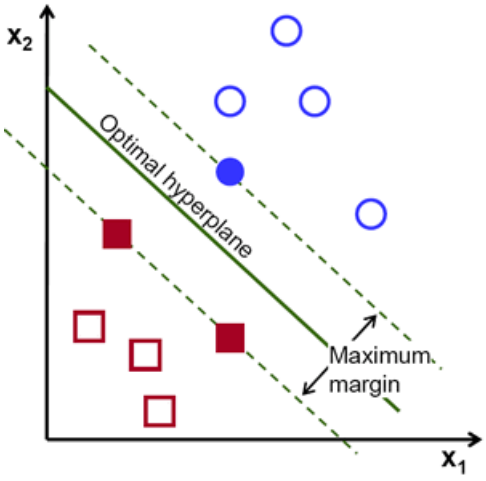
Weka[1] é um projeto *open source* que significa *Waikato Environment for Knowledge Analysis* – Ambiente para Análise de Conhecimento Waikato. Foi criado como um projeto de *Machine Learning* pela Universidade de Waikato na Nova Zelândia.

O projeto tem como objetivo disseminar técnicas de aprendizado de máquina, disponibilizando um software para utilização de pesquisadores, alunos com intuito de resolver problemas reais da indústria.

O software, feito em Java, possibilita consulta a base de dados, faz a análise e executa algoritmos de aprendizagem.

[1] https://www.cs.waikato.ac.nz/ml/index.html

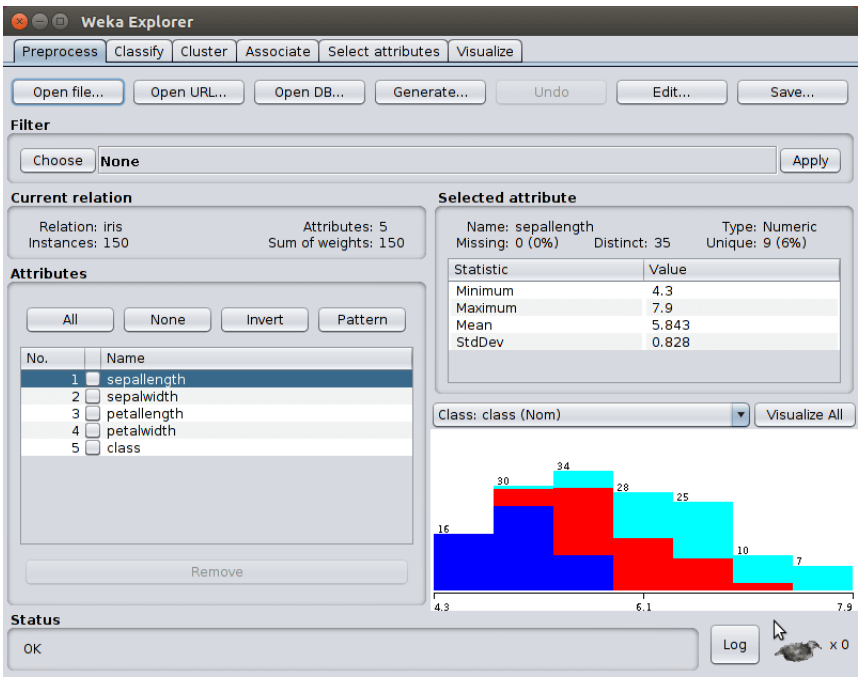
Figura 13 – Margem máxima entre planos



Fonte: Patel (2017)

O principal problema encontrado é o fato de o sistema ter apenas uma versão *desktop* e necessita ser instalado no computador para ser utilizado, apesar de ser livremente utilizado nas plataformas Windows, Linux e Mac.

Figura 14 – Weka



Fonte: Brownlee (2016)

2.5.2 Tableau

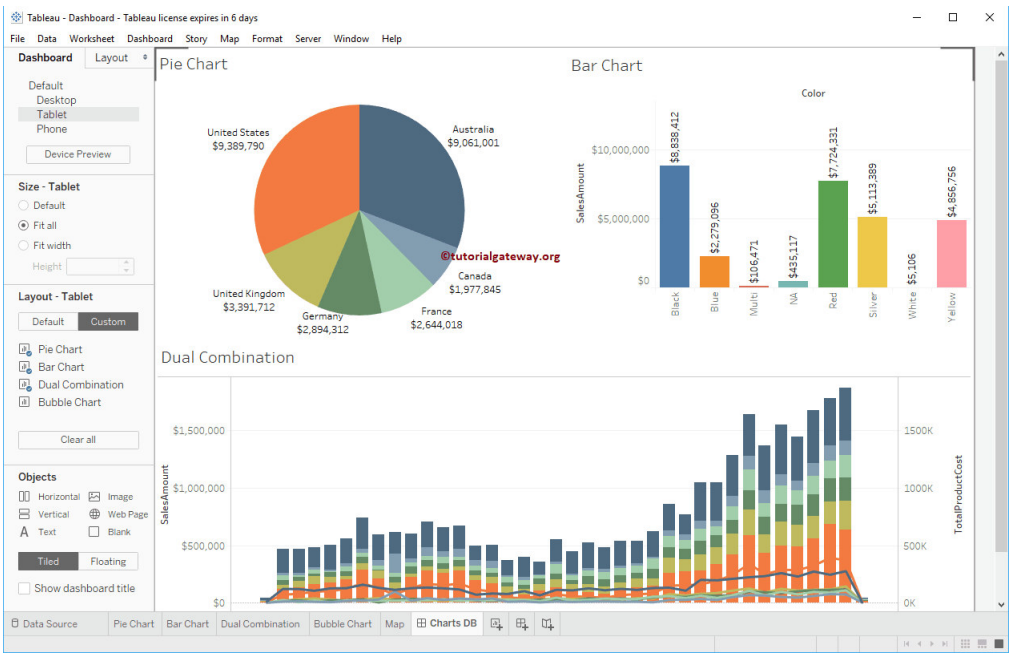
O Tableau[2] é uma outra ferramenta utilizada para visualização de dados.

[2] https://www.tableau.com/pt-br

Esta ferramenta possui duas versões, via Web e Desktop, e dois tipos de licenças, gratuita e paga. A versão gratuita para teste possui funcionalidades reduzidas, bloqueando algumas funções. Já a versão paga, todas as funções são liberadas com a permissão de uso de multiusuários.

A interface da ferramenta é *Drag-and-drop* (arrastar/soltar) o que facilita a utilização, apesar de não dispor de algoritmos de aprendizagem de máquina.

Figura 15 – Tableau



Fonte: (SOUZA, 2018)

2.5.3 Python

O Python[3] é uma linguagem de programação de alto nível, interpretada, orientada a objetos e fortemente tipada. É uma linguagem versátil, usada não só no desenvolvimento Web mas em muitos outros tipos de aplicação, sendo bastante utilizada para mineração e visualização de dados. Para tal dispõe de inúmeras bibliotecas e pacotes que podem ser agregadas para potencializar este processo.

O pacote **Pandas** é a ferramenta mais importante à disposição dos cientistas e analistas de dados que trabalham atualmente em Python (MCINTIRE; MARTIN; WASHINGTON, 2019). Ele disponibiliza mecanismos para carga, limpeza, transformação e análise dos dados.

A biblioteca Pandas é um componente do kit de ferramentas de ciência de dados do Python, sendo utilizada em conjunto com outras bibliotecas. Dentre estas utilizados na mineração e visualização de dados as que mais se destacam são:

[3] https://www.python.org/

Matplotlib: a ferramenta precisa de muitas linhas de código para gerar um gráfico, mas é bastante relevante porque é a base para diversas outras bibliotecas;

Seaborn: baseada em Matplotlib, porém mais fácil de usar, a seaborn possui vários tipos de gráficos e um visual bonito;

ggplot: biblioteca originada no R e adaptada para Python, é forte em estatística e permite plotar gráficos com dados estatísticos usando poucas linhas de código;

Bokeh: a bokeh é uma biblioteca com elementos interativos, isto é, o gráfico se movimenta e possui zoom.

Portanto, apesar da robustez do Python em garantir que grande volume de dados sejam tratados na ferramenta, ele requer bons conhecimentos de linguagem de programação e de algoritmos de aprendizagem de máquina para operacionalizar a análise de dados, ou seja, é necessário que seus usuários tenham a competência e habilidade em programação de computadores, conhecer seus comandos e principalmente conhecer o funcionamento de cada algoritmo que queira implementar.

2.5.4 RapidMiner

RapidMiner[4] é uma plataforma para trabalhar com ciência de dados de forma rápida, simples e visual. A ferramenta fornece uma interface gráfica com objetos e processos que simplificam as tarefas necessárias para trabalhar mineração de dados, além da possibilidade de utilização de modelos de predição.

Com o RapidMiner Studio é possível criar *workflows* intuitivos com objetos que executam todas as tarefas do processo de mineração de dados, como, leitura e carregamento dos dados, limpeza e transformação, filtragem, modelagem, aplicação de algoritmos de aprendizado de máquina e visualização dos resultados.

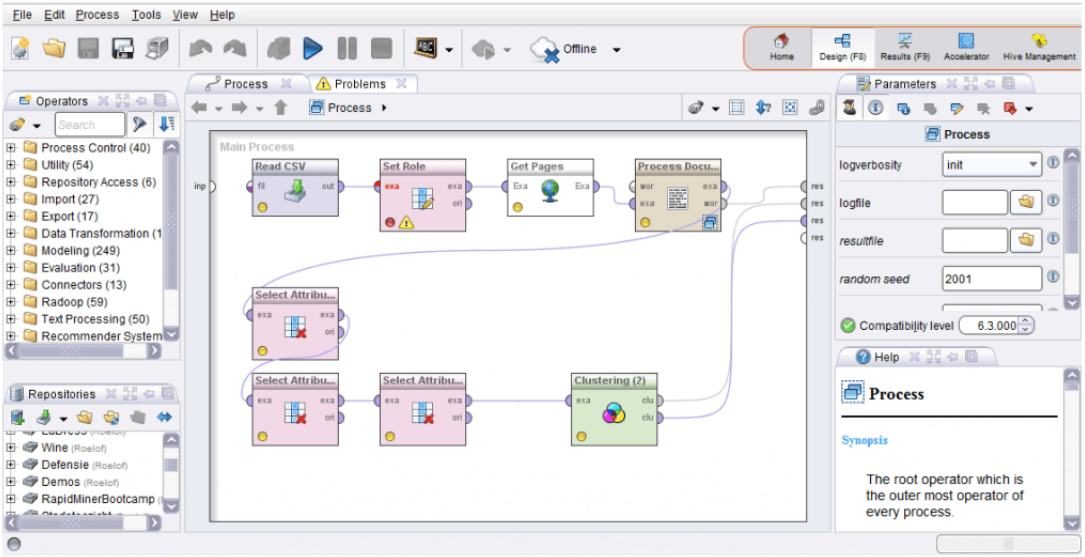
A plataforma oferece um tipo de licenciamento gratuito que permite a utilização do RapidMiner Studio, mas neste pacote limita o tamanho da base de dados.

2.5.5 Considerações

O ambiente proposto, LAWSMiner, se diferencia das opções listadas nesta seção pois foi desenvolvida em linguagem de programação bastante difundida no meio acadêmico, R, e de fácil manutenção. Seu uso é gratuito e todas as funcionalidades são liberadas para todos os usuários. Ambiente web, que não necessita de instalações adicionais além de um navegador de internet. Suas funcionalidades são claras e objetivas e o usuário tem o completo domínio sobre as funções que estão sendo executadas. E por fim, está

[4] https://rapidminer.com/

Figura 16 – RapidMiner Studio



Fonte: Sanyal (2018)

desenvolvido seguindo o processo de descoberta de conhecimento proposto por Fayyad et al. (1996) o que estabelece passos interativos e iterativos para o ambiente.

2.6 Tecnologias Utilizadas

2.6.1 A linguagem de Programação R

R[5] é uma linguagem de script para manipulação e análise de dados estatísticos e um ambiente de desenvolvimento voltado para a computação científica e estatística, programável e altamente extensível (R Core Team, 2019).

R é um software livre que pode ser instalado gratuitamente através da internet, redistribuído e/ou modificado sob os termos do GNU (*General Public License*). O ambiente R é constituído por um conjunto de ferramentas que permite o armazenamento, processamento, cálculo, análise e visualização de dados sendo este último o ponto forte do R, é possível a plotagem de gráficos de qualidade com relativa facilidade.

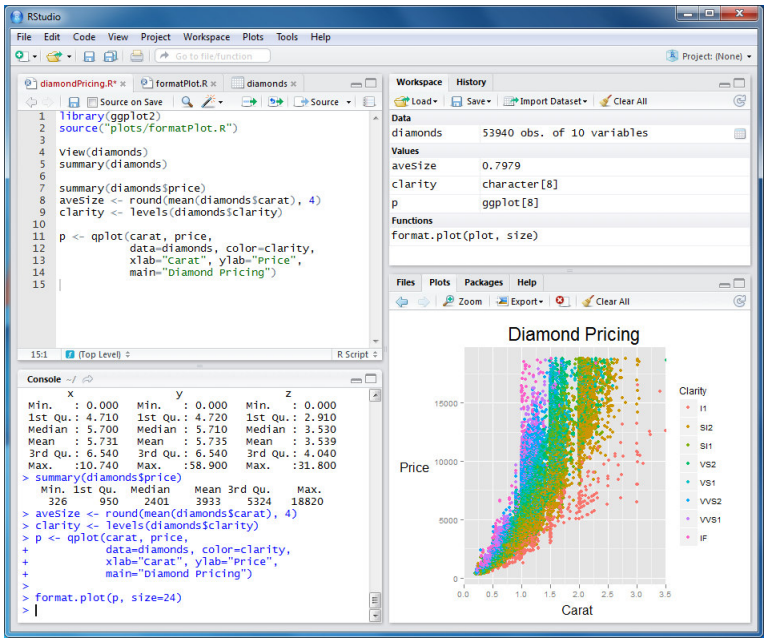
Outro ponto forte da linguagem/ambiente é a possibilidade que usuários adicionem novas funcionalidade através de pacotes, logo R é um conjunto integrado de recursos de software para a manipulação de dados, cálculo e exibição gráfica, além de um conjunto de operadores para cálculos em matrizes, instalações gráficas para análise de dados e exibição na tela, e uma linguagem de programação bem desenvolvida, simples e eficaz que inclui condicionais, loops, funções e recursos de entrada e saída. Portanto R é bem mais que uma

[5] https://cran.r-project.org/

linguagem ou um sistema de estatísticas é um ambiente no qual as técnicas estatísticas são implementadas.

Os Ambientes de desenvolvimentos integrados (*Integrated Development Environment* - IDE) foram criados com intuito de facilitar o desenvolvimento agregando funcionalidades, softwares, arquivos e configurações que ajudam na preparação do ambiente de desenvolvimento. Dentre esses ambientes que dão suporte a codificação em R destaca-se o RStudio[6]. Esta ferramenta reúne diversas funcionalidades, tornando mais amigáveis aos usuários a importação de dados, visualização de comandos, funções, resultados e gráficos, além da geração de documentos (Figura 17).

Figura 17 – RStudio



Fonte: Faria (2016)

2.6.2 Biblioteca Shiny

O Shiny[7] é um pacote R. Este pacote tem a função de auxiliar o desenvolvimento de aplicações *Web* interativas e amigáveis utilizando apenas a linguagem R, sem depender de javascripts, css, ou outros, mas sem limitar ou excluir o uso de outras linguagens ou *scripts* nele (Figura 18).

