UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA E TECNOLOGIA PARA RECURSOS AMAZÔNICOS

SELEÇÃO DE ATRIBUTOS RELEVANTES: aplicando técnicas na base de dados do Herbário Virtual da Flora e dos Fungos.

ADRIANO HONORATO DE SOUZA

ITACOATIARA
2017

UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA E TECNOLOGIA PARA RECURSOS AMAZÔNICOS

ADRIANO HONORATO DE SOUZA

SELEÇÃO DE ATRIBUTOS RELEVANTES: aplicando técnicas na base de dados do Herbário Virtual da Flora e dos Fungos.

Dissertação apresentada ao Programa de Pós-Graduação em Ciência e Tecnologia para Recursos Amazônicos do Instituto de Ciências Exatas e Tecnologia da Universidade Federal do Amazonas, como requisito para obtenção do título de Mestre em Ciência e Tecnologia para Recursos Amazônicos, área de concentração Estudos Teóricos e Computacionais.

Orientador: Dr. Jorge Yoshio Kanda

ITACOATIARA- AM
2017

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

Souza, Adriano Honorato de
S729s SELEÇÃO DE ATRIBUTOS RELEVANTES: aplicando técnicas na base de dados do Herbário Virtual da Flora e dos Fungos / Adriano Honorato de Souza. 2017
82 f.: il. color; 31 cm.

Orientador: Jorge Yoshio Kanda
Dissertação (Mestrado em Ciência e Tecnologia para Recursos Amazônicos) - Universidade Federal do Amazonas.

1. Aprendizado de Máquina. 2. Filtro. 3. Wrapper. 4. Embutido. I. Kanda, Jorge Yoshio II. Universidade Federal do Amazonas III. Título

ADRIANO HONORATO DE SOUZA

Seleção de atributos relevantes: aplicando técnicas na base de dados do Herbário Virtual da Flora e dos Fungos.

Dissertação apresentada ao Programa de Pós-Graduação em Ciência e Tecnologia para Recursos Amazônicos da Universidade Federal do Amazonas, como parte do requisito para obtenção do título de Mestre em Ciência e Tecnologia para Recursos Amazônicos, área de concentração Desenvolvimento Científico e Tecnológico em Recursos Amazônicos.

Aprovado em 29 de setembro de 2017.

BANCA EXAMINADORA

Dr. Jorge Yoshio Kanda, Presidente
Universidade Federal do Amazonas

Dr. José Pinheiro de Queiroz Neto
Instituto Federal do Amazonas

Dr. Fernando Ruy
Instituto Federal do Amazonas

DEDICATÓRIA

Dedico este trabalho primeiramente à Deus, pois o que seria de mim sem a fé que tenho n'Ele, ao meu orientador, aos meus pais, aos meus irmãos, esposa, filhos, familiares e amigos que de muitas formas me incentivaram e ajudaram para que fosse possível a concretização deste trabalho.

AGRADECIMENTOS

Em primeiro lugar à Deus pela força e determinação para superar todas as tribulações até a conclusão deste trabalho.

Ao meu pai, José Marques de Souza (*in memoriam*), mesmo não estando mais presente entre nós agradeço por ter mostrado o exemplo a ser seguido.

À minha mãe, Maria das Dores Honorato de Souza, agradeço a esta guerreira pelo seu imenso amor que tens não só por mim, mas, por todos os seus filhos e filhas, agradeço ainda toda a sua dedicação e incentivo em todos os meus projetos de vida, e a confiança que deposita em mim.

Ao meu orientador Professor Dr. Jorge Yoshio Kanda por todo seu ensinamento, compreensão e ter sempre uma palavra de ânimo para continuar e não desistir.

Ao Instituto Nacional de Ciência e Tecnologia Herbário Virtual da Flora e dos Fungos (INCT-HVFF) que conta com apoio financeiro do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

À minha esposa Aniklay de Oliveira Lamarão agradeço por compreender a importância desta conquista e aceitar a minha ausência quando foi preciso.

Aos meus filhos Alícia Luz Lamarão Honorato e Adam Davi Lamarão Honorato, onde busquei forças para continuar.

Aos meus amigos, pela força e companheirismo, no qual destaco aqui o companheiro de tantas batalhas deste mestrado Evren Ney e aos demais colegas que não deixaram desanimar nesta caminhada.

Por fim, agradeço também a todos, sem exceção, que torceram por mim, seja em orações, pensamento positivo ou simplesmente com uma palavra amiga dada na hora certa.

Epígrafe

"O sofrimento é passageiro.

O fracasso é para sempre! "

(Bernardo Fonseca)

RESUMO

Os herbários virtuais têm como objetivo disseminar informações científicas e contribuir para a conservação e uso sustentável dos recursos biológicos brasileiros. Atualmente integra 120 herbários nacionais e 25 herbários do exterior, juntos disponibilizam mais de 5,4 milhões de registros e mais de um milhão de imagens, além de várias ferramentas de livre acesso, abrindo espaço para a aplicação de técnicas de Aprendizagem de Máquina, entre elas os classificadores. No processo de Aprendizagem de Máquina a Seleção de Atributos faz parte do pré-processamento de dados e que pode corresponder a 80% da fase da mineração de dados, para isso se faz necessário um estudo sobre das abordagens utilizadas para fazer a seleção de um subconjunto de atributos que melhor generalize a base para ser induzido ao modelo de aprendizado de máquina. O objetivo deste trabalho é aplicar os processos de seleção de atributos com as seguintes abordagens filtro, *wrapper* e embutido, na base de dados do Instituto Nacional de Ciência e Tecnologia – Herbário Virtual da Flora e dos Fungos, esta base contém 87.732 registros e 51 atributos, sendo 119 coleções e sub-coleções, 86.967 registros *online*, 80.513 registros georreferenciados, 12.073 espécies aceitas distintas. A primeira fase dos processos de aprendizado de máquina é o pré-processamento, que analisará a base de dados e resultará em uma base mais genérica e pronta para aplicação dos modelos preditivos de classificação, após o filtro do subconjunto de atributos mais relevantes aplicam-se os algoritmos de Aprendizagem de Máquina, que nesta pesquisa foi: Árvore de Decisão, Rede Neural Artificial e Regressão Logística. A avalição dos modelos será através da matriz de confusão utilizando a acurácia e a análise da área sobre a curva ROC. Dentre os modelos estudados o de Regressão Logística obteve o desempenho de classificação de acurácia de 77,25%, com a abordagem filtro e 76,25% com a *wrapper*.

Palavras-Chaves: Aprendizado de Máquina; Filtro; *Wrapper;* Embutido.

ABSTRACT

Virtual herbariums aim to disseminate scientific information and contribute to the conservation and sustainable use of Brazilian biological resources. It currently includes 120 national herbaria and 25 herbariums from abroad, together provide more than 5,4 million records and more than one million images, in addition to several free access tools, opening space for the application of Machine Learning techniques, among them classifiers. In the Machine Learning process, Attribute Selection is part of the pre-processing of data and can correspond to 80% of the data mining phase, for this it is necessary to study the approaches used to make the selection of a subset of attributes that better generalize the basis to be induced to the model of machine learning. The objective of this work is to apply the attributes selection processes with the following filter, wrapper and embedded approaches in the National Institute of Science and Technology (NIST) - Virtual Herbarium of Flora and Fungi, this base contains 87,732 records and 51 features, with 119 collections and sub-collections, 86,967 online records, 80,513 georeferenced records, 12,073 different accepted species. The first phase of machine learning processes is the pre-processing, which will analyze the database and will result in a more general and ready basis for the application of the predictive models of classification, after the filter of the most relevant subset of attributes, the Machine Learning algorithms are applied, which in this research was: Decision Tree, Network Neural Artificial and Logistic Regression. The evaluation of the models will be through the confusion matrix using the accuracy and the analysis of the area on the ROC curve. Among the models studied, the Logistic Regression was the one that obtained the performance with a total accuracy of 77.25%, with the filter approach and 76.25% with the wrapper.

Key words: Machine Learning; Filter; Wrapper; Embedded.

LISTA DE FIGURAS

LISTAS DE QUADRO

LISTA DE SIMBOLOS

α – Alfa

β – Beta

γ – Gama

Σ – Sigma

Φ – Fi

SUMÁRIO

1 INTRODUÇÃO

Conceitos e técnicas de Inteligência Artificial (IA) têm sido incorporados cada vez mais às soluções de problemas reais de diversas áreas através do Aprendizado de Máquina (AM), tais sistemas dependem bastante do conhecimento, que são adquiridos, representados e manipulados. Para que um comportamento inteligente possa ser exibido por uma máquina, os processos de aquisição, representação e manipulação devem estar relacionados (MITCHELL, 1997).

Para ter um sistema de AM de qualidade as hipóteses induzidas dependem principalmente da quantidade e da qualidade dos atributos e exemplos utilizados no treinamento. Quando experimentos são obtidos a partir de grandes bases de dados o resultado são hipóteses de baixa precisão, pois existe na maioria das bases de dados muitos atributos ditos irrelevantes e a maioria dos sistemas de AM conhecidos não estão preparados para trabalhar com uma quantidade com muitos atributos (DIETTERICH, 2000), daí a necessidade da fase de pré-processamento ser de suma importância para a preparação dos dados antes de serem induzidos aos algoritmos de AM.

Um sistema de AM é, então, um programa de computador que toma decisões baseadas em experiências acumuladas contidas em exemplos (ou casos) previamente resolvidos com sucesso (MITCHELL, 1997). Na hierarquia do aprendizado proposta por Monard e Baranauskas (2003), pode ser dividido em duas categorias: o aprendizado supervisionado e o aprendizado não-supervisionado, sendo o aprendizado supervisionado dividido em Classificação e Regressão. O aprendizado supervisionado compreende os algoritmos de indução que realizam inferências a partir de dados rotulados. Quando os rótulos são discretos o problema é denominado de classificação e quando são contínuos de regressão. Já o aprendizado não-supervisionado, são disponibilizados para o algoritmo apenas os

exemplos de entrada, não existindo informação sobre a saída esperada. Uma das principais subáreas de aprendizado não-supervisionado é o agrupamento de dados. Algoritmos de agrupamento de dados têm como objetivo encontrar padrões entre os exemplos por meio da formação de agrupamentos, este tipo de aprendizado pode ser dividido em Sumarização, Associação e Agrupamento (DUDA, HART e STORK, 2000), como mostra a Figura 1 sobre a hierarquia de aprendizado.