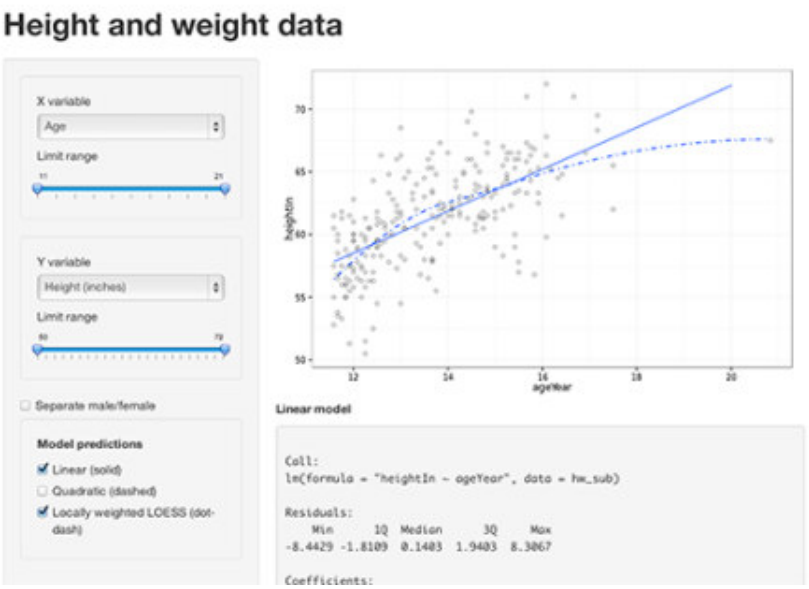
O Shiny é composto de:

Interface (*UI*): Controla o *layout* e a aparência da aplicação. Nele pode ser incluído linguagem de marcação de hipertexto(HTML), folhas de estilo em cascata (CSS), linguagem orientada a objetos (JavaScript) e biblioteca de funções JavaScript que interage

[6] http://rstudio.org/
[7] https://shiny.rstudio.com/

com o HTML (jQuery) e documentos interativos que contêm janelas, botões, ícones, menus, barras de rolagem e outras funcionalidades Shiny incorporados (BEELEY, 2013). A Tabela 4, Anexo C, detalha as funções utilizadas na interface.

Figura 18 – Shiny



Fonte: Cheng (2012)

Servidor (*Server*): Tem a função de processar as requisições e dados da interface, interagir com banco de dados, renderizar as páginas web. A Tabela 5, Anexo C, detalha as funções utilizadas camada do servidor.

shinyApp: Cria objetos do aplicativo Shiny a partir do par de interface/servidor.

As Tabela 6 e 7, Anexo D, descrevem o conjunto de dispositivos de saída, Render() e Output(), utilizados no Shiny.

Como visto neste capítulo, a aprendizagem é um processo em que conhecimento, habilidades e competência são adquiridos por meio do estudo, experiência e novas descobertas. É um processo de observações, experimentação e comparação teóricas. Os computadores também por meio de experiência obtida na execução de tarefas também conhecer aprender. Para isso utiliza de modelos preditivos para obter conhecimento a partir de relações matemáticas acerca dos dados empregados e assim propor soluções para problemas existentes.

Estes modelos são classificados segundo sua forma de execução e a natureza do que se propõe resolver. Basicamente são funções matemáticas e a escolha do algoritmo a ser usado é guiado pelos dados que se tem e o problema imposto.

3 O ambiente LAWSMiner

Este capítulo apresenta detalhes do ambiente interativo de visualização, análise e mineração de dados, LAWSMiner, sua arquitetura, interface, funcionalidades e implantação. É detalhado o ciclo de desenvolvimento, linguagem, frameworks e plugins, e demonstrados análises e gráficos realizados pelo ambiente no âmbito do Laboratório de Sistemas Avançados da Web (*Laboratory of Advanced Web Systems* – LAWS) da Universidade Federal do Maranhão.

O objetivo deste trabalho é criar e validar um ambiente interativo para automatização do processo de mineração de dados orientado ao usuário final, utilizável também como campo de prática para alunos de ciências de dados, capaz de auxiliar na exploração, mineração, análise e visualização de dados, garantindo eficiência, agilidade, confiabilidade e produtividade, além de um facilitador e potencializador no ensino e aprendizagem de Ciência de Dados.

O LAWSMiner é um ambiente web, interativa, *open source*, desenvolvido na linguagem de programação R, utilizando o padrão de projeto MVC (*Model-view-controller*), e interface *web*, plotagem de gráficos e tabelas no pacote Shiny.

O ambiente possibilita a execução dos algoritmos de aprendizagem, a análise dos gráficos e respostas, e a utilização por especialistas e pesquisadores para a análise de dados de negócios e dados de pesquisas, ajudando na descoberta de conhecimento, assim como, alunos que desejam aprofundar os conhecimentos executando os conceitos e técnicas aprendidas em sala de aula.

3.1 Requisitos Funcionais

- **Seleção e importação dos Arquivos de Dados**

  - Selecionar e importar os arquivos de dados nos formatos .cvs, .rdata, .rds e .arff.

- **Seleção de Atributos do Arquivo de Dados**

  - Permitir que após importado o arquivo de dados, o usuário possa selecionar quais atributos do arquivo serão utilizados na análise de dados.

- **Análise Exploratória de Dados**

  - Gerar o sumário dos atributos do arquivo de dados, mostrando as médias, quartis, máximos e mínimos;

- Gerar a correlação estatísticas dos atributos mostrando as dependências ou associação entres eles;

- Gerar diagrama *boxplot* para visualizar a distribuição e valores discrepantes (*outliers*) dos dados;

- Gerar distribuição de frequência (histogramas) do conjunto de dados;

- Gerar graficamente as correlações dos dados de forma a visualizar quais estão associadas diretamente.

- Gerar Análise de Componentes Principais para verificar a covariância entre os atributos.

- **Algoritmos de Regressão**

  - Implementar o algoritmo de aprendizagem de Regressão Linear;

  - Gerar o gráfico de Dispersão, Barras, Residuais e Normal Q-Q Plot das variáveis;

  - Gerar o modelo aprendizagem;

  - Permitir ao usuário a escolha do percentual do arquivo de dados destinado ao treino e ao teste do modelo;

  - Mostrar no treinamento os modelos gerados, os gráficos de dispersão antes e após o treinamento e os valores preditos;

  - Permitir a predição de novos valores, carregando um novo arquivo de dados para predição.

- **Algoritmos de Agrupamento**

  - Implementar o algoritmo de aprendizagem K-Means Clustering;

  - Permitir a escolha por parte do usuário da quantidade de cluster;

  - Gerar o gráfico com os centroides, a relação deles e mostrar os registros com seus respectivos centroides.

- **Algoritmos de Classificação**

  - Implementar os algoritmos de classificação Naïve Bayes, Máquina de Vetores de Suporte, K Vizinhos Mais Próximos, Árvore de Decisão e Regressão Logística;

  Para todos os algoritmos:

  - Implementar a geração do modelo;

  - Gerar gráfico de resultado da classificação

  - Permitir ao usuário a escolha do percentual do arquivo de dados destinado ao treino e ao teste do modelo;

- Permitir a predição de novos valores, carregando um novo arquivo de dados para predição.

- Mostrar no treinamento os modelos gerados, Matriz de Confusão e os valores preditos;

- Para o algoritmo Naïve Bayes mostrar no treinamento os modelos gerados, Matriz de Confusão e os valores preditos;

- Para o algoritmo Máquina de Vetores de Suporte permitir a escolha do kernel no treinamento;

- Para o algoritmo K Vizinhos Mais Próximos permitir a escolha do número de vizinhos para treinamento;

- Para o algoritmo Árvore de Decisão mostrar no treinamento os modelo gerado e os valores preditos;

= Para o algoritmo Regressão Logística permitir a escolha da Tendência para treinamento.

3.2 Arquitetura

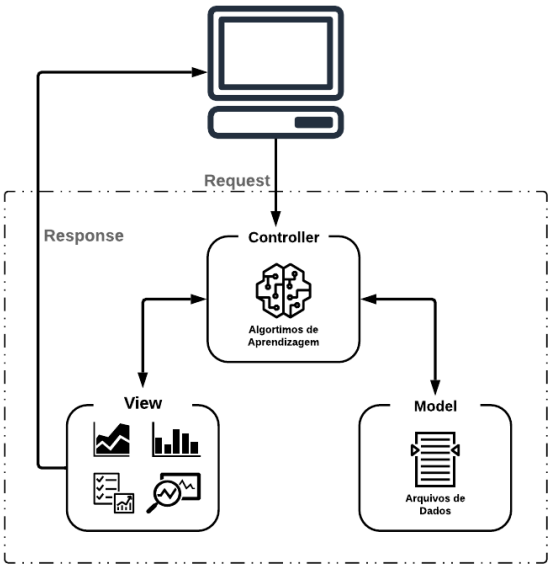
O LAWSMiner foi implementado utilizando o padrão de projeto MVC, detalhado na Figura 19, onde a camada Modelo (*Model*), camada que apresenta os dados da aplicação e que provê meios de acesso, validação e leitura dos dados, foi codificada no arquivo server.R. A Visão (*View*), camada responsável pelas interações do software com o usuário, e que apresenta os dados provenientes do modelo, a partir das interações do Controlador, foi codificada no arquivo ui.R e utilizado o pacote Shiny e a camada Controle (*Controller*), responsável pela realização das ações iniciadas pela interação do usuário na camada de Visão, ou por alterações no Modelo de dados e é responsável por refletir e manter a consistência dos dados armazenados no Modelo e na camada de Visão, foi implementada no arquivo global.R.

A Figura 20 detalha a arquitetura de implementação do LAWSMiner, onde explicita o processo utilizado no ambiente.

Importação de Dados: A entrada dos dados é dada por meio da seleção e carga de arquivo de dados, podendo ser local ou ainda por meio de endereço eletrônico (*link*). Possui suporte inicial para os arquivos de extensão CSV (*Comma-separated values* - Valores separados por vírgula), rdata e rds (extensão de arquivo usada pelo ambiente de desenvolvimento de software R) e arff (*Attribute-Relation File Format* - formato utilizado pelo Weka).

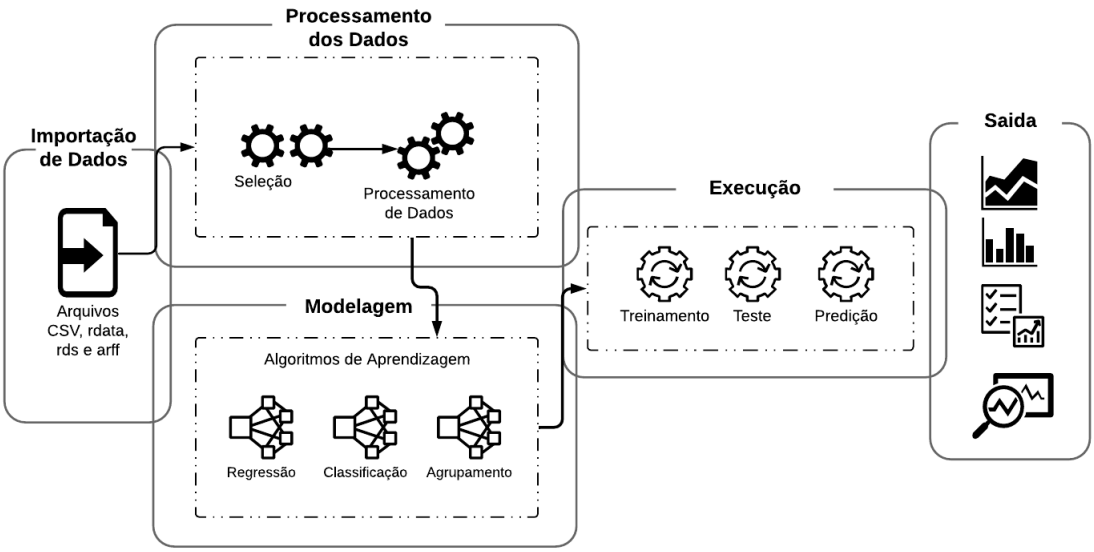
Processamento de Dados: Nesta etapa os dados carregados são processados e dispostos em tabelas possibilitando ao usuário a seleção e a remoção das colunas que

Figura 19 – Componentes MVC



Fonte: Elaborado pelo autor.

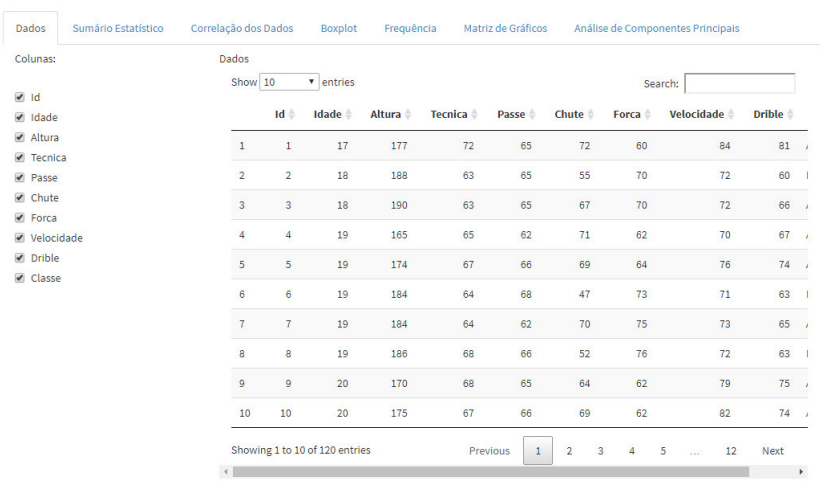
Figura 20 – Arquitetura LAWSMiner



Fonte: Elaborado pelo autor.

não são representativas para ele. Também são disponibilizadas tabelas e gráficos para a realização da análise exploratória dos dados. A Figura 21 mostra os dados carregados no ambiente e ao lado as colunas existentes. Esta funcionalidade possibilita ao usuário a retirada do *dataset* a ser analisado, os campos que não trazem relevância a análise, ou que, de alguma forma podem gerar confusão de entendimento ou engano (TUKEY, 1977). Um exemplo são os campos de identificador de registro (*id*) que de forma geral não contribui com informações relevantes para o esclarecimento dos dados.

Figura 21 – Análise Exploratória de Dados



Fonte: LAWSMiner.

O ambiente disponibiliza também, para auxiliar na análise exploratória, o sumário estatístico dos dados, a correlação entre as colunas, diagrama de caixa (*boxplot*), distribuição de frequência, matriz de gráficos e a análise dos componentes principais do *dataset*.

Modelagem: O processo de modelagem de dados disponibiliza ao usuário, após a análise exploratória dos dados e conhecendo como estes se apresentam e a relação entre eles, a opção entre as subcategorias de aprendizagem disponíveis, Regressão, Agrupamento ou Classificação, o algoritmo mais adequado para a operação e os dados.

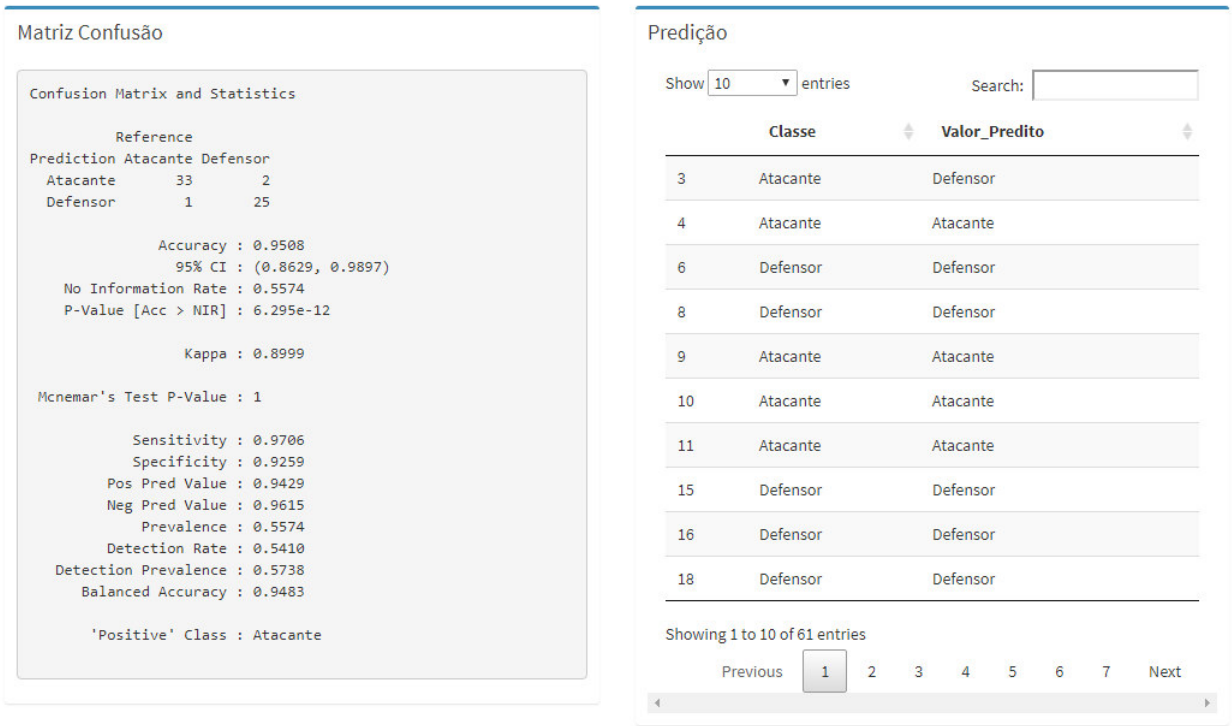
Execução: No processo de execução é realizada a mineração propriamente dita. Nela ocorre de padrões e correlações nos conjuntos de dados e é gerado o modelo analítico, que aprende com os dados de entrada e é utilizado para prever resultados.

A Figura 22 detalha o resultado de uma predição com os dados de testes do *Dataset* Jogadores (Apêndice B). Nela é apresentada a matriz de confusão, onde indica principalmente a acurácia do modelo, que mede a proximidade entre o valor predito ao valor verdadeiro.

Saída: A geração de matriz, gráficos, tabelas auxiliam a análise e o comportamento dos dados. A Figura 23 mostra o gráfico resultante do algoritmo SVM (*support vector machine*) aplicado a variável categórica Classe nas dimensões Altura e Técnica do *dataset* Jogadores, onde delimita a classificação dos jogadores Defensores e atacantes. Observa-se que, no conjunto de dados usados, os defensores são predominantemente de maior altura, e os atacantes estão concentrados na faixa de menor altura mais de uma técnica maior que os defensores.

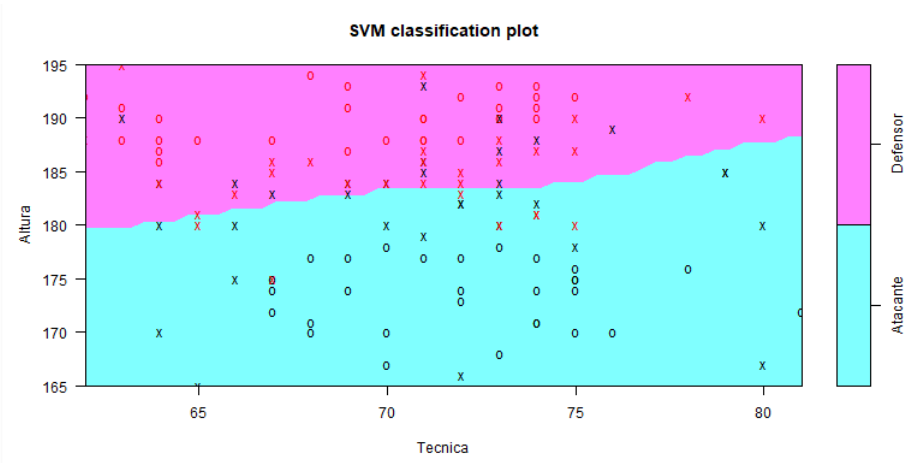
O conhecimento obtido após a análise e os pontos que foram disponibilizados, trazem o entendimento dos dados que apenas como o conjunto de dados estavam obscuros ou sem o entendimento apropriado.

Figura 22 – Matriz Confusão e Predição do algoritmo Naive Bayes



Fonte: LAWSMiner.

Figura 23 – Gráfico SVM. Classificação Classe tipo de jogador, nas dimensões de Altura e Técnica



Fonte: LAWSMiner.

3.3 Funcionalidades

O LAWSMiner é um ambiente de visualização, análise exploratória e mineração de dados que utilizou como referência na construção de suas funcionalidades as etapas descritas no processo de descoberta do conhecimento proposto por Fayyad et al. (1996). Cada uma destas etapas do processo está representada nele, sendo a mineração de dados o fator mais importante e de maior potencial e abrangência.

3.3.1 Análise Exploratória de Dados

Segundo Tukey (1977), a tarefa de análise exploratória é de grande importância na busca do conhecimento é o primeiro passo, a pedra fundamental. Mas para isso, o analista de dados precisa de ferramentas certas e de conhecimento técnico para desvendar as informações contidas nos dados e saber dentre elas quais são acidentais ou enganosas.

O LAWSMiner utiliza de tabelas, gráficos para explicitar os dados, e nele está implementado os seguintes pontos:

Sumário Estatístico: Resumo estatístico dos dados que detalha os quartis, os valores máximos e mínimos e a média dos valores de uma coluna para as variáveis contínuas, e a relação e quantidade de observações para as variáveis discretas. A Figura 24 demonstra o sumário gerado.

Figura 24 – Sumário Estatístico

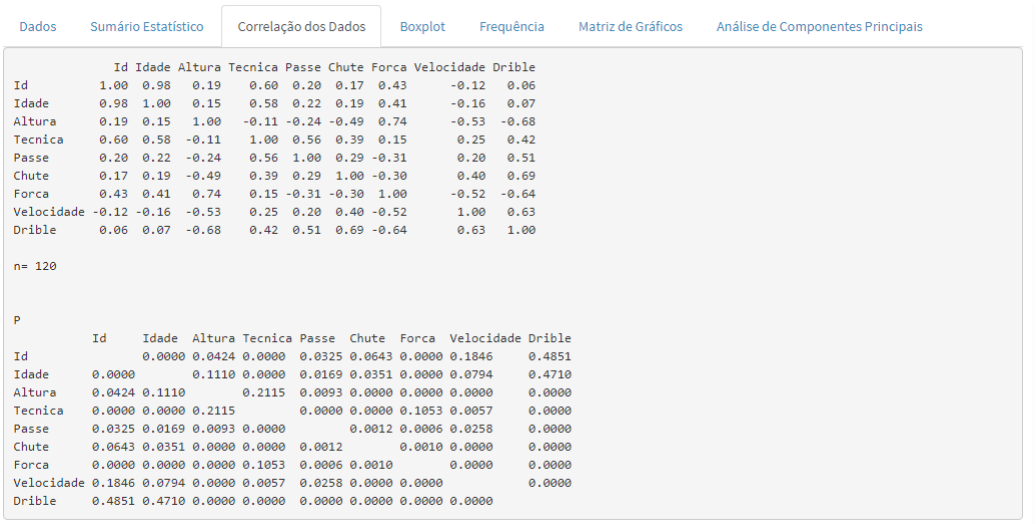
Fonte: LAWSMiner.

Correlação dos Dados: Outro ponto gerado é a correlação dos dados. Ela estabelece o grau de dependência entre duas variáveis, ou seja, quando a alteração no valor de uma variável (dita independente) provoca alterações no valor da outra variável (dita dependente). A Figura 25 mostra a relação entre as variáveis do *dataset* Jogadores.

Diagrama de caixa (*boxplot*): Diagrama de caixa ou *boxplot*, é uma ferramenta gráfica que permite visualizar a distribuição e valores discrepantes (*outliers*) dos dados, fornecendo um meio complementar para desenvolver uma perspectiva sobre o caráter dos dados. Além disso, o *boxplot* também é uma disposição gráfica comparativa. A Figura 26 mostra as medidas de estatísticas descritivas como o mínimo, máximo, primeiro quartil, segundo quartil ou mediana e o terceiro quartil.

Diagrama de Frequência: O diagrama de frequência ou histograma é um gráfico de barras que mostra a frequência de um determinado valor num intervalo de classe, de

Figura 25 – Correlação de Dados



	Id	Idade	Altura	Tecnica	Passe	Chute	Forca	Velocidade	Drible
Id	1.00	0.98	0.19	0.60	0.20	0.17	0.43	-0.12	0.06
Idade	0.98	1.00	0.15	0.58	0.22	0.19	0.41	-0.16	0.07
Altura	0.19	0.15	1.00	-0.11	-0.24	-0.49	0.74	-0.53	-0.68
Tecnica	0.60	0.58	-0.11	1.00	0.56	0.39	0.15	0.25	0.42
Passe	0.20	0.22	-0.24	0.56	1.00	0.29	-0.31	0.20	0.51
Chute	0.17	0.19	-0.49	0.39	0.29	1.00	-0.30	0.40	0.69
Forca	0.43	0.41	0.74	0.15	-0.31	-0.30	1.00	-0.52	-0.64
Velocidade	-0.12	-0.16	-0.53	0.25	0.20	0.40	-0.52	1.00	0.63
Drible	0.06	0.07	-0.68	0.42	0.51	0.69	-0.64	0.63	1.00

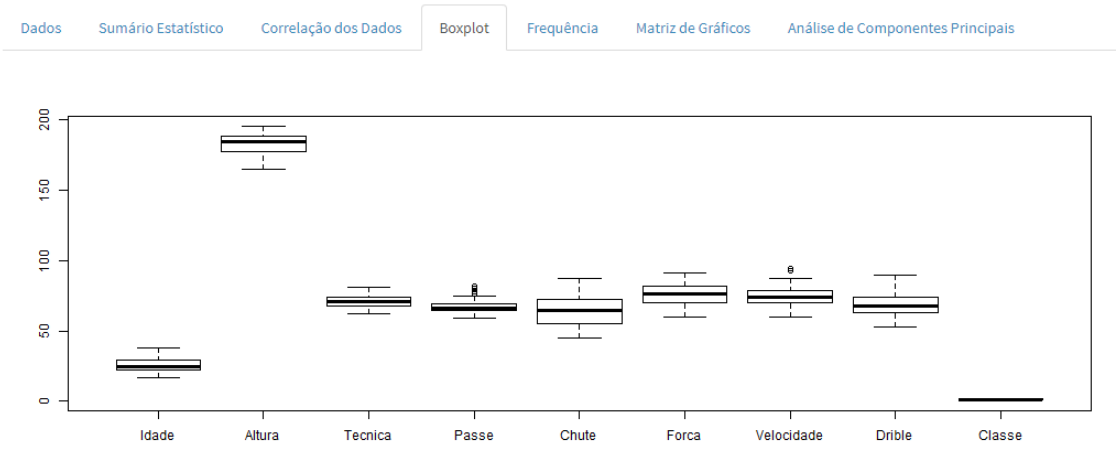
n= 120

P

	Id	Idade	Altura	Tecnica	Passe	Chute	Forca	Velocidade	Drible
Id		0.0000	0.0424	0.0000	0.0325	0.0643	0.0000	0.1846	0.4851
Idade	0.0000		0.1110	0.0000	0.0169	0.0351	0.0000	0.0794	0.4710
Altura	0.0424	0.1110		0.2115	0.0093	0.0000	0.0000	0.0000	0.0000
Tecnica	0.0000	0.0000	0.2115		0.0000	0.0000	0.1053	0.0057	0.0000
Passe	0.0325	0.0169	0.0093	0.0000		0.0012	0.0006	0.0258	0.0000
Chute	0.0643	0.0351	0.0000	0.0000	0.0012		0.0010	0.0000	0.0000
Forca	0.0000	0.0000	0.0000	0.1053	0.0006	0.0010		0.0000	0.0000
Velocidade	0.1846	0.0794	0.0000	0.0057	0.0258	0.0000	0.0000		0.0000
Drible	0.4851	0.4710	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

Fonte: LAWSMiner.

Figura 26 – Diagrama de caixa (*boxplot*)



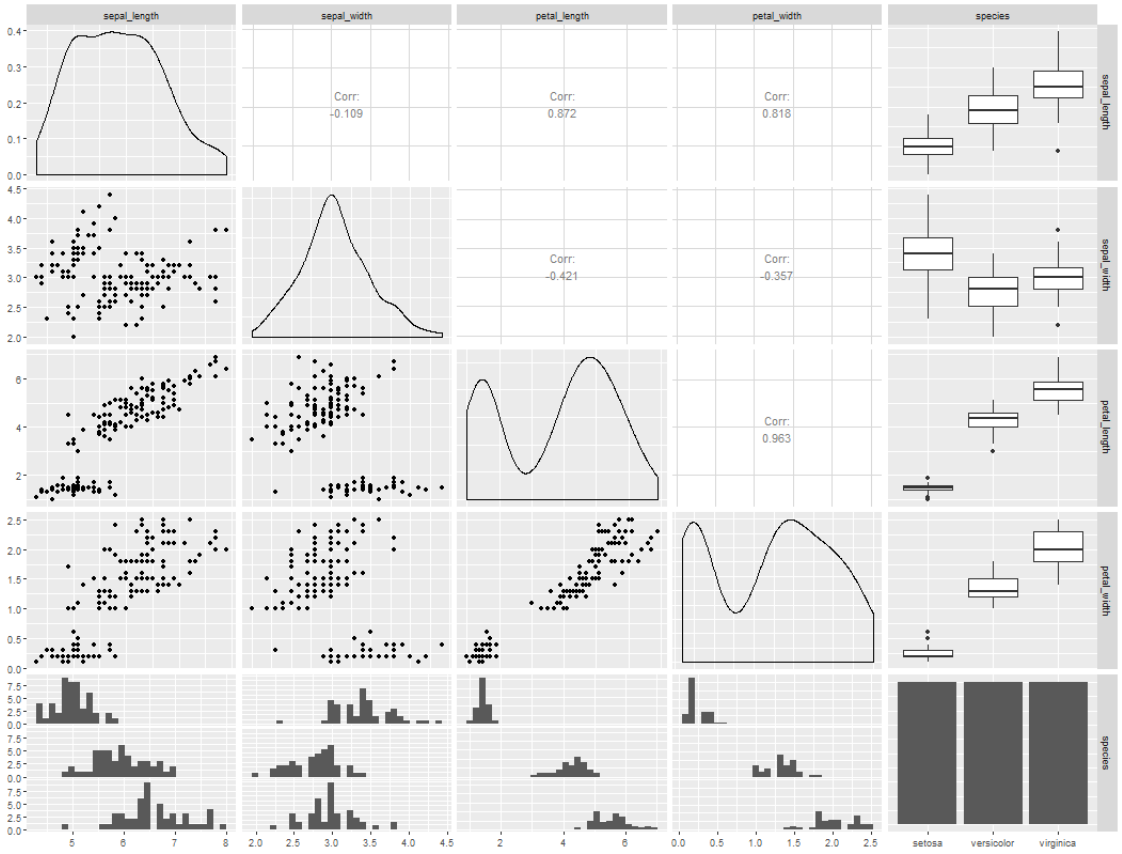
Fonte: LAWSMiner.

modo que é possível distinguir a forma, o ponto central e a variação da distribuição deles, além de diversas outras características, como a amplitude e a simetria nessa distribuição.

Matriz de Gráficos: A matriz de gráficos é um artifício visual onde se tem o comportamento dos dados e as correlações entre eles. Desta forma pode-se, visualmente, verificar como se distribuem, par a par, e assim facilitar a escolha dos algoritmos de aprendizagem mais apropriados.

PCA - Análise de Componentes Principais: Jolliffe (2002) define PCA (*Principal Component Analysis*) como uma transformação linear ortogonal que transforma os dados para um novo sistema de coordenadas de forma que a maior variância fica ao longo da primeira coordenada (o chamado primeiro componente), a segunda maior variância fica ao longo da segunda coordenada, e assim por diante. Ela encontra as formas mais representativa dos dados, os componentes principais do *dataset*, a partir das combinações

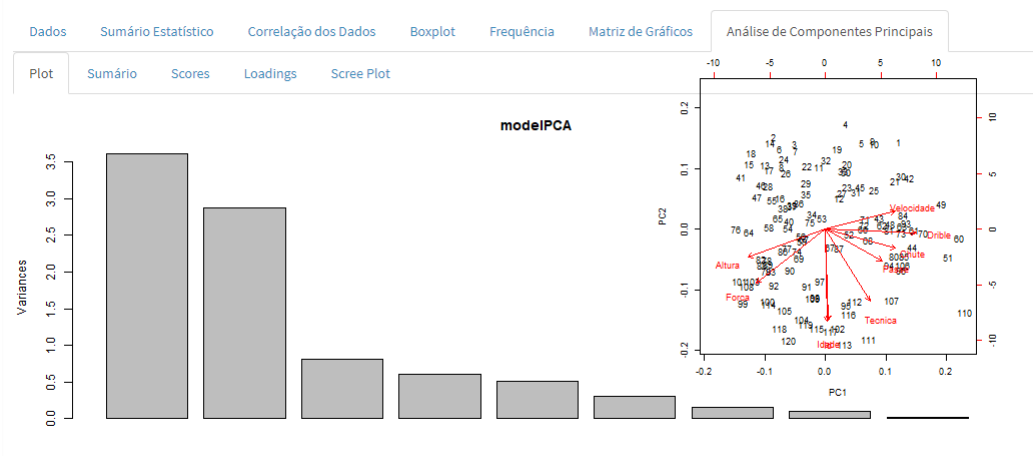
Figura 27 – Matriz de Correlação de Dados



Fonte: LAWSMiner.

lineares das variáveis originais.