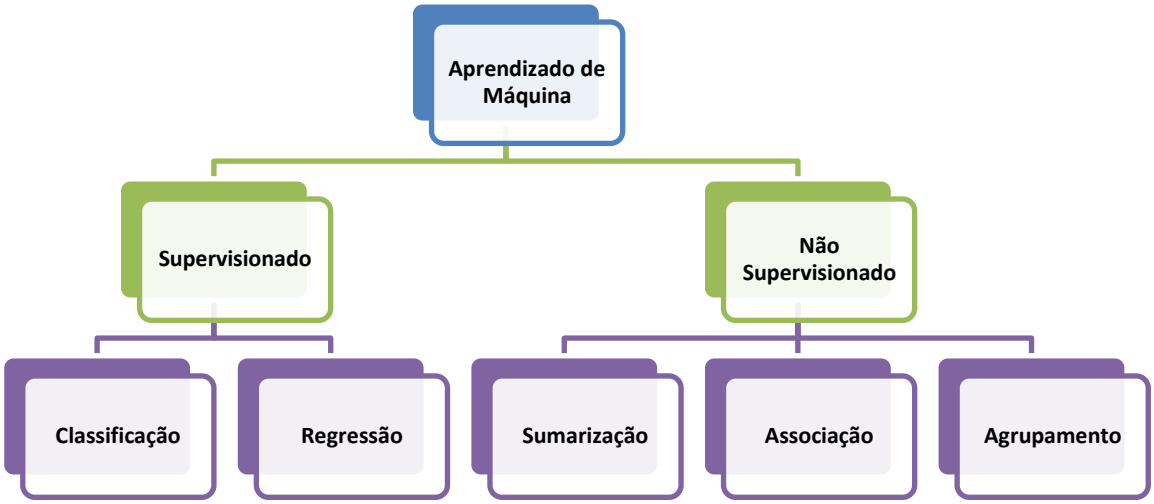


Figura 1. Hierarquia do Aprendizado de Máquina

O escopo deste trabalho será categorizado com o aprendizado supervisionado e restrito aos problemas de classificação. Assim, a saída gerada pelo sistema de AM será um classificador, o qual mapeará instâncias não classificadas para rótulos (classes) utilizando alguma estrutura interna armazenada. Um sistema de AM supervisionado é treinado a partir de um conjunto de exemplos descritos por características, chamadas de atributos preditivos, e pelo valor da classe previamente conhecido que é denominado atributo alvo.

A base de dados utilizada neste trabalho é do Instituto Nacional de Ciência e Tecnologia (INCT) – Herbário Virtual da Flora e dos Fungos, está disponível na rede *speciesLink* podendo ser acessado através do endereço eletrônico http://inct.splink.org.br. O *speciesLink* é um sistema distribuído de informações que integra em tempo real, dados

primários de coleções científicas que tem por princípio promover o acesso livre e aberto aos dados, informações e ferramentas disponíveis a qualquer indivíduo ou grupo. Só no ano de 2016, foram recuperados 591 milhões de registros pelos usuários da rede *speciesLink* (SPLINK, 2016).

Os métodos de AM para classificação podem ser divididos em duas fases, assim como ilustra a Figura 2: na primeira fase os exemplos rotulados (exemplos de treinamento) são fornecidos ao sistema de AM, que geralmente é um algoritmo de indução (ou simplesmente indutor) capaz de extrair conhecimento desses exemplos rotulados e gerar um classificador (modelo) representado em uma estrutura interna. Na segunda fase o classificador gerado pelo sistema de AM é utilizado para rotular novos exemplos (exemplos de teste) não vistos durante o treinamento, após o teste é feito a avaliação da precisão do modelo através de métodos de avaliação.

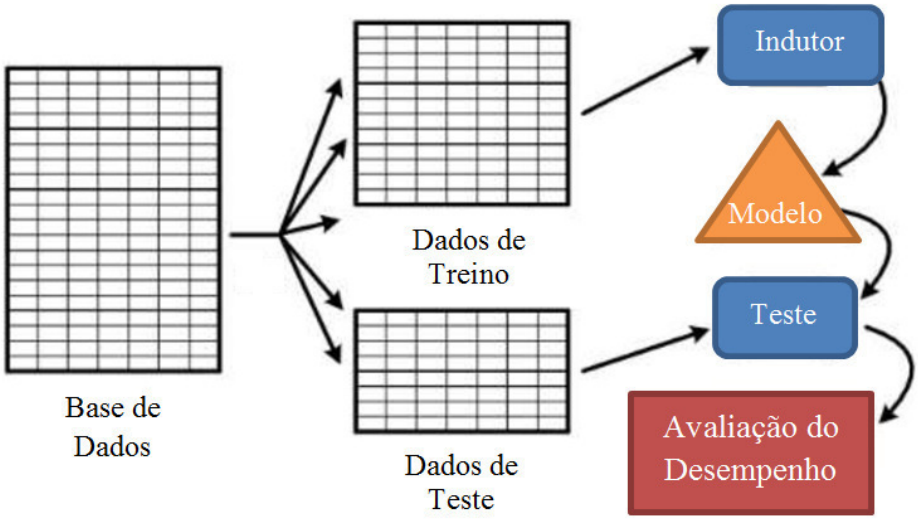


Figura 2. Fluxograma - Fases do método de classificação

Então, um sistema de AM para classificação é um programa capaz de extrair o conhecimento implícito nos exemplos e gerar um classificador cuja estrutura interna relaciona os valores dos atributos à classe (WEISS e KULIKOWSKI, 1990). Sendo seu objetivo obter uma função que seja capaz de predizer a saída para qualquer entrada válida, após ter vista um

número suficiente de exemplos de treinamento. Para atingir este objetivo, o algoritmo deve ter capacidade de generalização para que possa prever, de maneira aceitável, a saída para dados ainda não vistos (BREVE e ZHAO, 2012).

O conhecimento sobre o domínio pode ser usado para escolher os dados ou para fornecer alguma informação previamente conhecida como entrada ao indutor. Após induzido, o classificador é geralmente avaliado e o processo de classificação pode ser repetido, se necessário, por exemplo, adicionando outros atributos, exemplos ou mesmo ajustando alguns parâmetros no processo de indução (REZENDE, 2003).

A precisão com que o classificador prediz a classe dos exemplos está relacionada à sua representação interna utilizando os atributos, se os atributos não são capazes de representar o conhecimento implícito nos exemplos, a precisão do classificador pode ser baixa, ou seja, novos exemplos submetidos ao classificador terão grande chance da classe ser predita incorretamente. Portanto, quanto mais significativos forem os atributos utilizados para descrever os exemplos, mais confiável será a classificação. Em outras palavras, se os atributos apresentam as propriedades essenciais dos exemplos, cabe ao sistema de AM representá-los em uma estrutura capaz de generalizar o conhecimento implicitamente concebido nos exemplos. Essa estrutura é uma síntese de todos os exemplos e o sucesso da predição utilizando essa estrutura está diretamente relacionado ao poder de representação dos atributos.

Outro fator importante que influencia os algoritmos de AM é o número de atributos utilizados para representar os exemplos (KIRA e RENDELL, 1992). Geralmente, os algoritmos de AM computacionais viáveis não trabalham bem na presença de grandes quantidades de atributos, contudo, determinar quais atributos são relevantes é uma tarefa complexa.

Um problema central da Seleção de Atributos (SA) é como determinar quais atributos devem ser selecionados ou quais atributos devem ser ignorados? Intuitivamente,

seria desejável que um indutor utilizasse apenas os atributos relevantes para o aprendizado do conceito, porém, a inexistência de uma análise matemática que permita a avaliação do desempenho de um método de aprendizado sobre um conjunto de dados faz com que estudos experimentais sejam muito importantes na área de AM. Assim, também para a tarefa de SA é importante que estudos empíricos sejam realizados sobre os conjuntos de dados de interesse, a fim de avaliar que métodos de SA são mais apropriados, ou apresentam o melhor desempenho, para indutores e conjuntos de dados específicos.

1.1 Objetivos

1.1.1 Objetivo geral

Obter os atributos mais relevantes da base de dados do INCT – Herbário Virtual da Flora e dos Fungos, por meio de processos de seleção de subconjuntos de atributos utilizando técnicas de Aprendizado de Máquina.

1.1.2 Objetivos específicos:

- Analisar os processos de seleção de subconjunto de atributos utilizando abordagens filtro, *wrapper* e embutido;
- Induzir modelos de classificação descritos pelos subconjuntos de atributos selecionados;
- Avaliar o desempenho dos modelos induzidos de AM.

1.2 Organização do trabalho

Este trabalho está organizado da seguinte forma: a Sessão 2 explora os conceitos, as características e as etapas de Aprendizagem de Máquina. A Sessão 3 descreve os conceitos e abordagens de Seleção de Atributos. A Sessão 4 conceitua os modelos preditivos que serão aplicados neste trabalho. Na Sessão 5 apresenta a Base de Dados utilizada neste trabalho de Mestrado. A Sessão 6 contém a Metodologia adotada para aplicar os modelos de AM em uma

base de dados, desde a fase de importação da base, passando pela fase de pré-processamento e limpeza dos dados depois a fase de treinamento do classificador, o processo de classificação dos dados e a avaliação dos classificadores. Na Sessão 7 os resultados obtidos com os atributos selecionados. E na Sessão 8 a conclusão desta dissertação e sugestões de trabalhos futuros.