Figura 28 – Análise de Componentes Principais



Fonte: LAWSMiner.

3.3.2 Algoritmos Implementados no ambiente LAWSMiner

O reconhecimento de padrões é a principal tarefa da mineração de dados, cujo objetivo é construir sistemas que possam distinguir e classificar objetos (BEYERER;

RICHTER; NAGEL, 2018). Os algoritmos são parte importante no processo, haja vista, que são os responsáveis por procurar dentro do conjunto de dados os padrões existentes.

Os algoritmos de aprendizagem apresentam características próprias resultantes dos seus princípios de funcionamento que por sua vez determinam sua categoria (regressão, classificação, agrupamento) e abordagem (supervisionado, semi-supervisionado, não supervisionado).

Quanto ao funcionamento, se diferenciam por especificidades, tal como, natureza da tarefa, a quantidade e os tipos de dados (rotulados ou não), o tempo e recursos computacionais que precisam, o tipo de resposta, e o comportamento, pois uns são mais sensíveis a ruídos, outros se adaptam a eles, e ainda os que precisam de muito mais dados para ter resultados relevantes. O tempo de execução também diferencia os algoritmos, haja vista que, para determinada tarefa o tempo de processamento é um fator importante.

O LAWSMiner implementa algoritmos das subcategorias regressão, classificação e agrupamento, conforme a seguir.

3.3.2.1 Algoritmos de Regressão

Regressão é empregada para modelar e analisar a relação entre uma variável dependente (resposta) e uma ou mais variáveis independentes (preditoras).

Algoritmos de Regressão são algoritmos de aprendizagem supervisionada que é empregada para modelar e analisar a relação entre uma variável dependente (resposta) e uma ou mais variáveis independentes (preditoras). Por ser supervisionada possuem também duas etapas de execução: Uma etapa de treinamento e outra de teste, para então realizar a predição.

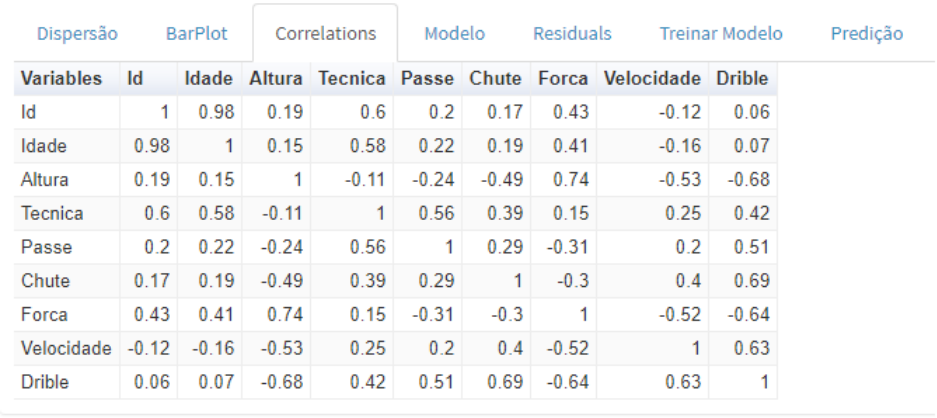
Regressão Linear: Na Regressão Linear, o algoritmo gera uma equação que descreve a relação estatística entre a variável alvo, contínua, a uma ou mais variáveis preditoras.

No ambiente LAWSMiner, o usuário escolhe, dentre as variáveis contínuas, qual será a variável alvo da sua regressão. Ele processa a solicitação e retorna ao usuário, em relação a variável alvo, a correlação entre as colunas do *dataset* (Figura 29), o gráfico de dispersão, o gráfico de barras , o modelo da regressão (Figura 30), residuais, treino de Modelo e Predição.

Na funcionalidade de Treino do Modelo (Figura 31) é disponibilizada a opção de seleção do percentual do *dataset* que será destinado para treino e teste do algoritmo, gerando o modelo que será utilizado para a predição dos novos valores.

Na predição o usuário seleciona o arquivo com os dados novos e realizar o processo. Como saída, é gerado o modelo utilizado e os valores preditos.

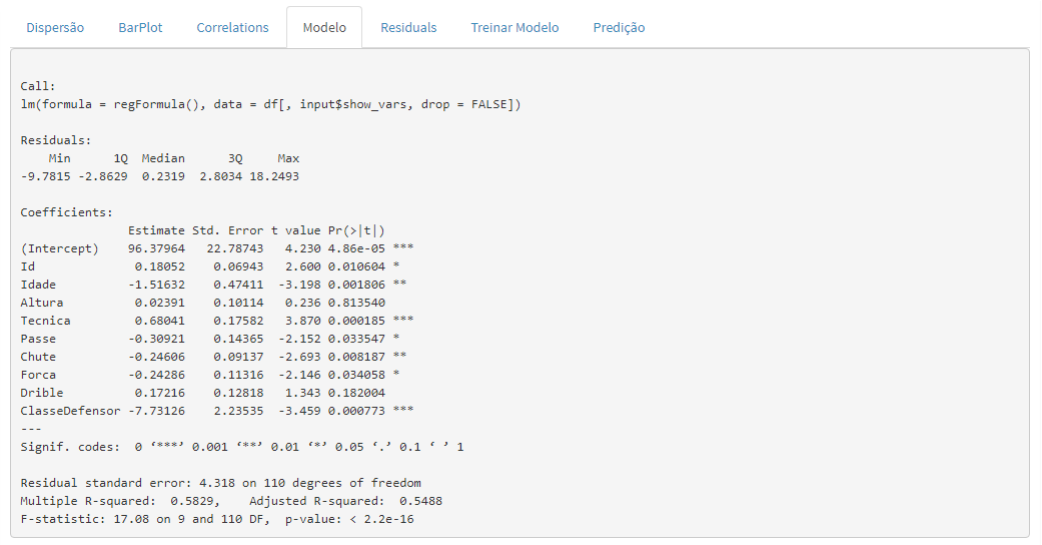
Figura 29 – Tabela de Correlação entre as variáveis



Variables	Id	Idade	Altura	Tecnica	Passe	Chute	Forca	Velocidade	Drible
Id	1	0.98	0.19	0.6	0.2	0.17	0.43	-0.12	0.06
Idade	0.98	1	0.15	0.58	0.22	0.19	0.41	-0.16	0.07
Altura	0.19	0.15	1	-0.11	-0.24	-0.49	0.74	-0.53	-0.68
Tecnica	0.6	0.58	-0.11	1	0.56	0.39	0.15	0.25	0.42
Passe	0.2	0.22	-0.24	0.56	1	0.29	-0.31	0.2	0.51
Chute	0.17	0.19	-0.49	0.39	0.29	1	-0.3	0.4	0.69
Forca	0.43	0.41	0.74	0.15	-0.31	-0.3	1	-0.52	-0.64
Velocidade	-0.12	-0.16	-0.53	0.25	0.2	0.4	-0.52	1	0.63
Drible	0.06	0.07	-0.68	0.42	0.51	0.69	-0.64	0.63	1

Fonte: LAWSMiner.

Figura 30 – Modelo da Regressão Linear



Fonte: LAWSMiner.

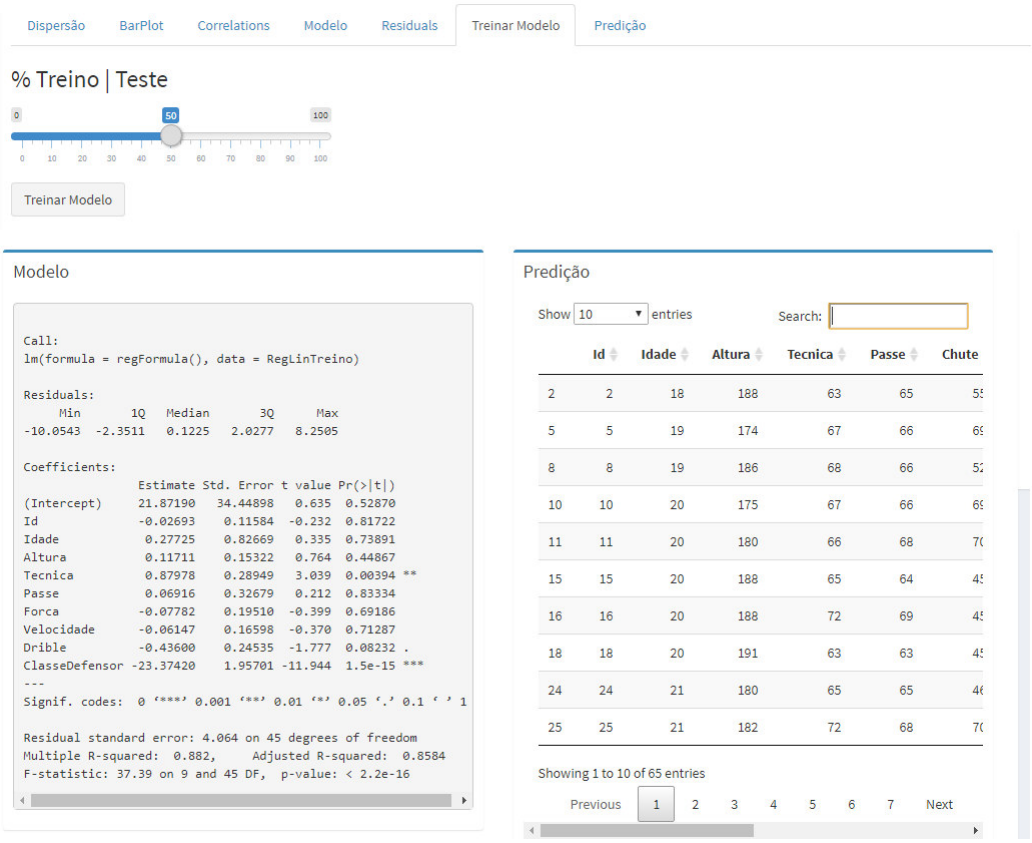
3.3.2.2 Algoritmos de Agrupamento

Algoritmos de agrupamentos são algoritmos não supervisionados que tem como objetivo separar objetos em grupos, baseando-se nas características que estes objetos possuem, colocando em um mesmo grupo, objetos que sejam similares de acordo com algum critério pré-determinado.

K-Means Clustering: é um algoritmo de agrupamento que objetiva particionar n observações dentre k grupos onde cada observação pertence ao grupo mais próximo.

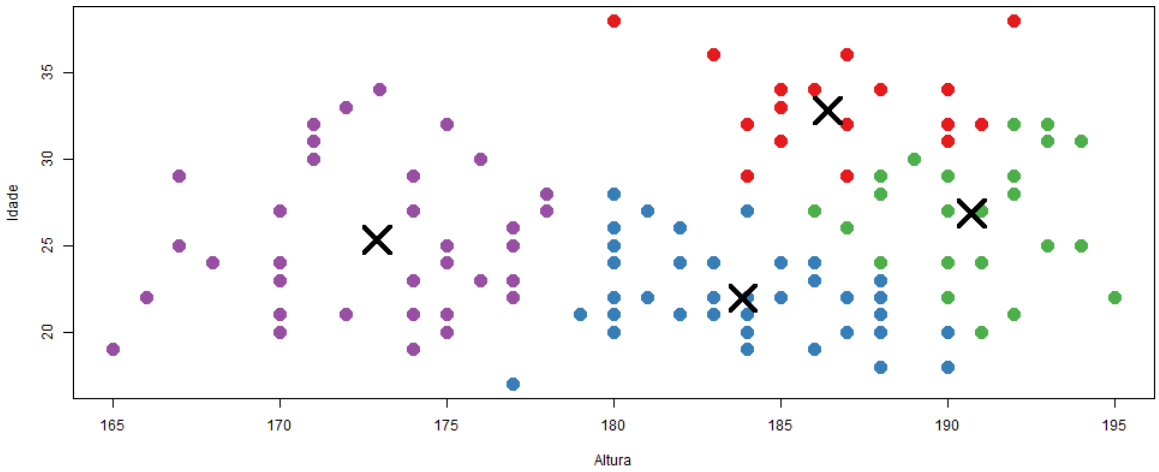
Para K-Means a ferramenta disponibiliza para o usuário a seleção do parâmetro K, que corresponde ao número de grupos que será dividido os registros do *dataset*.

Figura 31 – Treino Regressão Linear



Fonte: LAWSMiner.

Figura 32 – Agrupamento K Means Clustering



Fonte: LAWSMiner.

3.3.2.3 Algoritmos de Classificação

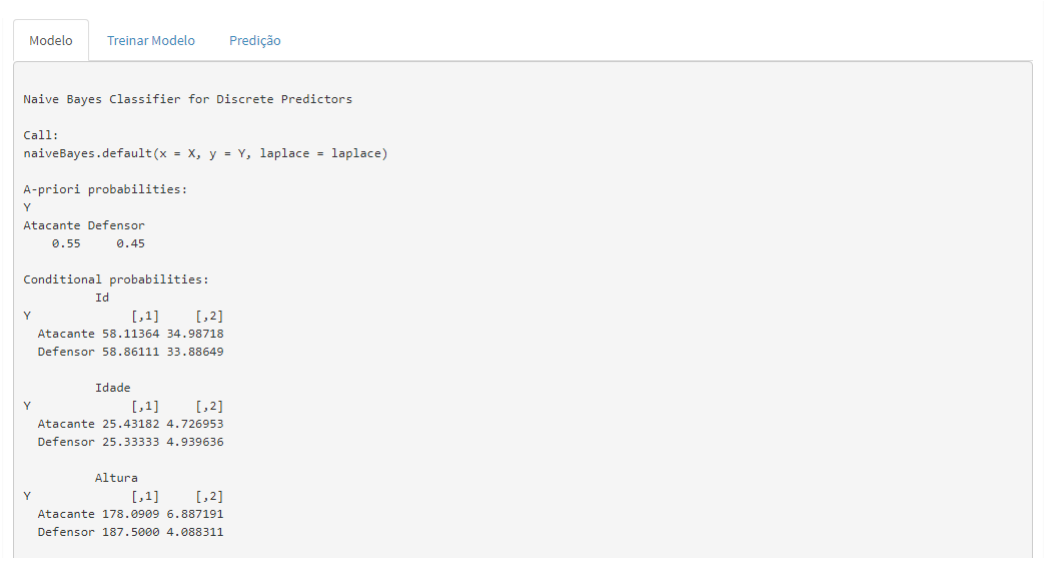
Algoritmos de Classificação são algoritmos supervisionados, cujo objetivo é aprender, a partir de uma correlação entre os atributos de entrada de tal maneira que para uma nova entrada não rotulada, o classificador seja capaz de determinar o rótulo vinculado a ela.

Naive Bayes: Segundo Follow (2016), os classificadores baseados em métodos

bayesianos utilizam dados de treinamento para calcular uma probabilidade observada de cada resultado com base nas evidências fornecidas pelos valores das características. Quando o classificador é aplicado posteriormente a dados não rotulados, ele usa as probabilidades observadas para prever a classe mais provável para os novos recursos.

Naïve Bayes, implementado no LAWSMiner, apresenta um modelo, como detalhado na Figura 33, onde mostra as probabilidades de ocorrência de cada item da classe. Há a possibilidade, na aba de Treinar Modelo, a possibilidade de ajustar o percentual do *dataset* usado para treinamento e teste, além de realizar a predição baseado aos modelos gerados nas etapas anteriores.

Figura 33 – Modelo de Classificação Naive Bayes com as opções de Treinar Modelo e Predição de novos valores



Fonte: LAWSMiner.

Máquinas de Vetores de Suporte: Por definição de Santhanam e Padmavathi (2015), o algoritmo Máquinas de Vetores de Suporte (*Support Vector Machines* - SVM) é um classificador que separa classes diferentes construindo entre elas hiperplanos em um espaço multidimensional. É um modelo de aprendizado supervisionado com algoritmos de aprendizado associados que analisam dados e reconhecem padrões.

O SVM procura o hiperplano ideal que separa duas classes maximizando a margem entre os pontos mais próximos entre elas. No espaço bidimensional os pontos nas margens são chamados de vetores de suporte e a linha que passa pelo ponto médio das margens é o hiperplano ideal.

Árvore de Decisão: A árvore de decisão é um modelo não paramétrico, que tem como estratégia de execução, a divisão dos dados em partes cada vez menores para identificar padrões que podem ser usados para previsão. É utilizada uma heurística chamada particionamento recursivo. Esta abordagem é geralmente conhecida como dividir

e conquistar, porque usa os valores dos recursos para dividir os dados em subconjuntos cada vez menores de classes semelhantes (LANTZ, 2013).

O modelo compreende uma série de decisões lógicas, com nós de decisão que indicam uma decisão a ser tomada em um atributo. Ele se divide em ramos que indicam as escolhas da decisão. A árvore é finalizada por nós folha, ou nós terminais, que denotam o resultado de seguir uma combinação de decisões. Os dados a serem classificados começam no nó raiz, onde são passados pelas várias decisões na árvore, de acordo com os valores de seus recursos. O caminho que os dados seguem canaliza cada registro em um nó folha, que atribui a ele uma classe prevista (RAMASUBRAMANIAN; SINGH, 2017).

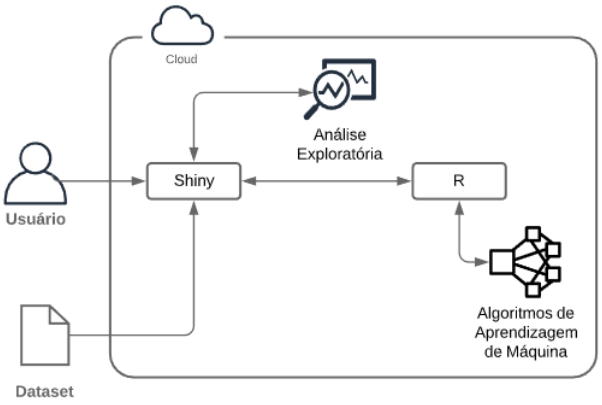
Regressão Logística: Conforme Géron (2019), alguns algoritmos de regressão podem ser utilizados para classificação, assim como, alguns de classificação também podem realizar uma regressão. A regressão logística é um exemplo disto. Ela é comumente usada para estimar a probabilidade de uma instância pertencer a uma classe específica. Se a probabilidade estimada for maior que 50%, o modelo prediz que a instância pertence a essa classe, classe positiva, rotulada com "1", ou então prediz que não pertence, classe negativa, rotulando-a como "0", logo a regressão logística é classificador binário.

3.4 Implantação na Nuvem

Considerando os ganhos de desempenho e de escabilidade, o ambiente LAWSMiner foi implantado em nuvem pública mantida pela DigitalOcean [1]. A instalação foi realizada em uma instância privada de máquina virtual chamada de Doplet.

A implantação na nuvem garante recursos sob demanda, aumento de hardware caso necessário, amplo acesso independentemente da plataforma. A Figura 34 detalha o funcionamento do LAWSMiner.

Figura 34 – Arquitetura da implantação na Nuvem



Fonte: Elaborado pelo autor

[1] www.digitalocean.com

O Usuário acessa o sistema através da camada do Shiny. Essa recebe as requisições e envia ao R, que por sua vez, dependendo da solicitação, acessa os algoritmos de aprendizagem e devolve para o Shiny ou apenas trata os dados retornando à camada anterior.

Em caso de Análise Exploratória de Dados, o Shiny desvia as solicitações para a camada específica, que recebe e trata as requisições. Havendo a necessidade de recursos do R, o Shiny realiza a intermediação.

4 Validação do ambiente LAWSMiner

O LAWSMiner tem em seu objetivo principal a criação de um ambiente interativo para a exploração, mineração, análise e visualização de dados, voltada a usuários de diversas áreas do conhecimento que buscam analisar os dados de suas pesquisas, experimentos e/ou negócios, mas que não desejam ou não possuem habilidades em programação de computadores, garantindo eficiência, agilidade, correção e produtividade. Contudo, por extensão, o ambiente LAWSMiner também é utilizado no ensino de Ciência de Dados, disponibilizando a alunos e professores um campo de prática para aplicação da teoria como suas aplicações e resultados.

A avaliação do ambiente foi realizada estabelecendo dois grupos de avaliadores: Alunos, para validar as características educacionais e pedagógicas; e especialistas e estatísticos, para avaliar os critérios de usabilidade, acurácia, coerência, conformidade e adequação dos gráficos, tabelas e matrizes.

Para tanto foi especificada uma atividade onde esses critérios poderiam ser observados e aplicado aos dois grupos. Ela era composta de um conjunto de dados composto por jogadores de futebol divididos em duas classes, Defensores e Atacantes. A base de dados era composta de 150 observações onde 120 tinham a classe definida. O objetivo da tarefa era predizer a classe dos outros 30 registros considerando as 120 conhecidas. A base de dados continha os seguintes atributos: idade, altura, técnica, passe, chute, força e velocidade, conforme mostra a Tabela 1

Tabela 1 – Dataset de Jogadores utilizado para a avaliação do ambiente LAWSMiner

Id	**Idade**	**Altura**	**Técnica**	**Passe**	**Chute**	**Força**	**Velocidade**	**Drible**	**Classe**
1	17	177	72	65	72	60	84	81	Atacante
2	18	188	63	65	55	70	72	60	Defensor
3	18	190	63	65	67	70	72	66	Atacante
4	19	165	65	62	71	62	70	67	Atacante
5	19	174	67	66	69	64	76	74	Atacante
6	19	184	64	68	47	73	71	63	Defensor
7	19	184	64	62	70	75	73	65	Atacante
8	19	186	68	66	52	76	72	63	Defensor
9	20	170	68	65	64	62	79	75	Atacante
10	20	175	67	66	69	62	82	74	Atacante
11	20	180	66	68	70	70	66	68	Atacante
12	20	184	73	65	79	79	77	71	Atacante
13	20	187	64	69	50	72	62	62	Defensor
14	20	188	62	64	47	70	72	64	Defensor
15	20	188	65	64	45	79	72	58	Defensor

Após a execução da tarefa utilizando o LAWSMiner, os entrevistados responderam os questionários de satisfação do usuário como forma de obter o *feedback* das percepções

acerca dele ((HAYES, 1998; FALEIROS et al., 2016)). O questionário foi composto por 12 perguntas para o grupo de alunos (Anexo B) e 9 questões para o grupo de especialistas (Anexo C).

Conforme a escala Likert, cada pergunta do questionário apresentava uma afirmação autodescritiva que pode ser respondida segundo a escala de pontos: Discordo Fortemente (- -), Discordo (-), Nem concordo e nem discordo (-/+), Concordo (+) e Concordo Fortemente (++). Com isso, é possível descobrir o grau de concordância com a frase, e consequentemente, os diferentes níveis de intensidade da opinião a respeito do LAWSMiner. O resultado obtido com a aplicação do questionário foi tabulado e mostrado na Tabela 2. As colunas +, ++ são de avaliações positivas, as - - e - negativas e o -/+ neutra.

Nesta seção são apresentados os resultados da avaliação dos grupos especificados acerca do LAWSMiner.

4.1 Avaliação Ensino e Aprendizagem

A avaliação do ambiente no âmbito educacional foi realizada pelos alunos de Inteligência Artificial do Curso de Ciência da Computação da UFMA. Na Tabela 2 é mostrado o resultado da avaliação realizada por estes.

Tabela 2 – Resultado do questionário de avaliação do ambiente LAWSMiner - Alunos (valores em %)

Questões Pesquisadas	**- -**	**-**	**-/+**	**+**	**++**
1. Possuo conhecimentos em Programação de Computadores	-	-	-	57,1	42,9
2. Possuo conhecimentos em Aprendizagem de Máquina	-	14,3	14,3	57,1	14,3
3. Conheço os algoritmos de aprendizado de máquina utilizados na ferramenta	-	-	28,6	57,1	14,3
4. Já implementei (em Python, R, entre outras) os algoritmos apresentados na ferramenta	28,6	14,3	28,6	28,6	-
5. A ferramenta apresenta um bom desempenho na execução das tarefas	-	14,3	-	85,7	-
6. Consegui executar com facilidade a atividade proposta	-	28,6	14,3	28,6	28,6
7. As respostas encontradas estavam coerentes com o propósito dos algoritmos	-	-	28,6	42,9	28,6
8. Os gráficos auxiliam na análise dos resultados de cada algoritmo	-	14,3	14,3	42,9	28,6
9. A abordagem empregada na ferramenta facilita o entendimento dos algoritmos	-	-	28,6	57,1	14,3
10. A ferramenta é útil para o aprendizado dos algoritmos de aprendizagem de máquina	-	-	-	85,7	14,3
11. Usaria a ferramenta para explorar os meus dados com frequência	14,3	-	57,1	-	28,6
12. Indicaria a ferramenta para quem quisesse estudar aprendizado de máquina	-	-	28,6	57,1	14,3

4.1.1 Resultado da Avaliação

Há cinco pontos principais do questionário que são relevantes para o escopo deste trabalho: o entendimento do aluno acerca da programação de computadores e sobre aprendizagem de máquina; a usabilidade e desempenho do ambiente LAWSMiner frente às atividades propostas em sala de aula; a avaliação quanto à adequação, acurácia e conformidade dos resultados; quanto ao efeito motivador e potencializador no estudo de ciência de dados; e referente ao seu uso para a exploração dos dados cotidianamente e a indicação para o estudo de aprendizagem de máquina.

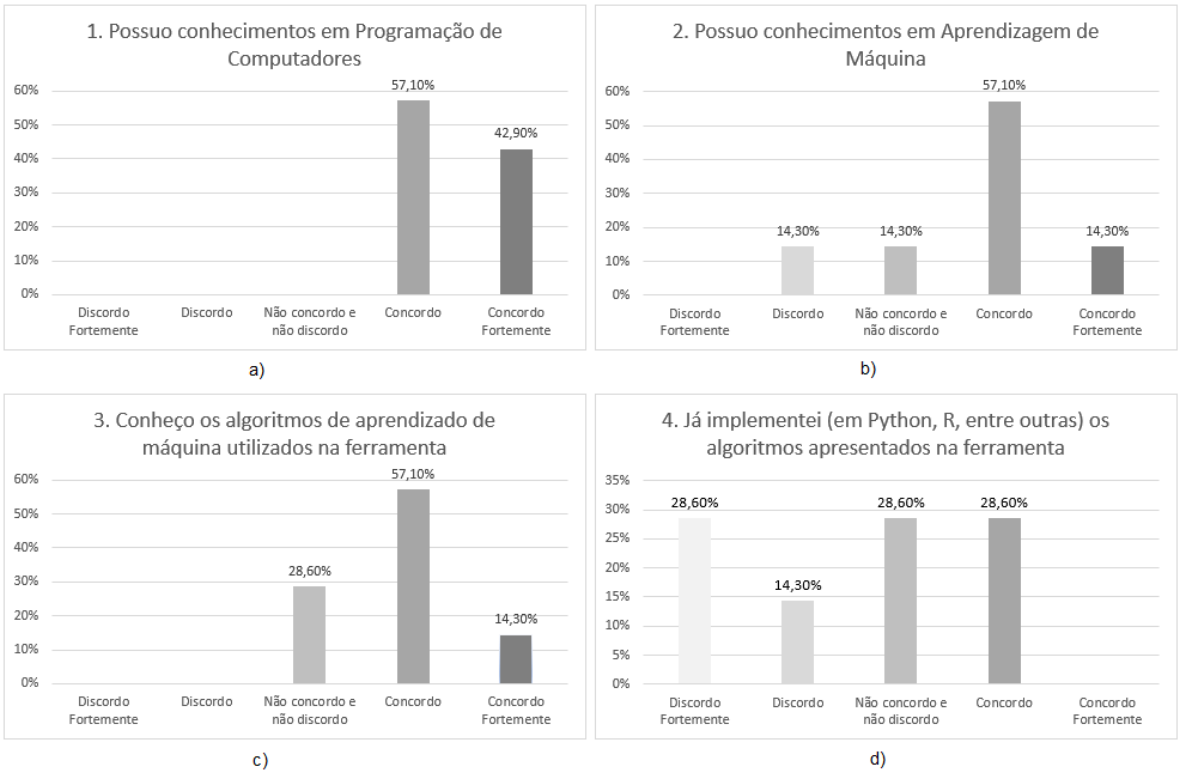
O primeiro ponto, conhecimentos prévios do aluno, o questionário se propôs a avaliar se o aluno reconheceria os esforços de implementação do LAWSMiner e se tinha embasamento teórico para avaliar o funcionamento e respostas geradas.

Sobre o conhecimento de programação de computadores (1. Possuo conhecimentos em Programação de Computadores), 100% dos alunos se autoavaliaram aptos, Figura 35a, com isso conseguem reconhecer a contribuição do LAWSMiner nos aspectos computacionais. A Figura 35b mostra a avaliação do aluno quanto ao seu conhecimento em aprendizagem de máquina (2. Possuo conhecimentos em Aprendizagem de Máquina), 71,4% são detentores do entendimento necessário, e apenas 14,3% dos entrevistados se avaliaram inaptos nesta questão, mesmo assim, no nível mais moderado da escala. A pergunta 3 (3. Conheço os algoritmos de aprendizado de máquina utilizados na ferramenta) avaliava o conhecimento do aluno acerca dos algoritmos de aprendizagem de máquina implementados. Neste item 71,4%, Figura 35c, se avaliou ter o entendimento necessário para opinar sobre a execução e sobre as respostas emitidas, enquanto 28% está na zona neutra da pergunta. A pergunta 4 (4. Já implementei (em Python, R, entre outras) os algoritmos apresentados na ferramenta) avalia os conceitos de programação em linguagens utilizadas na ciência de dados. Apenas 28% dos alunos, Figura 35d, já implementaram os algoritmos nessas linguagens.

O segundo ponto abordado é referente a usabilidade e a desempenho do LAWSMiner frente às atividades propostas em sala de aula. Neste grupo foi perguntado aos alunos se este foi satisfatório e se, dado a complexidade do problema a ser resolvido, o aluno conseguiu encontrar a solução facilmente. As questões 5 (5. A ferramenta apresenta um bom desempenho na execução das tarefas) e 6 (6. Consegui executar com facilidade a atividade proposta) avaliam o ambiente neste aspecto.

Quanto ao desempenho do LAWSMiner (Figura 36a), 85,7% dos entrevistados opinaram favoravelmente a esta afirmativa, enquanto 14,3% não perceberam o bom desempenho dela. Quanto a facilidade do aluno em executar as atividades propostas e alcançar os objetivos desejados (Figura 36b), 57,2% se mostraram satisfeitos, enquanto 14,3% não concordaram nem discordaram da afirmação e 14,3% não conseguiram executar a atividade com a facilidade desejada.

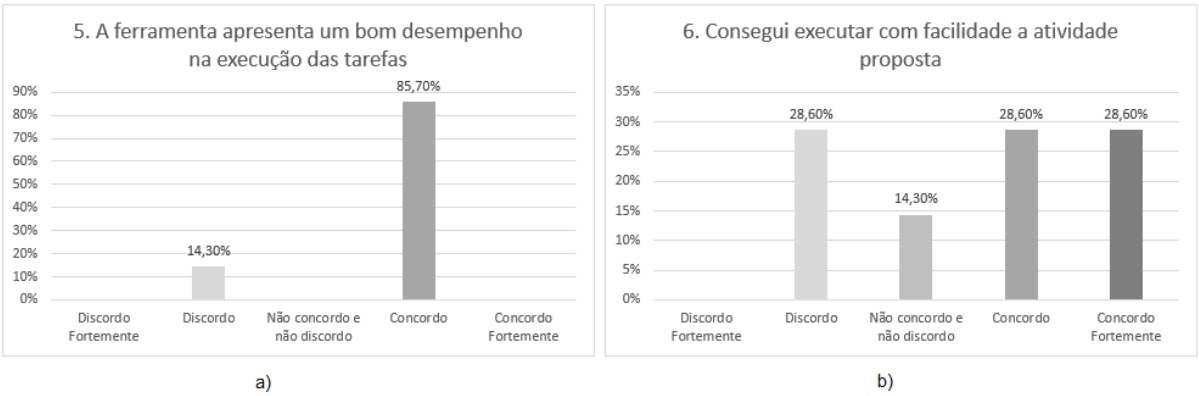
Figura 35 – Autoavaliação dos alunos referente a programação e algoritmos de aprendizagem



Fonte: Elaborada pelo autor

Neste ponto o ambiente alcançou índices bem favoráveis nos itens avaliados. A grande maioria dos entrevistados avaliaram que ele teve bom desempenho na realização das atividades, e facilidade de execução. O que demonstra que ele entrega nesses itens bons níveis de usabilidade e desempenho.