2 APRENDIZADO DE MÁQUINA

A área de AM estuda como desenvolver programas de computador que melhoram seu desempenho em alguma tarefa por meio da experiência sem que sejam programados para o mesmo (MITCHELL, 1997). O mesmo autor ressalta que aprender, nesse contexto, pode ser definido como: - Para situações em que o desempenho em alguma atividade possa ser mensurado: diz-se que em sistema computacional aprende a partir da experiência E, em relação a uma classe de tarefas T, com medida de desempenho P, se seu desempenho nas tarefas T, medida por P, melhora com a experiência E (MITCHELL, 1997), ou seja, com técnicas de AM o computador a partir de um conjunto de treinamentos poderá desenvolver uma função matemática que possa classificar de forma mais correta novas tarefas com o conhecimento adquirido no conjunto que serviu de treinamento.

Em linhas gerais, AM pode ser caracterizada por uma série de práticas voltadas para a solução de problemas para as quais geralmente não se conhece, *a priori*, uma solução ou modelagem capaz de resolvê-los, o que se conhece é um conjunto finito de fatos – também conhecidos como casos ou exemplos, que descrevem objetos, processos, situações ou ambientes e o objetivo é encontrar alguma solução a partir desses fatos. Essas práticas incluem, entre outros, generalização a partir de um conjunto de casos cujas classes são conhecidas (DUDA, HART e STORK, 2000).

A área de AM tem uma grande ligação com a Estatística, uma vez que ambas as áreas estudam a análise de dados. Entretanto, diferentemente da estatística, que tem como foco principal modelos teóricos bem definidos e ajustamento de parâmetros a esses modelos, AM tem um foco mais algorítmico, utilizando representações de modelos mais flexíveis e heurísticas para a realização de busca (KAUFMAN e MICHALSKI, 2005). Como por exemplo, uma análise estatística pode determinar distribuições, covariâncias e correlações entre os atributos que descrevem os fatos, mas não é capaz de caracterizar essas dependências

em um nível abstrato e conceitual como os humanos fazem, nem prover uma explicação causal do porque essas dependências existem. Enquanto uma análise estatística dos dados pode determinar as tendências centrais e variâncias de determinados fatores, ela não pode produzir uma descrição qualitativa das regularidades, nem tão pouco determinar as dependências em fatores não providos explicitamente com os dados (KAUFMAN e MICHALSKI, 2005).

2.1 Características Gerais de Aprendizagem de Máquina

Um sistema de AM é um programa de computador que toma decisões baseadas em experiências acumuladas por meio da solução de problemas anteriores (MITCHELL, 1997). Os sistemas de AM possuem características particulares e comuns que possibilitam certa classificação quanto ao modo, paradigma de aprendizagem e tarefa de aprendizado utilizado nesses sistemas, como mostrados na Tabela 1.

Modos de Aprendizado	**Paradigmas de Aprendizado**	**Tarefas de Aprendizado**
Supervisionado	Simbólico	Classificação
Não Supervisionado	Estatístico	
Semi Supervisionado	Baseado em Protótipos	Ordenação
Por Reforço	Conexionista Genético	Regressão

Quadro 1. Características gerais de sistemas de aprendizagem de máquina

2.1.1 Modos de Aprendizado

Para alguns sistemas de AM é necessário predizer se certa ação irá fornecer uma certa saída, com isso, é possível classificar os sistemas de AM nos seguintes modos, segundo (RUSSELL e NORVIG, 2003):

a. **Aprendizagem Supervisionada,** no qual dado um conjunto de observações ou exemplos rotulados, isto é, conjunto de observações em que a classe, denominada também atributo alvo, de cada exemplo é conhecida, o objetivo é encontrar uma hipótese capaz de classificar novas observações entre as classes já existentes.

b. **Aprendizagem Não Supervisionada,** no qual dado um conjunto de observações ou exemplos não rotulados, o objetivo é tentar estabelecer a existência de grupos ou similaridades nesses exemplos.

c. **Aprendizagem Semi-Supervisionada,** no qual dado um pequeno conjunto de observações ou exemplos rotulados e um conjunto de observações ou exemplos não rotulados, o objetivo é utilizar ambos os conjuntos para encontrar uma hipótese capaz de classificar novas observações entre as classes já existentes. Este tipo de aprendizagem é um meio termo entre aprendizagem supervisionada e não supervisionada.

d. **Aprendizagem por Reforço,** no qual o agente aprendiz interage com o meio ambiente que o cerca e aprende uma política ótima de ação por experimentação direta com o meio. Dependendo de suas ações, o aprendiz é recompensado ou penalizado. O objetivo do aprendiz é desenvolver uma política ótima que maximize a quantidade de recompensa recebida ao longo da sua execução.

2.1.2 Paradigmas de aprendizado

Muito se tem discutido com relação aos paradigmas de AM e de forma sucinta são abordados alguns deles tais como o paradigma simbólico, estatístico, baseado em protótipo e genético.

a. Paradigma Simbólico, este tipo de sistema busca aprender construindo representações simbólicas de um conceito por meio da análise de exemplos e contraexemplos desse conceito. As representações simbólicas estão tipicamente representadas na forma de alguma expressão lógica, Árvore de Decisão, regras de produção ou rede semântica.

Atualmente, entre as representações simbólicas mais estudadas estão as árvores de decisão e regras de decisão. Métodos de indução de árvores de decisão a partir de dados empíricos, conhecidos como particionamento recursivo, foram estudados por pesquisadores da área de Inteligência Artificial e Estatística. Os sistemas ID3 (QUINLAN, 1986) e C4 (QUILAN, 1987) para indução de árvores de decisão tiveram uma importante contribuição sobre a pesquisa de IA. O sistema de classificação de árvores de regressão CART (BREIMAN e FRIEDMAN, 1985) foi desenvolvido por estatísticos, durante praticamente o mesmo período que o ID3, no final dos anos 70.

Os trabalhos com indução de regras de decisão surgiram com a família de algoritmos AQ (KAUFMAN e MICHALSKI, 2005), e precederam os algoritmos como o CN2 (CLARK e NIBLETT, 1989; CLARK e BOSWELL, 1991). Outra abordagem surgiu da simples tradução das árvores de decisão para regras, com um posterior refinamento (QUILAN, 1987), resultando no algoritmo C4.5rules (QUILAN, 1993).

b. Paradigma Estatístico, pesquisadores em estatística tem criado diversos métodos de classificação, muitos deles semelhantes aos métodos empregados pela comunidade científica em aprendizagem de máquina, como exemplo, temos o método CART (BREIMAN

e FRIEDMAN, 1985), mencionado anteriormente, é um sistema muito conhecido para construir árvores de decisão, desenvolvido por estatísticos. Como regra geral, técnicas estatísticas tendem a focar tarefas em que todos os atributos têm valores contínuos ou ordinais. Muitos desses métodos também são paramétricos, assumindo algum modelo pré-estabelecido, e então ajustando valores apropriados para os parâmetros do modelo a partir dos dados, como por exemplo, um modelo de combinação linear dos valores dos atributos, e então procura uma combinação linear particular que fornece a melhor aproximação sobre o conjunto de dados. Os modelos estatísticos com frequência assumem que os valores de atributos estão normalmente distribuídos, e então usam os dados fornecidos para determinar média, variância e covariância da distribuição. Alguns autores têm considerado redes neurais como métodos estatísticos paramétricos, uma vez que treinar uma rede neural geralmente significa encontrar valores apropriados para pesos pré-determinados.

c. Paradigma Baseado em Protótipo, uma forma de classificar um caso é lembrar de um caso similar cuja classe é conhecida e assumir que o novo caso terá a mesma classe. Essa filosofia exemplifica os sistemas baseados em protótipos, que classificam casos nunca vistos utilizando casos similares conhecidos.

Em sua forma mais simples, sistemas que empregam esse paradigma armazenam todos os exemplos de treinamento. A classificação é dada pela maior quantidade de exemplos vizinhos de uma dada classe. Essa abordagem é conhecida como *k*-vizinhos-mais-próximos (kNN, do inglês *k-nearest-neighbours*). Outras abordagens empregam heurísticas para selecionar os exemplos armazenados. Saber quais casos de treinamento devem ser memorizados é importante para evitar dificuldades e lentidão de manuseio por parte do modelo de classificação. O ideal é reter apenas os casos com os quais seja possível resumir toda a informação. Aha *et al.* (1991) descreve algumas estratégias para decidir quando um

novo caso deve ser memorizado. A medida de similaridade para os casos nos quais todos os atributos são contínuos pode ser calculada por meio de alguma distância entre esses atributos. Na presença de atributos ordinais essa medida se torna complicada, bem como na presença de atributos irrelevantes, os quais pode fazer com que dois casos similares sejam interpretados como muito diferentes. Métodos sensíveis ao contexto, que alterem a escala dos atributos, podem melhorar estas medidas (STANFILL e WALTZ, 1986).

d. Paradigma Conexionista, exemplo desse paradigma são as redes neurais que são construções matemáticas relativamente simples, que foram inspiradas em modelos biológicos do sistema nervoso humano. A representação de uma rede neural envolve unidades altamente interconectadas e, por esse motivo, o nome conexionismo é utilizado.

As pesquisas em redes neurais foram iniciadas com o trabalho pioneiro de McCulloch e Pitts (1943). McCulloch era um psiquiatra e pesquisou por vinte anos uma forma de representar um evento no sistema nervoso. A rede neural *Perceptron* foi apresentada por Rosenblatt (1962), cuja grande contribuição foi a prova do teorema de convergência.

A metáfora biológica com as conexões neurais do sistema nervoso tem interessado muitos pesquisadores e tem subsidiado discussões sobre os méritos e as limitações dessa abordagem de aprendizagem. Em particular, as analogias com a biologia têm levado muitos pesquisadores a acreditar que as redes neurais possuem um grande potencial na resolução de problemas que requerem processamento sensorial humano, tal como visão e reconhecimento de voz e imagens.

e. Paradigma Genético, este formalismo de classificação é derivado do modelo evolucionário de aprendizagem (HOLLAND e THAYER, 1988). Um modelo de classificação genérico consiste de uma população de elementos de classificação que competem para fazer a predição. Elementos que possuem um desempenho fraco são descartados, enquanto os elementos mais fortes proliferam, produzindo variações de si mesmos. Este paradigma possui

uma analogia direta com a teoria de Darwin, na qual sobrevivem os mais bem adaptados ao ambiente.