Figura 36 – Avaliação dos alunos referente ao desempenho e usabilidade do ambiente LAWSMiner



Fonte: Elaborada pelo autor

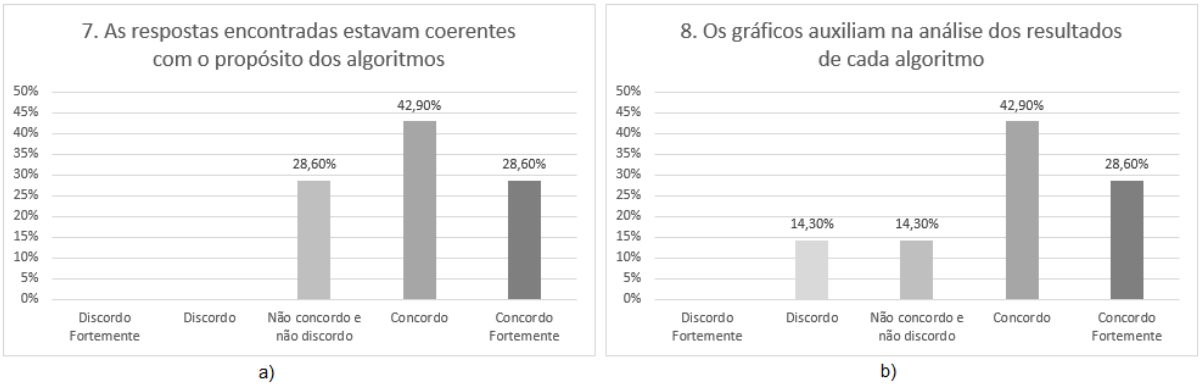
O terceiro ponto é referente a avaliação do ambiente quanto à adequação, acurácia e conformidade das funcionalidades. Foram avaliados neste grupo a forma com que o

LAWSMiner dispõe seus gráficos, listas e tabelas e se estes são coerentes com os algoritmos de aprendizagem utilizados. As questões 7 (7. As respostas encontradas estavam coerentes com o propósito dos algoritmos) e 8 (8. Os gráficos auxiliam na análise dos resultados de cada algoritmo), da Tabela 2, ressaltam a avaliação dos alunos nestes quesitos.

Quanto à conformidade, coerência e adequação das respostas apresentadas 71,5% (Figura 37a) dos entrevistados se mostraram satisfeitos e 28,6% não concordaram e nem discordaram com a afirmativa. No tocante a questão 8, também 71,5% (Figura 37b) dos entrevistados opinaram que os gráficos contribuem para entendimento e análise dos resultados de cada algoritmo.

Neste ponto o ambiente atendeu aos critérios de adequação, acurácia e conformidade esperados, se mostrado eficaz na análise de dados, segundo a grande maioria dos entrevistados.

Figura 37 – Avaliação dos alunos referente a coerência das respostas e gráficos gerados pelo LAWSMiner



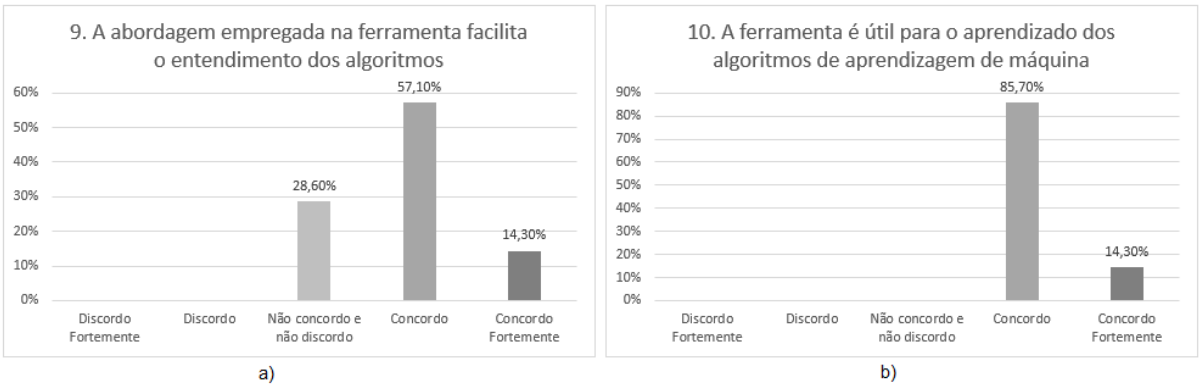
Fonte: Elaborada pelo autor

O quarto ponto analisado diz respeito às características motivadores e potencializadoras que o LAWSMiner traz ao estudo de ciência de dados, uma vez que facilita o entendimento dos algoritmos de aprendizagem. As questões 9 (9. A abordagem empregada na ferramenta facilita o entendimento dos algoritmos) e 10 (10. A ferramenta é útil para o aprendizado dos algoritmos de aprendizagem de máquina) colheram o sentimento dos alunos quanto a essas características.

A Figura 38 mostra a avaliação dos alunos quanto à capacidade do LAWSMiner de auxiliar no aprendizado de ciência de dados. Pela avaliação de mais de 70% dos alunos, Figura 38a, a ferramenta facilita o entendimento dos algoritmos de aprendizagem de máquina. E para 100% deles a LAWSMiner é uma importante aliada no ensino-aprendizagem, Figura 38b.

Considerando os níveis de aceitação nos quesitos deste grupo, o LAWSMiner correspondeu muito bem ao papel que foi proposto inicialmente, e consolida o papel de incentivador, facilitador e motivador na aprendizagem de ciência de dados.

Figura 38 – Avaliação dos alunos quanto o agente facilitador e potencializador do ambiente LAWSMiner na aprendizagem de Ciência de Dados



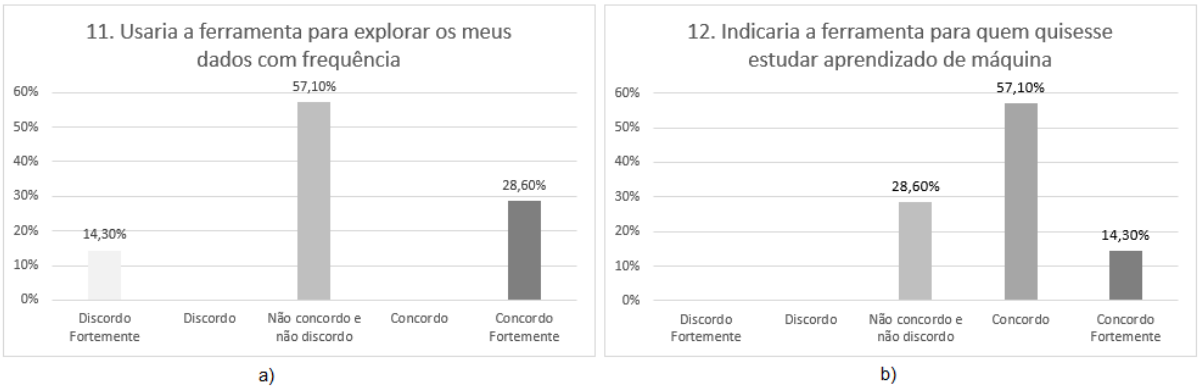
Fonte: Elaborada pelo autor

O quinto ponto avaliado pelos alunos diz respeito a aceitação do LAWSMiner como uma ambiente apropriado para a mineração de dados para uso frequente e se o aluno, por entender que esta é uma boa ferramenta para mineração, indicaria para outros usuários o seu uso. As questões 11 (Usaria a ferramenta para explorar os meus dados com frequência) e 12 (Indicaria a ferramenta para quem quisesse estudar aprendizagem de máquina) resumem a avaliação quanto ao índice de aceitação e indicação do LAWSMiner.

Na questão 11 (Figura 39a), 28% usariam a ferramenta com frequência para analisar dados fora do contexto acadêmico, sendo que apenas 14% julgaram o ambiente inapropriado para tal atividade, e 57,1% dos alunos não se sentiram aptos para avaliar essa questão.

Na questão 12 (Figura 39b), 71,40% dos alunos indicariam o LAWSMiner para fins de estudo de aprendizagem de máquina e enquanto 26,60% não tiveram uma concepção mais objetiva quanto a esta tarefa.

Figura 39 – Avaliação dos alunos referente ao uso frequente do ambiente LAWSMiner e sua indicação para o ensino de Aprendizagem de Máquina



Fonte: Elaborada pelo autor.

De forma geral o ambiente LAWSMiner se mostrou um forte aliado no ensino de aprendizagem de máquina. Na avaliação dos alunos ele possui bom desempenho, precisão

e adequação ao que se espera em um ambiente de ensino e aprendizagem em Ciência de Dados, pois motiva o aluno no aprofundamento e na assimilação dos conceitos da área, e que dentro das suas características educacionais pode ser utilizado sistematicamente em sala de aula.

4.2 Avaliação de Especialistas em Ciência de Dados

O segundo grupo de avaliadores do LAWSMiner foi composto por estatísticos e profissionais de ciência de dados. De forma semelhante à aplicada ao grupo de ensino, os especialistas executaram a atividade proposta e responderam as perguntas do questionário específico.

Para isso, foi disponibilizado no site da ferramenta na plataforma *Github*[1], a atividade, os arquivos de carga, e o link para o questionário. A Tabela 3 mostra o resultado da avaliação realizada por este grupo de avaliadores.

Tabela 3 – Resultado do questionário de avaliação do ambiente - Especialistas (valores em %)

Questões Pesquisadas	**- -**	**-**	**-/+**	**+**	**++**
1. A ferramenta tem uma boa usabilidade	-	-	15,4	61,5	23,1
2. As funcionalidades da ferramenta são claras e objetivas	-	7,7	23,1	46,1	23,1
3. A ferramenta se mostrou útil para análise de dados	-	7,7	7,7	30,8	53,8
4. A ferramenta agiliza a análise de dados	-	15,3	-	38,5	46,2
5. Os resultados fornecidos pela ferramenta são coerentes	-	-	23,0	38,5	38,5
6. Os conceitos apresentados pela ferramenta estão corretos	-	-	46,1	23,1	30,8
7. Os gráficos apresentados para cada algoritmo de aprendizado são adequados	-	7,7	-	61,5	30,8
8. Usaria a ferramenta para explorar os meus dados com frequência	7,7	7,7	-	53,8	30,8
9. Indicaria a ferramenta para quem quisesse estudar aprendizado de máquina	7,7	-	7,7	61,5	23,1

4.2.1 Resultado da Avaliação

O questionário de avaliação para usuários especialistas foi planejado com o intuito de validar no ambiente requisitos importantes dentro de uma perspectiva mais técnica, baseado nos requisitos de usabilidade, conformidade, correção, coerência, além de adequação dos conceitos estatísticos e de mineração de dados. Além disso, nesta fase também, foram avaliados os aspectos autoinstrucional em aprendizagem de máquina.

As duas primeiras perguntas do questionário dizem respeito a forma como o ambiente se apresenta ao usuário. Elas avaliam a facilidade com que o usuário executa as

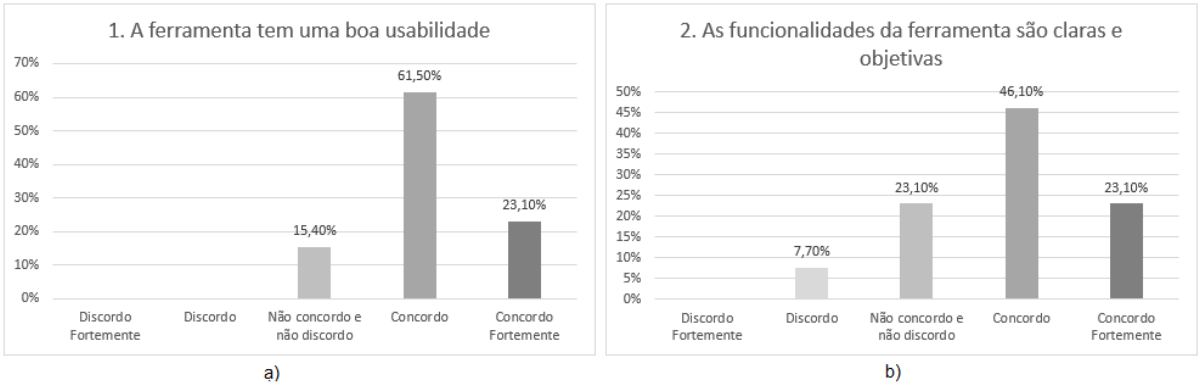
[1] https://lawsminer.github.io/

ações e atividades no ambiente, validando a usabilidade, a clareza e a objetividade das funcionalidades.

Para a primeira pergunta (1.A ferramenta tem uma boa usabilidade), 84,6% avaliaram que a ferramenta possui uma boa usabilidade, e 15,4% não souberam opinar com precisão (Figura 40a). Neste ponto o LAWSMiner se mostrou intuitivo e de fácil operacionalização, alcançando excelentes níveis de aceitação.

Para a segunda pergunta do questionário (2. As funcionalidades da ferramenta são claras e objetivas) cujo intuito era de avaliar a clareza e objetividade das funcionalidades implementadas no LAWSMiner, 69,2% responderam positivamente, sendo que 23,1% não concordaram nem discordaram da afirmação, e apenas 7,7% opinaram negativamente, conforme a Figura 40b. Para esse critério o ambiente novamente foi bem avaliado, se mostrando coeso e coerente na avaliação dos especialistas.

Figura 40 – Avaliação dos especialistas quanto a usabilidade, clareza e objetividade das funcionalidades do ambiente LAWSMiner



Fonte: Elaborada pelo autor.

A contribuição do LAWSMiner para a análise de dados é avaliada na terceira e quarta perguntas. Dentro da avaliação dos entrevistados estas perguntas visam, dentro do grau de satisfação, mensurar o nível de contribuição que a ferramenta tem à tarefa de análise de dados.

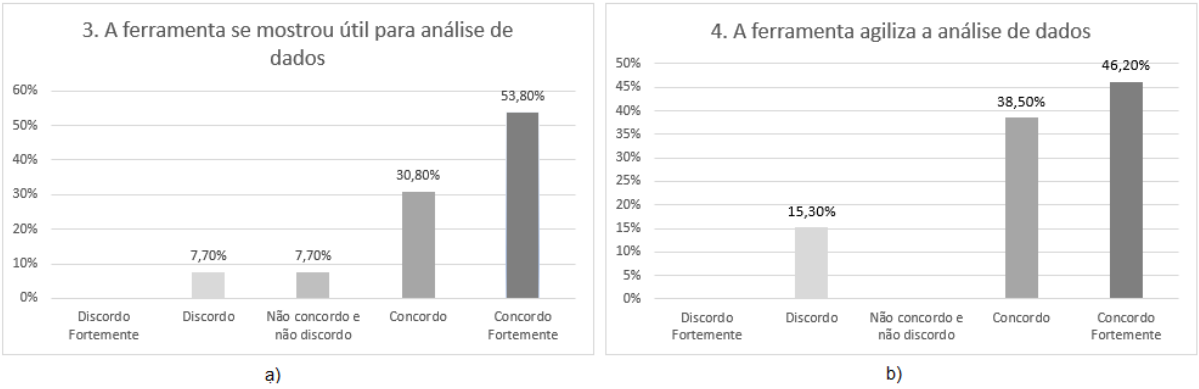
A primeira pergunta (3. A ferramenta se mostrou útil para análise de dados) objetiva verificar se o ambiente pode ser utilizado para análise de dados em âmbito profissional e não apenas como campo de prática de ensino. Dos entrevistados 84,6 % responderam positivamente à afirmação da questão e apenas 7,7 % não concordaram com ela, sendo que outros 7,7% não souberam responder com precisão. sendo que pelos entrevistados o ambiente pode ser utilizado com satisfação para análise de dados cotidianamente, conforme a Figura 41a.