Alguns operadores genéticos básicos que aplicados a uma população geram novos indivíduos são Reprodução, Cruzamento, Mutação e Inversão. Esses operadores atuam no controle da quantidade de cópias produzidas de um indivíduo, na troca de material genético, na preservação de uma espécie e na manutenção de uma certa diversidade na nova população.

2.1.3 Tarefas de aprendizado

Em geral, no problema de aprendizado supervisionado, cada exemplo é descrito por um vetor de valores de características e por um atributo especial que descreve uma característica de interesse na qual se pretende criar o modelo conhecido como classe. Esse atributo pode ser discreto, ordinal ou contínuo. No caso do atributo discreto, o problema é conhecido como problema de classificação, e o objetivo é classificar futuros casos em cada uma das classes pré-definidas. Caso o atributo seja contínuo, o problema é geralmente conhecido como problema de regressão, e o objetivo é prever o valor desse atributo com base nas características dos exemplos. No caso do atributo alvo ser ordinal, o problema é conhecido como ordenação ou Regressão Logística, e o objetivo é ordenar um conjunto de casos de acordo com uma característica de interesse.

Mesmo que os problemas sejam definidos de acordo com o tipo do atributo alvo, é possível utilizar variáveis de outros tipos para cumprir a tarefa. Como por exemplo, é possível discretizar um atributo alvo contínuo e prever uma faixa de valores – atributo discretizado, ao invés de um valor contínuo. Também é possível prever um valor contínuo para um problema com atributo alvo discreto. Nesse caso, cada classe pode ser associada a uma faixa de valores contínuos. Além disso, é possível adequar essas faixas, para melhorar o desempenho do classificador.

2.1.4 Redução da dimensionalidade

Redução da dimensionalidade é um problema que se refere à dificuldade em determinar relações de proximidade entre os dados que possuem muitos atributos, pois o volume cresce exponencialmente com o número de dimensões. Quando a dimensão de um conjunto de dados é alta, torna-se menos significativa a diferença entre os dados mais próximos e os dados mais distantes. Dentre as formas de contornar esse problema destacam-se as duas abordagens Agregação que combina os atributos originais por meio de funções lineares ou não lineares (ZAMONER, 2013) e a principal que é seleção de atributos.

3 SELEÇÃO DE ATRIBUTOS

A tarefa de SA é utilizada para identificar os atributos mais significativos em um conjunto de dados, os quais podem contribuir para a construção de modelos de indução com menor custo computacional. A SA contribui para aumentar o conhecimento acerca das características mais importantes em uma base de dados. Outra vantagem de se trabalhar com a SA consiste na tarefa em tratar da maldição da dimensionalidade (HASTIE et *al,* 2001), tornando assim as amostras dos exemplos exponencialmente esparsas e, consequentemente, pode prejudicar a qualidade de modelos construídos com uma quantidade limitada de exemplos de treinamento (LIU e MOTODA, 2008). Estes modelos podem também serem afetados pela presença de atributos irrelevantes no conjunto de dados.

Várias aplicações reais apresentam um grande número de atributos. Além do problema da maldição da dimensionalidade, parte desses atributos pode ser irrelevante, redundante ou conter grande quantidade de ruído. A seleção de atributos permite:

- Identificar atributos importantes;
- Melhorar o desempenho de várias técnicas de AM;
- Reduzir a necessidade de memória e tempo de processamento;
- Eliminar atributos irrelevantes e reduzir ruído;
- Lidar com a maldição da dimensionalidade;
- Simplificar o modelo gerado e tornar mais fácil sua compreensão;
- Facilitar a visualização dos dados;
- Reduzir o custo de coleta de dados e com isso aumentar o acesso a novas tecnologias.

Serão utilizadas três abordagens para avaliar a qualidade ou o desempenho de um subconjunto de atributos: Embutida, baseada em filtro e baseada em *wrapper.*

3.1 Abordagens de avaliação de seleção de atributos

Os critérios de avaliação da importância de subconjuntos de atributos podem ser categorizados de acordo com a interação que realizam com o algoritmo de indução (KOHAVI e JOHN, 1997). As abordagens que serão avaliadas neste trabalhado para selecionar os atributos mais relevantes na base de dados são a embutida (*embedded*), a abordagem *wrapper* e a abordagem filtro.

3.1.1 Abordagem Embutida

Nesta abordagem é identificada quando a SA é realizada internamente pelo próprio indutor durante seu treinamento, ou seja, dado um conjunto de exemplos representado no formato atributo-valor, o próprio algoritmo de AM é capaz de decidir quais são os atributos relevantes para representar o conhecimento extraído, conforme ocorre, por exemplo, em árvores de decisão, nas quais a inserção de cada ramificação nova requer a seleção de um atributo (QUINLAN, 1993).

Ao contrário da abordagem embutida, nas estratégias mencionadas a seguir a SA é realizada como uma etapa separada e anterior ao processo de aprendizado do indutor.

3.1.2 Abordagem *Wrapper*

Ocorre externamente ao algoritmo de AM, porém utilizando o mesmo algoritmo como uma caixa preta para analisar o conjunto de atributos, embora nesta abordagem a aplicação do algoritmo de aprendizado esteja envolvida, essa interação está relacionada à avaliação da qualidade dos subconjuntos de atributos candidatos.

O critério de avaliação pode ser baseado na precisão obtida pelo modelo em uma projeção do conjunto de treinamento, a qual é definida a partir do subconjunto em análise, em outras palavras, métodos *wrapper* geram um subconjunto candidato de atributos, executam o algoritmo de indução considerando apenas este subconjunto de atributos selecionado do conjunto de treinamento, e utilizam a precisão resultante do classificador induzido para

avaliar o subconjunto de atributos em questão. Este processo é repetido para cada subconjunto de atributos até que o melhor subconjunto de atributos seja encontrado. Conforme mostra a Figura 3.

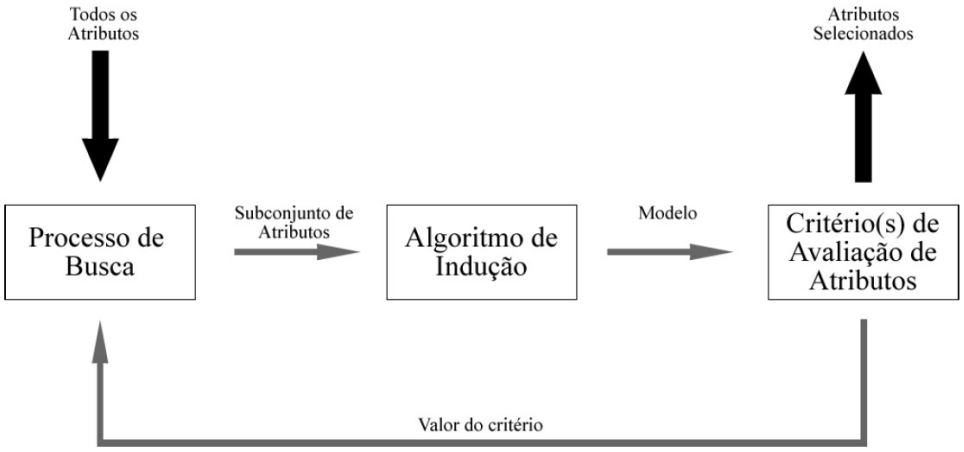


Figura 3. Fluxograma – Abordagem Wrapper

3.1.3 Abordagem Filtro

Com esta abordagem não necessita do auxílio de um indutor para a avaliação dos subconjuntos de atributos (YU e LIU, 2003). Os critérios pertencentes a esta abordagem são definidos a partir de propriedades intrínsecas dos dados. Segundo Blum e Langley (2007), um dos esquemas mais simples de filtragem é a avaliação de cada atributo, baseada na sua correlação com o atributo alvo, escolhendo os *k* atributos que fornecem o melhor valor. Veja a Figura 4 que mostra como é o esquema desta abordagem.

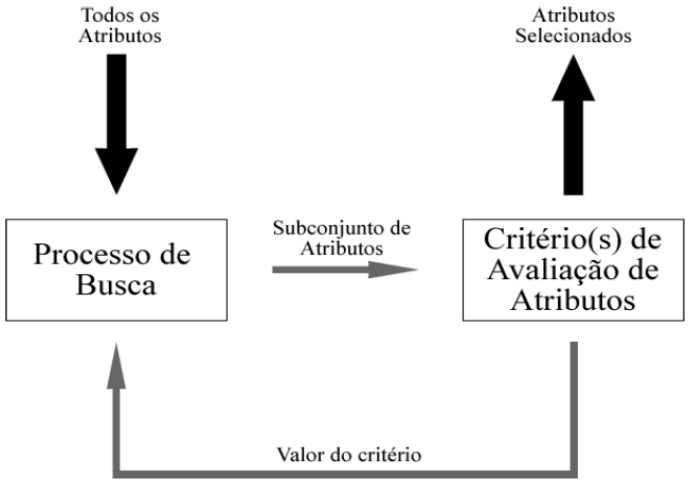


Figura 4. Fluxograma – Abordagem Filtro

4 MODELOS PREDITIVOS

Um algoritmo de AM é dito preditivo quando é induzido a partir de um conjunto de exemplos identificados e classificados. A classe toma valores num domínio conhecido. Se este domínio for um conjunto de valores nominais, tem-se um problema de classificação, conhecido também como aprendizado de conceitos, gerando um classificador, assim tem-se (i) um conjunto de classes; (ii) um conjunto de instâncias e (iii) um classificador (DIETTERICH, 1998).

A partir disso, a classificação é uma atividade realizada pelo classificador, de atribuir uma classe a cada uma das instâncias. Para executar a tarefa de classificação, é necessário construir o classificador, ou método de classificação, que será responsável por fazer as atribuições entre os elementos do conjunto de amostras e os elementos do conjunto das classes. Para se construir o classificador, é preciso dispor de um conjunto de treinamento, ou seja, um conjunto de amostras cujas classes sejam previamente conhecidas. Após o seu treinamento, o classificador é exposto a uma série de amostras cujas classes são desconhecidas para ele prever as classes dessas amostras (BREVE e ZHAO, 2012). Os modelos de classificação utilizados neste trabalho serão brevemente descritos a seguir:

4.1 Árvores de Decisão

Um classificador baseado em Árvore de Decisão é aquele que expressa uma aproximação para os valores resultantes de uma função-alvo cuja imagem é formada por valores discretos e utiliza uma estrutura de árvore para representar essa estrutura (MITCHELL, 1997).

A construção ou a estratégia básica de aprendizado das árvores de decisão é a aprendizagem não-incremental através de exemplos (QUINLAN, 1986). Ou seja, na medida em que os registros de treinamento são apresentados, a árvore é construída de cima para baixo, sendo que esse processo não é guiado pela ordem com que as informações aparecem, mas sim por um critério de frequência da informação cuja variante mais usada é o ganho de informação. Assim, inicialmente, o atributo que trouxer o maior ganho de informação é o que mais segmentará o conjunto de treinamento entre as classes e será o escolhido como nó raiz da árvore. E em seguida, cada valor desse atributo definido como nó raiz gera um ramo descendente da árvore. O processo é reiniciado para a definição de qual será o atributo que dará origem ao próximo nó da árvore. O ganho de informação é definido pela Equação 01, onde *S* é um conjunto de exemplos; A_i é um atributo; *v* é um valor do atributo e $atributo_v$ é o subconjunto de *S* no qual o valor desse *atributo* é *v*. E a entropia é a medida mais conhecida para impuridade baseada na entropia de Shannon (SHANNON, 1948), conforme a Equação 02.

$$GanhoInfo(S, atributo) = Entropia(S) - \sum_{v \in Dom(atributo)} \frac{|atributo|}{|S|} \times Entropia(atributo_v) \quad (1)$$

$$Entropia(S) = -p + log_2 p - p - log_2 p \quad (2)$$

A "pureza" dos subconjuntos obtidos pelos atributos é a medida comparando a impuridade do nó pai com a impuridade ponderada dos filhos, conforme a Equação 02 na qual *p*+ é a proporção de exemplos positivos em *S* e *p* – é a proporção de exemplos negativos.

A construção da árvore pode continuar indefinidamente até que o ganho de informação praticamente seja zero. No entanto, quando isso acontece a árvore tende a tornar-se muito específica para o conjunto de treinamento, ou seja, apresenta um alto grau de precisão para os exemplos de treinamento e uma alta taxa de erro para o conjunto de teste,

caracterizado como *overfitting*[1], em geral, é definido um limiar do critério de particionamento para determinar se a construção da árvore continua ou interrompe naquele ponto (HAN e KAMBER, 2011).

Uma das grandes vantagens do algoritmo de Árvore de Decisão é a facilidade de interpretação do conceito por ela representado, sendo assim amplamente utilizada em mineração de dados e em tarefas que requerem a validação semântica da hipótese obtida. Outra vantagem é que a classificação de exemplos usando uma Árvore de Decisão não requer a avaliação de todos os atributos que compõem a árvore. Devido a isso, o classificador expresso na linguagem de árvores de decisão é um dos mais rápidos entre os algoritmos de aprendizado de máquina existentes (TAKASHI, 2008).

4.2 Redes Neurais Artificiais

As Redes Neurais Artificiais – RNA têm se tornado o foco de muita atenção, devido a sua ampla aplicabilidade e, principalmente, por tratar de casos considerados complicados. RNA podem identificar e aprender padrões relacionando conjunto de dados de entrada e valores de saída correspondentes, após o treinamento, a RNA podem ser usadas para predizer o resultado relacionado a um novo grupo de dados de entrada. Elas podem resolver problemas com dados não lineares e complexos, mesmo sendo dados imprecisos e ruidosos (PEREIRA e CENTENO, 2017).

A RNA tem sido descrita como um processador maciçamente paralelo, constituído por unidade de processamento simples, que tem uma tendência natural para armazenar conhecimento empírico e torná-lo disponível para o uso (HAYKIN; LIPPMANN, 1994). São

[1] ***Overfitting*** **–** quando um classificador é induzido, é possível que ele seja muito específico para o conjunto de treinamento, apresenta um alto grau de precisão para o conjunto de exemplos de treinamento e uma alta taxa de erro para o conjunto de teste.

poderosas ferramentas para modelagem, especialmente quando são desconhecidas as relações entre os dados (LEK; GUÉGAN, 1999).

A ideia principal da RNA está no cálculo inspirado no sistema biológico, no cérebro humano, resumidamente, uma RNA compreende um grande número de unidades de processamento simples, os nós ou neurônios, organizados em camadas, conectados por ligações ponderadas de acordo com uma arquitetura específica (PEREIRA e CENTENO, 2017).

Usualmente, as camadas são classificadas em três grupos:

- Camada de Entrada: onde os padrões são apresentados à rede;
- Camadas Oculta ou Intermediárias: onde é feita a maior parte do processamento, através das conexões ponderadas; podem ser consideradas como extratoras de características;
- Camada de Saída: onde o resultado final é concluído e apresentado.

A representação de uma RNA com os neurônios dispostos nas camadas de entrada, camada oculta e a camada de saída. O número de neurônios em cada camada pode ser variável, assim como o número de camadas ocultas, conforme apresentada na Figura 5.

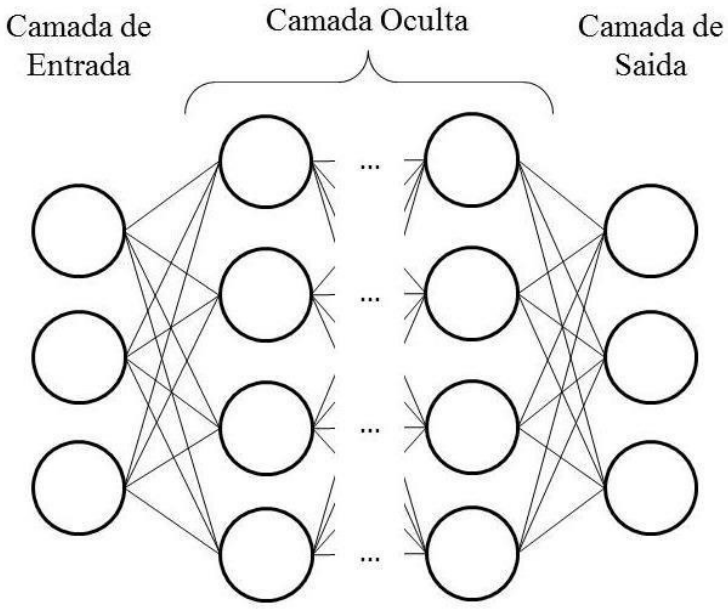


Figura 5. Representação Gráfica de RNA

Conforme Kira e Rendell (1992) todos os nós da RNA, exceto os de entrada, executam as mesmas funções: recolhem os valores transferidos da camada anterior e definem um resultado de saída. Os nós de entrada são determinados pelos dados de entrada, essa função que recolhe os valores da camada anterior é dada pela Equação 03.

$$net_{pi} = \sum_{j} w_{ij} a_{pj} + bias_i \tag{3}$$

Onde w_{ij} representa o peso da ligação de um nó *j* da camada anterior com o nó *i* da camada atual, a_{pj} é o valor passado do nó anterior *j* do padrão *p*, e o *bias* representa um termo independente, podendo ser considerado caso o nó atual *i* sempre possua ativação.

As primeiras unidades básicas de processamento, ou *perceptrons*, do modelo computacional, de modo que o processo de aprendizado de uma RNA possa ser visto como o problema de atualizar a arquitetura da rede e os pesos das interconexões para que a rede possa realizar uma tarefa específica de modo eficiente. Para que haja o aprendizado são atribuídos pesos adequados aos atributos. O Algoritmo 01 mostra o funcionamento da mais utilizada RNA *gradiente descendente*: (OLIVEIRA, 2015).

Para classificação padrões de entrada como pertencentes ou não a uma dada classe, seja o conjunto de treinamento formado por N amostras $\{\mathbf{x}_1, d_1\}$, $\{\mathbf{x}_2, d_2\}$, ..., $\{\mathbf{x}_N, d_N\}$, onde $\mathbf{x}_j$ é o vetor de entradas e d_j a saída desejada, que em notação vetorial tem-se $\{\mathbf{X}, \mathbf{d}\}$, onde: $\mathbf{X} \in \Re^{mxN}$ e $\mathbf{d} \in \Re^{1xN}$.

Algoritmo 01. Rede Neural Artificial

```
Procedure [w] = perceptron (max_it, E, a, X,d)
inicializar w     // para simplicidade, com zeros
inicializar b     // para simplicidade, com zero
t ← 1
while t < max_it & E > 0 do
```

```
    for i from 1 to N do            // para cada padrão de entrada
        y_i ← f(w x_i + b)          // determinar a saída
        e_i ← d_i − y_i             // determinar o erro
        w ← w + a e_i x_i           // atualizar o vetor peso
        b ← b + a e_i               // atualizar o bias
    end for
    E ← sum (e_i)                   //quantidade de erros
    t ← t + 1
  end while
end procedure
```

Além desse modelo mais simples, há também a RNA multicamadas, que podem conter diversas camadas intermediárias entre as camadas de entrada e saída. Em uma rede neural com alimentação para frente, os nós de uma camada somente estão ligados aos nós da camada seguinte, já que em uma rede neural recorrente, os nós podem estar ligados a nós da camada seguinte, da mesma camada ou até mesmo de camadas anteriores. A função de ativação também pode ser diferente da descrita acima para permitir que os nós das camadas intermediárias produzam valores que não sejam lineares com seus parâmetros de saída, sendo que o algoritmo chamado *backpropagation* que é uma extensão do algoritmo *perceptron* é o mais utilizado para uma rede multicamada (OLIVEIRA, 2015).