A quarta pergunta (4. A ferramenta agiliza a análise de dados) discorre sobre a satisfação dos entrevistados na utilização do LAWSMiner para análise de dados de

forma geral. Nas respostas 84,70% avaliaram o ambiente como uma boa ferramenta para mineração de dados, sendo que apenas 15,30% dos entrevistados discordaram da afirmação da pergunta, conforme a Figura 41b.

Com a avaliação do LAWSMiner nestes dois quesitos, o ambiente foi aprovado satisfatoriamente como um instrumento que contribui na tarefa de mineração e análise de dados.

Figura 41 – Avaliação dos especialistas quanto ao uso da LAWSMiner como ambiente de mineração e análise de dados



Fonte: Elaborada pelo autor.

As próximas perguntas avaliam a correção e coerência dos conceitos utilizados e dos resultados gerados. Essas perguntas validam se as tabelas, gráficos e matrizes estão coerentes com os algoritmos de aprendizagem e se estes algoritmos geram as saídas que deveriam.

A quinta pergunta (5. Os resultados fornecidos pelas ferramenta são coerentes) verifica se há uma coerência entre saída produzida pelo LAWSMiner e o que se espera para cada caso. A pergunta abrange tabelas, gráficos, modelos, correlações, entre outros. Para esta pergunta 77% foram favoráveis a afirmativa, ou seja, após avaliar os resultados e confrontar com que se esperava, a grande maioria atestou a coerência das respostas, contudo, 28% não souberam responder, não concordaram nem discordaram na sua avaliação. Neste caso, o ambiente também foi bem avaliado, conforme a Figura 42a

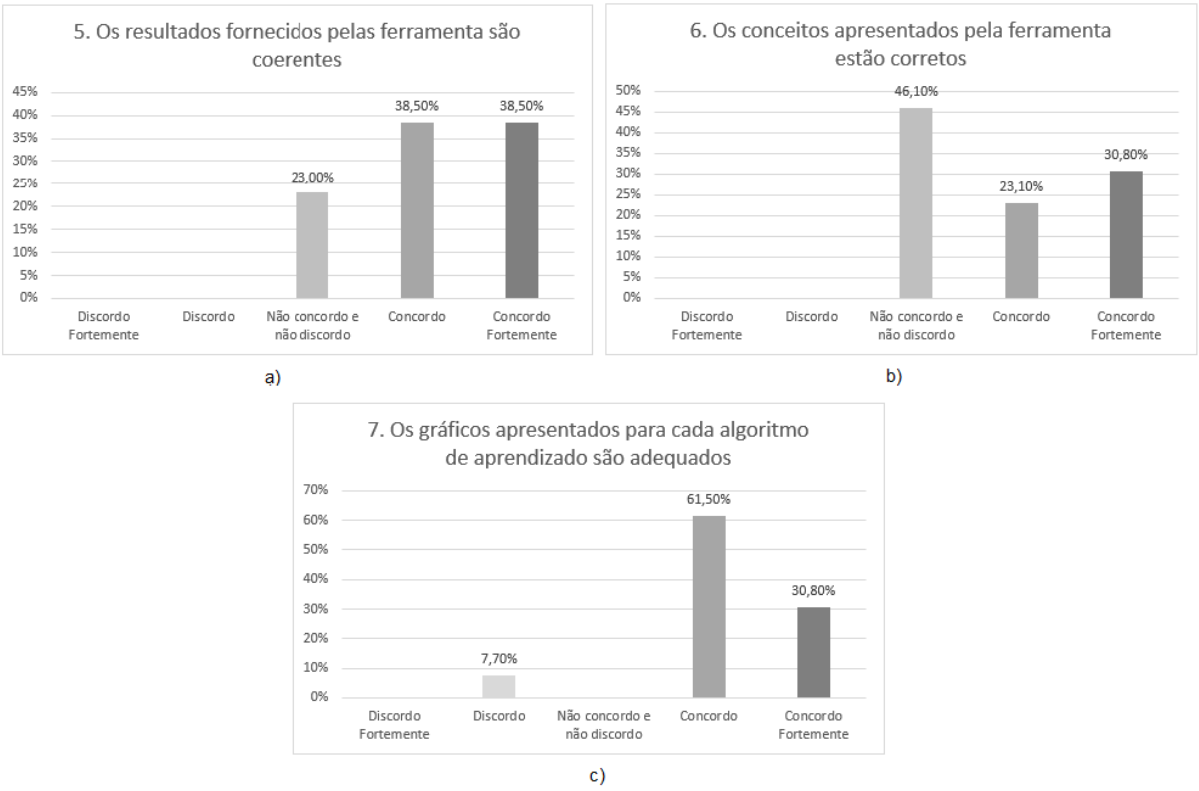
A sexta pergunta (6. Os conceitos apresentados pela ferramenta estão corretos) complementa a pergunta anterior no que se refere aos conceitos de estatística, aprendizagem de máquina, mineração de dados e análise exploratória. Nesta avaliação 53,90% foram favoráveis à afirmação que os conceitos aplicados no ambiente estavam corretos, sendo que 46,10% não souberam avaliar com exatidão, conforme a Figura 42b

A sétima pergunta (7. Os gráficos apresentados para cada algoritmo de aprendizado são adequados) valida os algoritmos de Regressão, Classificação e Agrupamento disponibilizados no LAWSMiner e os gráficos que são gerados na execução destes algoritmos. Na avaliação dos entrevistados, 92,30% concordaram que os gráficos gerados são adequados

aos algoritmos de aprendizagem apresentados e apenas 7,70% discordaram, conforme apresenta a Figura 42c.

Com os quesitos 5, 6 e 7 os avaliadores certificaram a correção dos conceitos dos gráficos e demais resultados apresentados pelo LAWSMiner, assim como os algoritmos implementados no ambiente.

Figura 42 – Avaliação dos especialistas referente a coerência dos resultados e conceitos implementados no ambiente LAWSMiner



Fonte: Elaborada pelo autor.

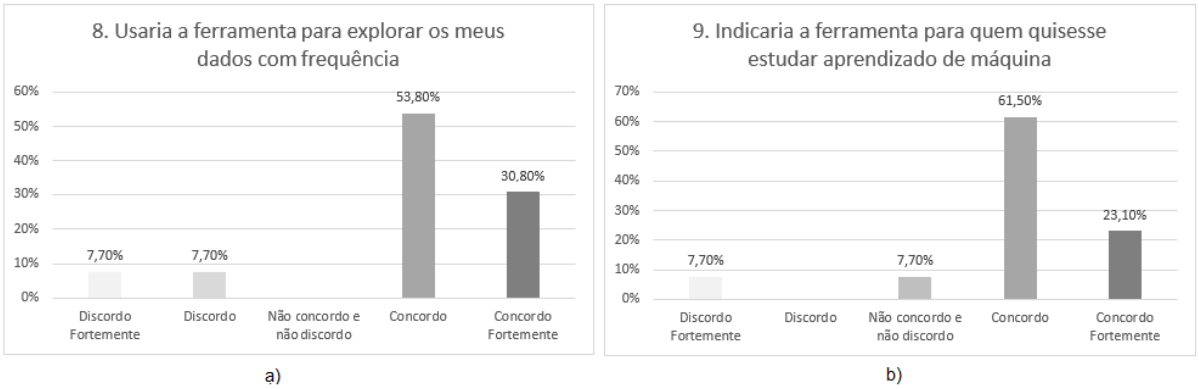
As perguntas seguintes tratam do grau de satisfação do avaliador quanto a utilização do LAWSMiner. Dois aspectos são avaliados nestes itens: A indicação do uso pessoal e frequente do ambiente e a sua indicação para o autoestudo de aprendizado de máquina. A oitava pergunta (8. Usaria a ferramenta para explorar os meus dados com frequência) trata da primeira parte. Ela mede a satisfação do avaliador no uso do ambiente assim como sua efetividade sendo uma opção viável na mineração de dados cotidianamente. A Figura 42a mostra os resultados, sendo que 84,60% concordaram com a afirmação que o LAWSMiner é um ambiente útil para mineração de dados para uso frequente, com 15,40% discordando da proposição. A ampla maioria então atestou a qualidade do LAWSMiner e seu uso para exploração de dados pessoais.

A nona e última pergunta do formulário (9. Indicaria a ferramenta para quem quisesse estudar aprendizado de máquina) tem relação ao propósito de uso do ambiente como campo de prática para o ensino de aprendizagem de máquina. Na avaliação dos

entrevistados 84,60% aprovam o uso indicariam para o estudo de aprendizagem de máquina, sendo 7,7% não souberam opinar com precisão e 7,7% discordaram.

Para o grupo de especialistas que englobam estatísticos e profissionais de ciência de dados, o LAWSMiner é um ambiente com boa usabilidade, possui funcionalidades claras e objetivas, possui coerência de conceitos, gráficos e resultados em geral e potencializa o estudo de aprendizagem de máquina.

Figura 43 – Avaliação dos especialistas referente ao uso na mineração de dados e no ensino de aprendizagem de máquina



Fonte: Elaborada pelo autor.

5 Conclusão

A análise exploratória de dados (AED) é uma estratégia de análise que busca fornecer ferramentas conceituais e computacionais para a elicitação de padrões e relacionamento entre os dados a fim de promover o desenvolvimento e o refinamento de hipóteses e descoberta de conhecimento encoberto pela grande quantidade de dados.

Uma ferramenta de AED facilita a tarefa de explorar dados pois abstrai a programação de sistema e de codificação dos algoritmos, facilitando a análise e a interpretação de gráficos, próprios da atividade de um analista.

Este trabalho propôs, implementa e valida um ambiente que automatiza o processo de mineração de dados, orientada ao usuário final, que auxilia na busca e extração de conhecimento sistematizando cada fase do processo.

O ambiente LAWSMiner foi implementado com tecnologia *open source*, voltado a web, interativo. Foi utilizada no desenvolvimento a linguagem de programação R com auxílio do pacote Shiny seguindo-se o padrão de projeto MVC.

A validação do ambiente foi realizado por dois grupos de usuários, alunos de Inteligência Artificial da UFMA e Especialistas na área de Ciência de dados que executaram um experimento controlado e responderam um questionário próprio para cada grupo.

Como resultado extraído dos questionários, o ambiente LAWSMiner se mostrou capaz de auxiliar na exploração, mineração, análise e visualização de dados, garantindo eficiência, agilidade, confiabilidade e produtividade, além de ser um elemento facilitador e potencializador no ensino e aprendizagem de Ciência de Dados, como proposto inicialmente. Mais que isso, ele apresentou bom desempenho e usabilidade, gráficos bem definidos para cada algoritmo de aprendizagem. As respostas obtidas mostraram-se coerentes com os respectivos algoritmos e os modelos gerados demostraram boa acurácia.

Outro quesito bem avaliado nos questionários foram as questões relacionadas à indicação de utilização por outros usuários, o que demonstra a satisfação gerada pelo uso do ambiente, contribuindo com os outros índices para o prosseguimento do projeto LAWSMiner além do proposto nesta pesquisa.

Como trabalhos futuros, sugere-se a ampliação dos estudos e aplicabilidade de algoritmos de aprendizagem e das demais fases do processo de descoberta do conhecimento, ampliação dos recursos de pré-processamento, como retirada e preenchimento de dados faltantes.

Referências

AWAD, E. M.; GHAZIRI, H. M. Knowledge management. *Upper Saddle River, NJ, Pearson Education International*, 2004. Citado na página 20.

BAŠKARADA, S.; KORONIOS, A. Data, information, knowledge, wisdom (DIKW): A semiotic theoretical and empirical exploration of the hierarchy and its quality dimension. *Australasian Journal of Information Systems*, v. 18, n. 1, p. 5–24, 2013. ISSN 14498618. Citado na página 19.

BEELEY, C. *Web Application Development with R using Shiny*. Birmingham B3 2PB, UK: Packt Publishing Ltd, 2013. v. 2. 110 p. ISBN 9781783284474. Citado na página 38.

BEHRENS, J. T. Principles and Procedures of Exploratory Data Analysis in: Psychological Methods Vol.2. v. 2, n. 2, p. 131–160, 1997. Citado na página 15.

BEYERER, J.; RICHTER, M.; NAGEL, M. *Pattern Recognition : Introduction, Features, Classifiers and Principles*. Berlin, Germany: De Gruyter, 2018. 283 p. ISBN 9783110537949. Citado na página 48.

BODDY D., B. A.; KENNEDY, G. Managing information systems: an organizational perspective. *Harlow, FT Prentice Hall*, 2005. Citado na página 20.

BRAGA, A. A gestão da informação. *Millenium*, v. 19, p. 1–10, 2000. ISSN 1647-662X. Disponível em: <http://repositorio.ipv.pt/handle/10400.19/903>. Citado na página 15.

BROWNLEE, J. *A Tour of the Weka Machine Learning Workbench*. 2016. Acesso em: junho. 2020. Disponível em: <https://machinelearningmastery.com/tour-weka-machine-learning-workbench/>. Citado na página 33.

CHAFFEY, D.; WOOD, S. Business information management: Improving performance using information systems. *Harlow, FT Prentice Hall*, 2005. Citado 2 vezes nas páginas 15 e 20.

CHENG, J. *Introducing Shiny: Easy web applications in R*. 2012. Disponível em: <https://www.r-bloggers.com/introducing-shiny-easy-web-applications-in-r/>. Citado na página 38.

CORRAR, L. J.; PAULO, E.; FILHO, J. M. D. Análise multivariada para os cursos de administração, ciências contábeis e economia. *Atlas*, v. 2, jan 2007. Citado na página 28.

CORTELLA, M. S. O professor e a leitura do jornal. In: SILVA, Ezequiel Theodoro da (org.). O jornal na vida do professor e no trabalho docente. *São Paulo, Campinas: Global*, 2008. Citado na página 15.

FACELI, K. et al. Inteligência artificial: Uma abordagem de aprendizado de máquina. *Livros Técnicos e Científicos Editora Ltda.*, LTC, 2015. Citado na página 22.

FALEIROS, F. et al. Uso de questionário online e divulgação virtual como estratégia de coleta de dados em estudos científicos. *Texto & Contexto-Enfermagem*, SciELO Brasil, v. 25, n. 4, 2016. Citado na página 55.

FARIA, J. C. *Qual a melhor opção de Editor, GUI, IDE para o R?* 2016. Acesso em: ago. 2019. Disponível em: <http://nbcgib.uesc.br/lec/llec/avale-es/editor-gui-ide>. Citado na página 37.

FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From Data Mining to Knowledge Discovery in Databases. *American Association for Artificial Intelligence*, p. 37–54, 1996. Citado na página 15.

FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH, P. Knowledge discovery and data mining: Towards a unifying framework. AAAI/MIT Press, v. 87, n. 1, p. 54–61, 1996. Citado na página 21.

FAYYAD, U. M. et al. (Ed.). *Advances in Knowledge Discovery and Data Mining*. Menlo Park, CA, USA: American Association for Artificial Intelligence, 1996. ISBN 0-262-56097-6. Citado 6 vezes nas páginas 15, 16, 21, 23, 36 e 44.

FOLLOW, R. *R: Unleash Machine Learning Techniques*. Birmingham B3 2PB, UK: Packt Publishing Ltd, 2016. 1123 p. ISBN 9781787127340. Citado 5 vezes nas páginas 24, 25, 30, 31 e 50.

GATTAL, A.; FAYCEL, A.; LAOUAR, M. *Automatic Parameter Tuning of K-Means Algorithm for Document Binarization*. 2018. 1-4 p. Acesso em: junho. 2020. Citado na página 30.

GéRON, A. Hands-On Machine Learning with Scikit-Learn, Keras, and Tensorflow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media, 2019. Citado 3 vezes nas páginas 29, 31 e 52.

HAYES, B. E. *Measuring Customer Satisfaction: Survey Design, Use, and Statistical Analysis Methods*. Milwaukee, WI: ASQC Quality Press, 1998. v. 2. 278 p. Citado na página 55.

HODEGHATTA, U. R.; NAYAK, U. *Business analytics using R-A practical approach*. New York, NY: Apress Media, 2016. 1–280 p. ISBN 9781484225141. Citado 3 vezes nas páginas 15, 26 e 27.