O processo de treinamento do algoritmo *backpropagation* é baseado em tentativa e erro, por isso, um dos problemas deste algoritmo é seu tempo de treinamento e o que influência nesse tempo é a taxa de atualização dos pesos, se a taxa definida for muito baixa a rede consome um tempo muito grande para o treinamento, por outro lado se a taxa é alta a rede consegue convergir em um pequeno espaço de tempo, porém, quando é apresentada uma outra entrada a rede se torna instável ocasionando outro problema que é a confiabilidade dos resultados.

4.3 Regressão Logística

Os métodos de regressão objetivam o entendimento da relação entre um conjunto de variáveis independentes (ou explicativas) e uma variável dependente (ou resposta) para construir um modelo para explicar essa associação, conforme a Estatística; com a construção do modelo, é possível prever o valor que a variável dependente terá diante dos valores das variáveis independentes, fazendo com que o método possa, diante de algumas circunstâncias, atuar como um classificador (ALLISON, 1999).

Ao se aplicar o método de regressão, um item chave é o valor médio do atributo resposta dado valores para cada um dos atributos do conjunto de atributos independentes (aqueles que ajudam a explicar o fenômeno e das quais o valor médio do atributo resposta depende), esse valor médio é chamado de *média condicional*, ou $E(Y \mid x)$. Assim, tem-se a Equação 04:

$$E(Y|x) = \beta_0 + \overrightarrow{\beta_\iota}\,\overrightarrow{X_\iota} \tag{4}$$

Onde:

- $\overrightarrow{X_\iota}$ é o vetor dos atributos explicativos;
- $\overrightarrow{\beta_\iota}$ é o vetor de coeficientes.

Quando o modelo considera mais de um atributo para tentar prever a existência da condição de interesse, é desejável conhecer o quanto cada um desses atributos contribui na formação do valor resposta para, inclusive, retirar um ou mais desses atributos do modelo caso a contribuição seja baixa ou até mesmo nula (OLIVEIRA, 2015).

Na técnica de Regressão Logística essa tarefa é mais usualmente realizada através do Teste da Razão de Verossimilhança, que consiste de uma série de etapas que calculam a relevância dos atributos, suportando a inclusão ou exclusão delas do modelo mediante uma regra bem definida. Em termos práticos, a relevância dos atributos é traduzida por uma

medida de significância estatística dos coeficientes β que acompanham esses atributos na função resposta. No caso da Regressão Logística os erros seguem a distribuição binomial, feito o uso do Teste da Razão de Verossimilhança, portanto, em cada etapa do teste, o atributo mais importante será o que produzir uma maior alteração no logaritmo da verossimilhança em relação ao modelo que não contém o atributo assim sendo, o Teste da Razão de Verossimilhança é baseado na estatística expressa pela Equação 05 (HOSMER e LEMESHOW, 2000).

$$D = -2\log\left[\frac{verossimilhança\ do\ modelo\ atual}{verossimilhança\ do\ modelo\ saturado}\right] \quad (5)$$

Para estimar a significância de um coatributo, calculam-se os valores da estatística *D* para o modelo com e sem o atributo desejado, obtendo-se, assim, a estatística *G*, como a Equação 06.

$$G = D(modelo\ sem\ o\ coatributo) - D(modelo\ com\ o\ coatributo) \quad (6)$$

Apesar de várias funções de distribuição já terem sido propostas para a análise de atributos resposta categóricas, a distribuição logística possui duas vantagens significativas: facilidade e flexibilidade de manipulação matemática e facilidade de interpretação e apesar da existência de outros testes para verificar a significância dos atributos explicativos, como por exemplo, o teste de *Wald* e o teste de *Escore*, sendo o teste mais recomendado na literatura é o da Razão de Verossimilhança (OLIVEIRA, 2015).

4.4 Avaliação de modelos preditivos

A avaliação de um modelo de AM pode ser realizada seguindo diferentes aspectos, tais como acurácia do modelo gerado, compreensibilidade do conhecimento extraído, tempo de aprendizado, requisitos de armazenamento do modelo e outros. Neste trabalho os modelos

preditivos serão avaliados conforme medidas relacionadas ao desempenho obtido nas predições realizadas, tais como:

- Métricas de erro: que são avaliação analisadas conforme o desempenho do classificador gerado por ele na rotulação de novos objetos, não apresentados previamente em seu treinamento (MONARD e BARANAUSKAS, 2003).

- Amostragem: utiliza-se métodos alternativos para obter estimativas de desempenho preditivos mais confiáveis, definindo os subconjuntos de treinamento e de teste. Os dados de treinamento são empregados na indução e no ajuste do modelo, enquanto os exemplos de teste simulam a apresentação de objetos novos ao classificador, os quais não foram vistos em sua indução. Esses subconjuntos são disjuntos para assegurar que as medidas de desempenho sejam obtidas a partir de um conjunto de exemplos diferente daquele usado no aprendizado (FACELLI *et al*., 2011).

4.4.1 Matriz de contingência

A matriz de contingência de um classificador oferece uma medida efetiva do modelo de classificação, uma vez que apresenta o número de classificações corretas versus as classificações preditas pelo modelo para cada classe. Dado um conjunto de teste T_e e um conjunto de categorias $C = \{c_1, c_2, ..., c_{|C|}\}$, os resultados são totalizados em duas dimensões: classificação verdade e classificação predita, onde:

$$M(c_i, c_j) = \sum_{\{d_k \in T_e : \Phi(d_k, c_i) = 1\}} \left|\left|\Phi(d_k, c_i) = 1\right|\right| (23) \quad (7)$$

Classe	Predita c_1	Predita c_2	...	Predita $c_{\|C\|}$
Verdadeira c_1	M(c_1, c_1)	M(c_1, c_2)	...	M(c_1, $c_{\|C\|}$)
Verdadeira c_2	M(c_2, c_1)	M(c_2, c_2)	...	M(c_1, $c_{\|C\|}$)
...	...	...	...	...
Verdadeira $c_{\|C\|}$	M($c_{\|C\|}$, c_1)	M($c_{\|C\|}$, c_2)	M($c_{\|C\|}$, ...)	M($c_{\|C\|}$, $c_{\|C\|}$)

Quadro 2. Matriz de contingência de um classificador

O número de acertos para cada classe localiza-se na diagonal principal da matriz, os demais elementos representam erros na classificação. A matriz de contingência de um classificador ideal possui todos esses elementos iguais à zero. Para simplificar, considere um problema de classificação binária de classe c_i. O problema deve classificar documentos de teste em c_i ou $\overline{c_i}$, assim temos a seguinte matriz de contingência:

	Preditos Positivos	Preditos Negativos	
Exemplos Positivos	*VP*	*FN*	Pos
Exemplos Negativos	*FP*	*VN*	Neg
	PPos	PNeg	Total

Quadro 3. Matriz de contingência de um classificador binário

Onde VP, FP, VN e FN significam respectivamente, o número de exemplos verdadeiros positivos, falsos positivos, verdadeiros negativos e falsos negativos, Pos é o total de exemplos positivos; Neg é o total de exemplos negativos; PPos é o total de exemplos preditos positivos; PNeg é o total de exemplos preditos negativos e Total é o número total de exemplos no conjunto de exemplos considerado.

Com os valores da matriz de contingência é possível definir as seguintes medidas, onde, TVP indica a taxa de verdadeiros positivos e TFP a taxa de falsos positivos:

$$TVP = \frac{positivos\ classificados\ corretamente}{total\ de\ positivos} = \frac{VP}{Pos} \tag{8}$$

$$TFP = \frac{negativos\ classificados\ incorretamente}{total\ de\ negativos} = \frac{FP}{Neg} \tag{9}$$

A acurácia é a razão entre a quantidade de casos corretamente classificados pelo modelo e todos os casos que passaram pelo classificador:

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN} \tag{10}$$

A taxa de acerto é a mais conhecida e talvez a mais utilizada. No entanto, quando utilizada em problemas de classes desbalanceadas, ela pode esconder o erro de predição da classe minoritária. Isso pode ser verificado ao observar o termo *Total* que soma indistintamente exemplos positivos e negativos na fórmula da *taxa de acerto*.

$$Taxa\ de\ Acerto = \frac{VP + VN}{Total} \tag{11}$$

4.4.2 Análise da curva ROC

O Acrônimo de *Receiver Operating Characteristic* – ROC, teve suas origens na área de Detecção de Sinais e é usado para designar a relação entre taxa de acerto e a taxa de falsos alarmes em um canal com ruídos, sendo assim, a curva ROC é um gráfico no qual o eixo das ordenadas é dado pela taxa de verdadeiros positivos – TVP, enquanto o eixo das abcissas é dado pela taxa de falsos positivos – TFP. Logo, a área sob essa curva é tida como uma medida de qualidade do classificador, pois quanto maior a área, melhor o desempenho do classificador (HOSMER e LEMESHOW, 2000). A Tabela 1 fornece um mapa para através do valor da área sob a curva ROC (AROC), definir o poder de classificação de um modelo:

Valor da AROC	**Poder de Classificação**
AROC = 0,5	Não há
0,7 ≤ AROC < 0,8	Aceitável
0,8 ≤ AROC < 0,9	Muito bom
AROC ≥ 0,9	Excelente

Tabela 1. Poder de classificação de um modelo dados pela AROC

O gráfico ROC é bidimensional no qual os eixos Y e X do gráfico sempre representam as medidas *TVP* e *TFP* respectivamente, assim como exemplificado na Figura 6 é identificado as quatro regiões importantes do gráfico, descritas e uma linha diagonal que representa classificadores aleatórios:

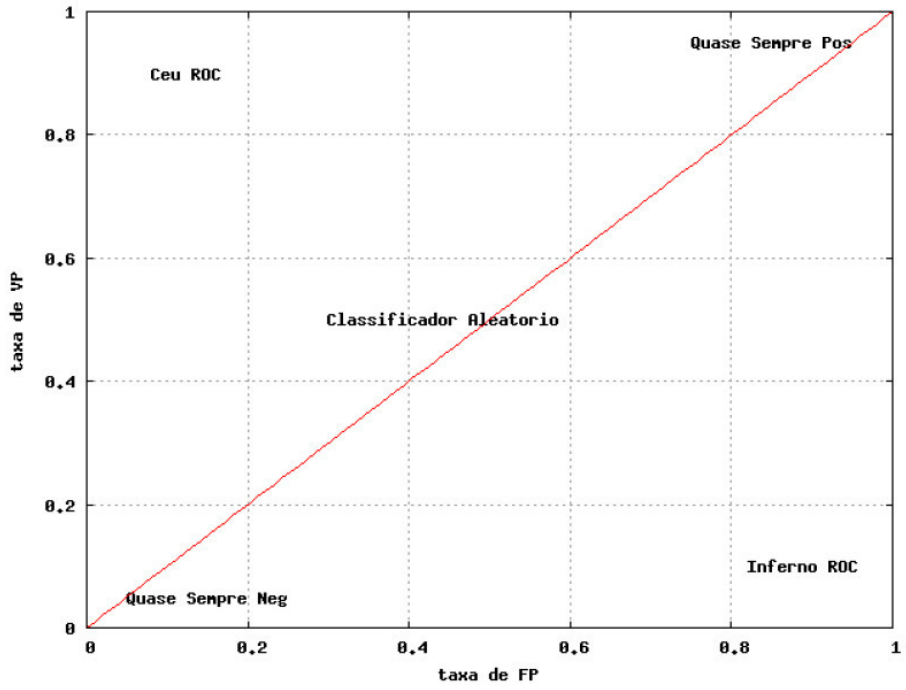


Figura 6. Gráfico ROC

Céu ROC: o ponto (0, 1) representa uma classificação perfeita, na qual todos os exemplos positivos e negativos são rotulados corretamente. Na região *Céu ROC* encontram-se os pontos mais próximos da classificação perfeita e representam bons resultados. Quando classificadores atingem valores muito altos, ou seja, tem, segundo ela, poder de classificação excelente, portanto, quando um classificador atinge esse patamar de resultado, tem-se que aprofundar as análises para verificar se o classificador não foi acometido da Maldição da Dimensionalidade ou sofreu *overfitting*.

Inferno ROC: região localizada no lado oposto ao *Céu*, pode ser considerada uma região na qual são encontrados os resultados "ruins". No entanto, classificadores

representados nessa região possuem informações com capacidade de distinguir as classes, mas não as utilizam corretamente (FLACH e WU, 2005).

Quase Sempre Neg: classificadores que são representados nessa região rotulam quase sempre os exemplos como negativos.

Quase Sempre Pos: classificadores que são representados nessa região rotulam quase sempre os exemplos como positivos.

Normalmente, os classificadores representados por pontos próximos a linha diagonal são considerados classificadores aleatórios, não possuem informação sobre a classe, classificadores aleatórios sorteiam aleatoriamente a classe que atribuem aos exemplos (TAKASHI, 2008).

5 BASE DE DADOS INCT – HERBÁRIO VIRTUAL DA FLORA E DOS FUNGOS

O Centro de Referência em Informação Ambiental (CRIA) foi criado em 2000 como uma associação sem fins lucrativos com o objetivo de disseminar informações científicas e, dessa forma, contribuir para a conservação e uso sustentável dos recursos biológicos brasileiros. Para cumprir sua missão, a CRIA é responsável pelo desenvolvimento de *speciesLink*, uma e-infraestrutura de biodiversidade conceituada em 2001 no âmbito do Programa Biota-FAPESP - Instituto Virtual de Biodiversidade, voltada para a biodiversidade do Estado de São Paulo, depois todo o país graças ao financiamento do Ministério da Ciência, Tecnologia e Inovação e ao envolvimento de sociedades científicas e coleções biológicas. Atualmente, o *speciesLink* oferece acesso livre e aberto a cerca de 8,2 milhões de registros primários de pesquisa, de biodiversidade de 449 conjuntos de dados de 135 instituições nacionais, abrangendo todos os estados brasileiros e de 13 instituições do exterior (CANHOS *et al,* 2015).

Atualmente, há uma cobertura nacional que inclui pelo menos um provedor de dados em cada estado do Brasil, sendo cerca de 70% de dados são de herbários, do INCT – Herbário Virtual da Flora e dos Fungos, e que atualmente integra 105 herbários nacionais associados e 25 herbários do exterior, além de 15 herbários nacionais não associados, mas que também compartilham seus dados *on-line*, juntos disponibilizam mais de 5,4 milhões de registros e mais de um milhão de imagens, além de várias ferramentas de livre acesso (SPLINK, 2016).

A missão do INCT – Herbário Virtual da Flora e dos Fungos é prover infraestrutura de dados de qualidade de acesso público e aberto integrando as informações dos acervos dos

herbários do país e repatriando dados sobre coletas realizadas em solo brasileiro depositadas em acervos no exterior. (SPLINK, 2016).

Dentre os vários objetivos do Herbário Virtual ressalta-se:

- Tornar o compartilhamento livre e aberto de dados e informações não sensíveis de herbários em formato utilizável, que é um princípio fundamental da conduta científica;
- Melhorar a qualidade dos acervos dos herbários brasileiros;
- Tornar os dados sobre a ocorrência de espécies no Brasil, base fundamental para a tomada de decisão e formulação de políticas públicas sobre a biodiversidade;
- Ampliar a base de conhecimento sobre a diversidade da flora e dos fungos macroscópicos do Brasil.

Sua ação focal é na transferência de conhecimento para a sociedade quanto à determinação do nome científico de um espécime e sua divulgação em um sistema de acesso livre e aberto é parte fundamental da estratégia de transferência do conhecimento taxonômico para a sociedade. E é através do conhecimento taxonômico representado por um nome científico, integrando assim de forma dinâmica dados, informações e conhecimento de diferentes acervos e produzir informações que possam subsidiar a análise de especialistas dos mais diversos seguimentos tais como: agricultura, meio ambiente, saúde e indústria. (SPLINK, 2016).

O acesso aos dados do INCT – Herbário Virtual da Flora e dos Fungos foram coletados a partir do site do *speciesLink*[2], no formulário de busca foi inserido o texto "herbários", como mostra a Figura 7, em que retornou com os dados do INCT – Herbário Virtual da Flora e dos Fungos.

Esta base de dados contém 87.732 registros e 51 atributos, sendo 119 coleções e sub-coleções, 86.967 registros *online*, 80.513 registros georreferenciados, 12.073 espécies aceitas

[2] http://inct.splink.org.br/

distintas. Sendo estes atributos assim nomeados, conforme mostra a Quadro 4, de sua descrição e o respectivo tipo de dado.

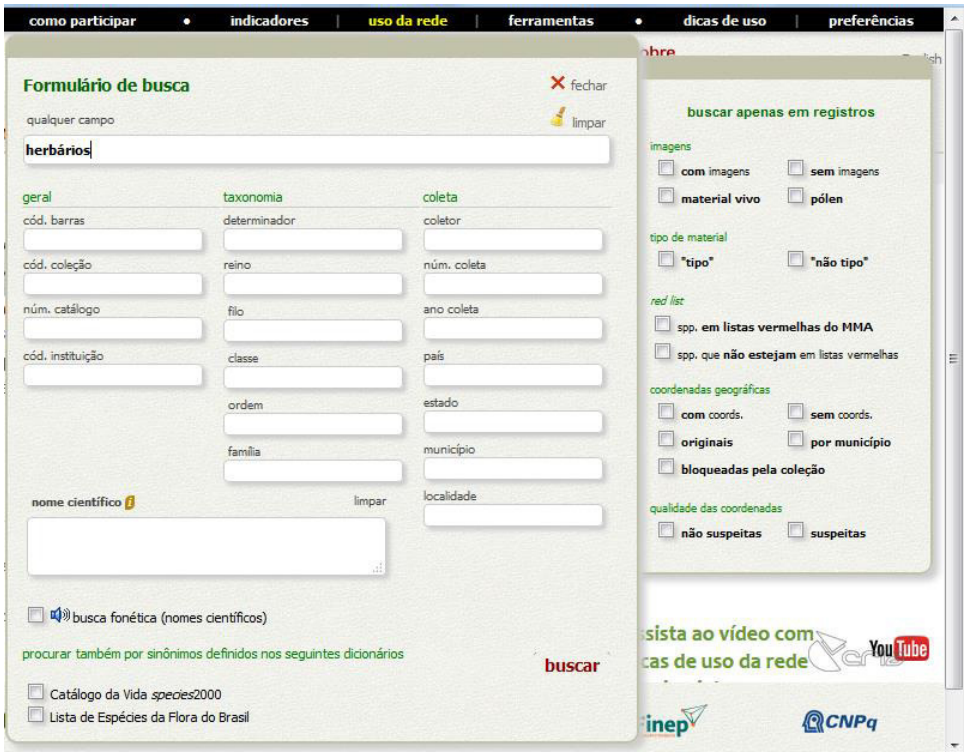


Figura 7. Formulário de busca do site speciesLink

Descrição dos atributos da base de dados do Herbário Virtual da Flora e dos Fungos:

Nº	Atributo	Descrição	Tipo de Dado
1	datelastmodified	Data da última modificação	Caracter
2	institutioncode	Código da instituição	Caracter
3	collectioncode	Código da coleção	Caracter
4	catalognumber	Número do catálogo	Caracter
5	scientificname	Nome científico	Caracter
6	basisofrecord	Base de registro	Caracter
7	kingdom	Reino	Caracter
8	phylum	Filo	Caracter
9	class	Classe	Caracter
10	ordem	Ordem	Caracter
11	family	Família	Caracter