JOLLIFFE, I. T. Interpreting Principal components: Examples. *Principal Component Analysis*, 2002. ISSN 01621459. Disponível em: <http://link.springer.com/10.1007/b98835>. Citado na página 46.

KELLEHER, J. D.; NAMEE, B. M.; D'ARCY, A. *Fundamentals of machine learning for predictive data analytics : algorithms, worked examples, and case studies*. Cambridge, Massachusetts: The MIT Press, 2015. 631 p. ISBN 9780262029445. Citado na página 26.

KONRATH, A. C. et al. Desenvolvimento de Aplicativos Web Com R e Shiny : Inovações. *Abakös*, v. 6, n. 2, p. 55–71, 2018. Citado 2 vezes nas páginas 84 e 85.

LANTZ, B. *Machine Learning with R*. Birmingham B3 2PB, UK: Packt Publishing Ltd, 2013. ISBN 9781782162148. Citado 2 vezes nas páginas 29 e 52.

LAUDON, K. C.; LAUDON, J. P. Management information systems: Managing the digital firm. *Upper Saddle River, NJ, Pearson Prentice Hall*, 2004. Citado na página 20.

LINDEN, R. Técnicas de agrupamento. *Revista de Sistemas de Informação da FSMA*, v. 1, 2009. Disponível em: <http://www.fsma.edu.br/si/edicao4/FSMA_SI_2009_2_Tutorial.pdf>. Citado na página 27.

MACIEL, T. et al. Mineração de dados em triagem de risco de saúde. *Revista Brasileira de Computação Aplicada*, v. 7, n. 2, p. 26–40, maio 2015. Disponível em: <http://seer.upf.br/index.php/rbca/article/view/4651>. Citado na página 23.

MCINTIRE, G.; MARTIN, B.; WASHINGTON, L. *Python Pandas Tutorial: A Complete Introduction for Beginners*. 2019. Acesso em: jun. 2020. Disponível em: <https://www.learndatasci.com/tutorials/python-pandas-tutorial-complete-introduction-for-beginners/>. Citado na página 34.

MITCHELL, T. M. *Machine Learning*. New York, NY: McGraw-Hill, Inc., 1997. Citado na página 22.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre Aprendizado de Máquina. p. 39–56, 2003. Disponível em: <http://dcm.ffclrp.usp.br/{~}augusto/publications/2003-sistemas-inteligentes-cap4.p>. Citado na página 24.

NOGARO A.; ECCO, I. R. L. Aprendizagem e fatores motivacionais relacionados. *Revista Espaço Pedagógico*, v. 21, n. 2, 2014. Disponível em: <http://seer.upf.br/index.php/rep/article/view/4309>. Citado na página 22.

OLEKSY, A. *Data Science with R: A Step by Step Guide with Visual Illustrations & Examples*. Traverse City, Michigan: Independently Published, 2018. ISBN 9781729017456. Citado na página 30.

OZDEMIR, S. *Principles of Data Science*. Packt Publishing Ltd., 2016. 389 p. ISBN 9781785887918. Disponível em: <www.packtpub.com>. Citado 4 vezes nas páginas 15, 18, 27 e 29.

PATEL, S. *SVM (Support Vector Machine) — Theory*. 2017. <https://medium.com/machine-learning-101/chapter-2-svm-support-vector-machine-theory-f0812effc72>. Acesso: out. 2019. Citado 2 vezes nas páginas 32 e 33.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2019. Disponível em: <https://www.R-project.org/>. Citado na página 36.

RAMASUBRAMANIAN, K.; SINGH, A. Machine Learning Using R. *Apress*, v. 1st ed., 2017. Citado na página 52.

SANTHANAM, T.; PADMAVATHI, M. S. Application of K-Means and Genetic Algorithms for Dimension Reduction by Integrating SVM for Diabetes Diagnosis. *Procedia - Procedia Computer Science*, Elsevier Masson SAS, v. 47, p. 76–83, 2015. ISSN 1877-0509. Disponível em: <http://dx.doi.org/10.1016/j.procs.2015.03.185>. Citado na página 51.

SANTOS, R. Conceitos de Mineração de Dados na Web. *XV Simpósio Brasileiro de Sistemas Multimıdia e Web*, p. 40, 2009. Disponível em: <http://www.lac.inpe.br/{~}rafael.santos/Docs/WebMedia/2009/webmedia2009.p>. Citado na página 26.

SANYAL, P. *Optimizing Decision Tree Parameters using RapidMiner Studio*. 2018. Acesso em: junho. 2019. Citado na página 36.

SOUZA, E. F. *Data science — Um panorama geral*. 2018. Acesso em: jun. 2020. Disponível em: <https://medium.com/trainingcenter/data-science-um-panorama-geral-87edbbd35885>. Citado na página 34.

SUTTON, R. S.; BARTO, A. G. Reinforcement Learning: An Introduction. *A Bradford Book*, 1998. Citado na página 25.

TIRZITE, M. et al. Detection of lung cancer with electronic nose and logistic regression analysis. *Journal of Breath Research*, v. 13, 09 2018. Citado na página 28.

TOLPYGO, A. *Time-Series Analysis: Wearable Devices using DTW and kNN*. 2016. Acesso em: ago. 2019. Disponível em: <https://sflscientific.com/data-science-blog/2016/6/4/time-series-analysis-fitbit-using-dtw-and-knn>. Citado na página 29.

TSIHRINTZIS, G. A. et al. *Applications of Learning and Analytics in Intelligent Systems*. New York, NY: Springer Nature, 2019. 1–6 p. ISBN 9783030156275. Citado na página 28.

TUKEY, J. W. Exploratory Data Analysis. *Pearson*, 1977. Citado 2 vezes nas páginas 42 e 45.

VELASQUEZ, F. R.; SAUCEDA, M. Enfoques sobre el aprendizaje humano. 2001. Citado na página 22.

Wu, X. et al. Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, v. 26, n. 1, p. 97–107, Jan 2014. Citado na página 18.

Apêndices

APÊNDICE A – Atividade de Avaliação

Avaliação da ferramenta LawsMiner.

O *dataset_jogadores* contém dados de 150 jogadores de futebol, que estão divididos entre atacantes e defensores.

Para cada jogador, registrou-se: idade, altura, técnica, passe, chute, força e velocidade. As características (exceto a idade e altura) são valores discretos entre 1 e 100, onde valores maiores são melhores.

A classe (atacante ou defensor) dos primeiros 120 atletas é conhecida, e a partir destas informações, deseja-se descobrir a classe de outros 30 atletas.

Dentre os algoritmos apresentado na ferramenta, selecione e classificar os 30 atletas que não possuem classe atribuída:

- K-Means
- K-Nearest Neighbor (KNN)
- Regressão Linear
- Árvore de Decisão
- Naïve Bayes
- Suport Vector Machine (SVM)

Utilizem os 120 casos conhecidos para criar um modelo de classificação, junto ao classificador selecionado. Realizem as etapas de treino e teste supervisionado. Depois, classifiquem as instâncias desconhecidas.

Por fim, preencha o formulário de avaliação da ferramenta no endereço: https://docs.google.com/forms/d/1nQH_UJ_rR61r2qfKbW6jvlBUh12sWylqktpSQUJtTqI/prefill

Link Ferramenta: lawsminer.ml

APÊNDICE B – Dataset Jogadores

Dataset Jogadores

Id	Idade	Altura	Técnica	Passe	Chute	Força	Velocidade	Drible	Classe
1	17	177	72	65	72	60	84	81	Atacante
2	18	188	63	65	55	70	72	60	Defensor
3	18	190	63	65	67	70	72	66	Atacante
4	19	165	65	62	71	62	70	67	Atacante
5	19	174	67	66	69	64	76	74	Atacante
6	19	184	64	68	47	73	71	63	Defensor
7	19	184	64	62	70	75	73	65	Atacante
8	19	186	68	66	52	76	72	63	Defensor
9	20	170	68	65	64	62	79	75	Atacante
10	20	175	67	66	69	62	82	74	Atacante
11	20	180	66	68	70	70	66	68	Atacante
12	20	184	73	65	79	79	77	71	Atacante
13	20	187	64	69	50	72	62	62	Defensor
14	20	188	62	64	47	70	72	64	Defensor
15	20	188	65	64	45	79	72	58	Defensor
16	20	188	72	69	45	82	74	63	Defensor
17	20	190	64	69	58	70	62	61	Defensor
18	20	191	63	63	45	74	73	60	Defensor
19	21	170	64	65	70	66	70	67	Atacante
20	21	172	67	67	66	68	72	71	Atacante
21	21	174	72	70	74	65	81	76	Atacante
22	21	175	67	68	49	70	69	65	Defensor
23	21	179	71	65	76	71	74	71	Atacante
24	21	180	65	65	46	73	76	60	Defensor
25	21	182	72	68	70	71	86	76	Atacante
26	21	183	66	64	58	75	72	62	Defensor
27	21	184	70	67	70	75	80	72	Atacante
28	21	188	67	67	51	75	66	62	Defensor
29	21	192	62	74	64	71	68	70	Defensor
30	22	166	72	70	60	61	76	81	Atacante
31	22	177	69	65	71	77	77	80	Atacante
32	22	180	64	61	68	70	79	70	Atacante
33	22	181	65	67	62	67	75	77	Defensor
34	22	181	74	68	56	80	78	62	Defensor
35	22	183	67	62	73	81	74	68	Atacante
36	22	184	71	66	46	79	74	71	Defensor
37	22	185	72	65	61	74	70	60	Defensor
38	22	187	69	69	57	80	74	58	Defensor
39	22	188	71	65	58	77	76	62	Defensor
40	22	190	73	66	58	81	75	63	Defensor
41	22	195	63	61	46	76	71	63	Defensor
42	23	170	70	66	72	62	84	82	Atacante
43	23	174	75	65	78	77	84	74	Atacante
44	23	176	78	80	72	70	74	81	Atacante

Anexos

ANEXO A – Questionário de Avaliação do Ambiente LAWSMiner - Alunos

Questionário de Avaliação da Ferramenta LAWSMiner - Alunos

*Obrigatório

Termo de Consentimento

Você está sendo convidado(a) para responder às perguntas deste questionário de forma voluntária.Ao responder você autoriza que as respostas sejam utilizadas em uma pesquisa acadêmica relacionada a aprendizagem de máquinas. *

- ◯ Concordo
- ◯ Não Concordo

Possuo conhecimentos em Programação de Computadores*

- ◯ Discordo Fortemente
- ◯ Discordo
- ◯ Não concordo e não discordo
- ◯ Concordo
- ◯ Concordo Fortemente

Possuo conhecimentos em Aprendizagem de máquina*

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Conheço os algoritmos de aprendizado de máquina utilizados na ferramenta*

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Já implementei (em Python, R, entre outras) os algoritmos apresentados na ferramenta *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

A ferramenta apresenta um bom desempenho na execução das tarefas*

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Consegui executar com facilidade a atividade proposta*

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

As respostas encontradas estavam coerentes com o propósito dos algoritmos*

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Os gráficos auxiliam na análise dos resultados de cada algoritmo*

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

A abordagem empregada na ferramenta facilita o entendimento dos algoritmos*

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

A ferramenta é útil para o aprendizado dos algoritmos de aprendizagem de máquina *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Usaria a ferramenta para explorar os meus dados com frequência *

- ◯ Discordo Fortemente
- ◯ Discordo
- ◯ Não concordo e não discordo
- ◯ Concordo
- ◯ Concordo Fortemente

Indicaria a ferramenta para quem quisesse estudar aprendizado de máquina *

- ◯ Discordo Fortemente
- ◯ Discordo
- ◯ Não concordo e não discordo
- ◯ Concordo
- ◯ Concordo Fortemente

ANEXO B – Questionário de Avaliação do Ambiente LAWSMiner - Especialistas

Questionário de Avaliação da Ferramenta LAWSMiner - Especialistas

*Obrigatório

Termo de Consentimento

Você está sendo convidado(a) para responder as perguntas deste questionário de forma voluntária.Ao responder você autoriza que as respostas sejam utilizadas em uma pesquisa acadêmica relacionada a aprendizagem de máquinas. *

- ○ Concordo
- ○ Não Concordo

Formulário de avaliação da Ferramenta LAWS Miner - Especialistas

A ferramenta tem uma boa usabilidade*

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

As funcionalidades da ferramenta são claras e objetivas *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo

A ferramenta se mostrou útil para análise de dados *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo

A ferramenta agiliza a análise de dados *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Os resultados fornecidos pela ferramenta são coerentes * *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Os conceitos apresentados pela ferramenta estão corretos *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Os gráficos apresentados para cada algoritmo de aprendizado são adequados *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Usaria a ferramenta para explorar os meus dados com frequência *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

Indicaria a ferramenta para quem quisesse estudar aprendizado de máquina *

- ○ Discordo Fortemente
- ○ Discordo
- ○ Não concordo e não discordo
- ○ Concordo
- ○ Concordo Fortemente

ANEXO C – Estrutura Shiny: *Server* e *Ui*

Tabela 4 – Shiny - Estrutura do *UI*

Função	**Finalidade**
library(shiny)	Carregar o pacote Shiny.
shinyUI(fluidPage)	Criar uma interface com o usuário.
titlePanel()	Criar um painel contendo um título do aplicativo.
sidebarLayout	Criar um layout com uma barra lateral e área principal. A barra lateral é exibida com uma cor de fundo distinta e geralmente contém controles de entrada. A área principal ocupa 2/3 da largura horizontal e geralmente contém saídas.
sidebarPanel()	Criar um painel com barra lateral, que contenha controles de entrada que, por sua vez, possam ser passados para SidebarLayout
mainPanel()	Criar um painel principal contendo elementos de saída que, por sua vez, pode ser passado para sidebarLayout.

Fonte: Konrath et al. (2018)

Tabela 5 – Shiny - Estrutura do *Server*

Função	**Finalidade**
library(shiny)	Carregar o pacote Shiny.
shinyServer()	Definir a lógica do servidor do aplicativo Shiny. Isso geralmente envolve a criação de funções que mapeiam entradas de usuários para vários tipos de saída..
mainPanel()	Criar um painel principal contendo elementos de saída que, por sua vez, pode ser passado para sidebarLayout.

Fonte: Konrath et al. (2018)

ANEXO D – Dispositivos de Saída Shiny: *Render* e *Output*

Tabela 6 – Shiny - Dispositivos de saída - Render()

Função	**Finalidade**
renderDataTable	Criar uma versão reativa de uma dada função que retorna um dataframe (ou matriz), que será renderizada com a biblioteca DataTables.
renderImage	Renderizar uma imagem reativa que é adequada para atribuir a um slot de saída.
renderPlot	Renderizar um gráfico reativo que é adequado para atribuir a um slot de saída.
renderPrint	Criar uma versão reativa da função dada que captura qualquer saída impressa.
renderTable	Criar uma tabela reativa que é adequada para atribuir a um slot de saída.
renderText	Criar uma versão reativa de uma dada função, que também usa o cat para transformar seu resultado em um vetor de caracteres de um único elemento.
renderUI	Criar uma versão reativa de uma função que gera HTML usando a biblioteca Shiny UI.

Fonte: Konrath et al. (2018)

Tabela 7 – Shiny - Dispositivos de saída - Output()

Função	**Finalidade**
dataTableOutput	Renderizar uma renderTable ou renderDataTable dentro de uma página do aplicativo. A renderTable usa uma tabela HTML padrão, enquanto a renderDataTable usa a biblioteca JavaScript DataTables para criar uma tabela interativa com mais recursos.
imageOutput	Renderizar um renderPlot ou renderImage dentro de uma página do aplicativo.
plotOutput	Renderizar um renderPlot ou renderImage dentro de uma página do aplicativo.
verbatimTextOutput	Renderizar uma variável de saída reativa como texto dentro de uma página de aplicativo.
tableOutput	Renderizar uma renderTable ou renderDataTable dentro de uma página do aplicativo. O renderTable usa uma tabela HTML padrão, enquanto o renderDataTable usa a biblioteca JavaScript DataTables para criar uma tabela interativa com mais recursos.
textOutput	Renderizar uma variável de saída reativa como texto dentro de uma página de aplicativo. O texto será incluído dentro de uma tag HTML div por padrão.
UiOutput & htmlOutput	Renderizar uma variável de saída reativa como HTML dentro de uma página de aplicativo. O texto será incluído em uma tag HTML.

Fonte: Konrath et al. (2018)