12	genus	Gênero	Caracter
13	species	Espécies	Caracter
14	subspecies	Subespécies	Caracter
15	scientificnameauthor	Autor da espécie	Caracter
16	idenfifiedby	Autor da determinação	Caracter
17	yearidentified	Ano da identificação	Inteiro
18	monthidentified	Mês da idetificação	Inteiro
19	dayidentified	Dia da identificação	Inteiro
20	typestatus	Status do tipo	Caracter
21	collectornumber	Número do coletor	Caracter
22	fieldnumebr	Número da expedição	Caracter
23	collector	Nome do coletor	Caracter
24	yearcollected	Ano da coleta	Inteiro
25	monthcollected	Mês da coleta	Inteiro
26	daycollected	Dia da coleta	Inteiro
27	julianday	Data	Lógico
28	timeofday	Hora do dia	Caracter
29	continentocean	Continente	Caracter
30	country	País	Caracter
31	stateprovince	Estado	Caracter
32	county	Cidade	Caracter
33	locality	Localidade	Caracter
34	longitude	Longitude	Caracter
35	latitude	Latitude	Caracter
36	longitude_mun	Longitude do município	Inteiro
37	latitude_mun	Latitude do município	Inteiro
38	coordinateprecision	Coordenada de precisão	Inteiro
39	boundingbox	Caixa delimitadora	Lógico
40	minimumelevation	Elevação mínima	Inteiro
41	maximumelevation	Elevação máxima	Inteiro
42	minimumdepth	Profundidade mínima	Lógico
43	maximumdepth	Profundidade máxima	Lógico
44	sex	Sexo	Caracter

45	preparationtype	Tipo de preparação	Caracter
46	individualcount	Contador individual	Inteiro
47	previouscatalognumber	Número do catálogo anterior	Caracter
48	relationshiptype	Tipo de relação	Caracter
49	relatedcatalogitem	Item relacionado ao catálogo	Caracter
50	notes	Notas	Caracter
51	barcode	Código de barra	Caracter

Quadro 4. Descrição dos atributos da base de dados e seus respectivos tipos

Os atributos serão trabalhados com seus respectivos nomes em inglês para facilitar a programação em linguagem R, uma vez que não precisa de acentos como é o caso do nosso idioma português e as palavras compostas sem espaço. Na linguagem R são trabalhados com os tipos de dados: lógico, numérico que pode ser do tipo inteiro ou real e caracter que são dados de textos. E os dados do tipo *Factor* são utilizados para armazenar dados categóricos e são caracterizados por conterem apenas valores pré-definidos chamados de níveis, são esses tipos de dados que são trabalhados em um sistema de AM preditivo.

6 METODOLOGIA

A metodologia para seleção de atributos relevantes proposta neste trabalho tem como base as técnicas de abordagens do tipo filtro, *wrapper* e embutida, descritas na Seção 3.2, utilizando a base de dados do Herbário Virtual da Flora e dos Fungos apresentada na Seção 6.

Foi utilizada a linguagem R para o desenvolvimento dos modelos preditivos desta pesquisa; sendo esta de código aberto, livremente distribuído e proporciona um ambiente para análises de dados estatísticas, com recursos gráficos de alta qualidade e programação, está disponível no site do CRAN[3] - *The Comprehensive R Archive Network* (TEAM, 2013). Com o auxílio da IDE - *Integrated Development Environment,* RStudio trazendo várias funcionalidade, oferecendo mais facilidade de uso desta linguagem (TEAM, 2014).

As fases da metodologia são descritas conforme mostra a Figura 8:

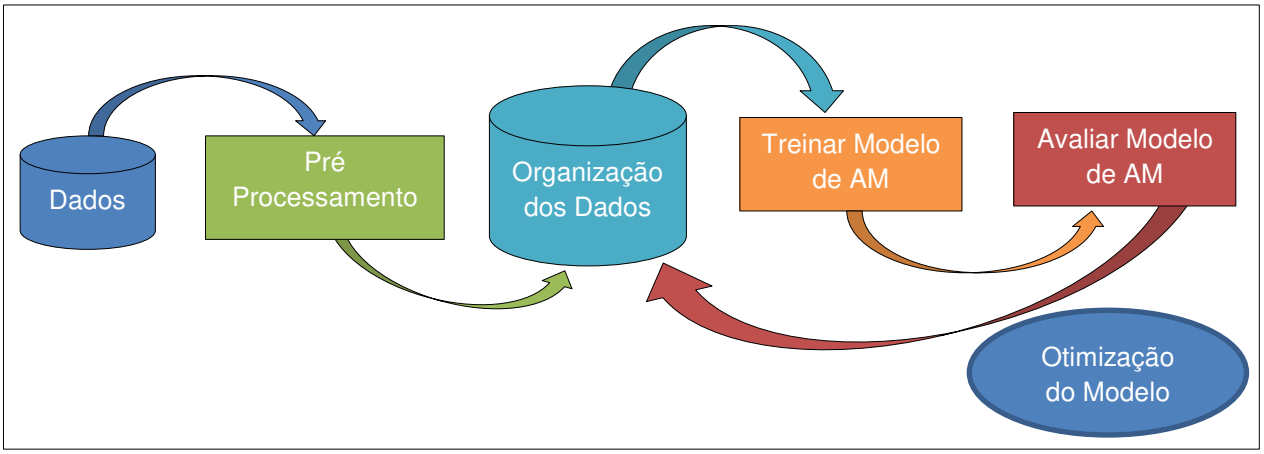


Figura 8. Fluxograma - Metodologia da pesquisa

A primeira fase é a importação da base de dados para o ambiente de programação. Após a importação temos o pré-processamento que é uma das fases mais trabalhosa de AM, depois dessa etapa onde já teremos os dados limpos e estruturados é feito a organização dos dados para receber o algoritmo para treinar com os atributos selecionados e avaliar com que precisão o modelo conseguiu acertar com os dados de teste, se caso ainda não conseguiu um

[3] http://www. r-project.org

resultado satisfatório é feito a otimização do modelo reorganizando os dados e novamente é feito o treino do modelo, ficando neste ciclo que com alguns parâmetros alterados consiga chegar a um nível de acertos satisfatório para o modelo de AM.

6.1 Pré-processamento

Em grandes bancos de dados é comum a presença de dados incompletos (ausência de atributos), ruídos (presença de erros ou valores que desviam muito do esperado – *outliers*) e dados inconsistentes (HAN, 2011). O pré-processamento dos dados, é uma das etapas mais importantes para a mineração de dados, podendo corresponder a 80% de todo o processo (ZHANG *et al.*, 2011), as etapas de pré-processamento são ilustradas na Figura 9.

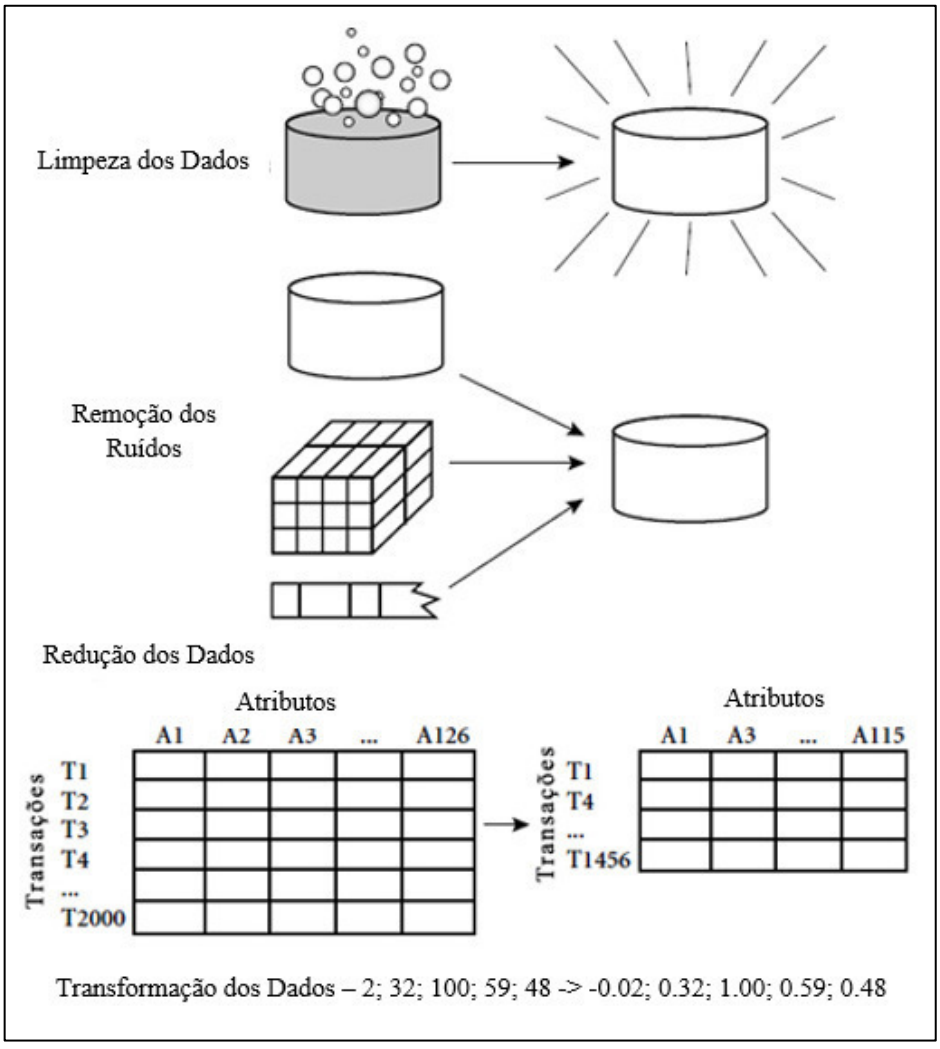


Figura 9. Atividades do pré-processamento

6.1.1 Limpeza de dados

A limpeza dos dados é o processo responsável em ajustar os dados para evitar que ocorram erros ao se aplicar técnicas de aprendizado de máquina (HAN, 2011). A limpeza de dados pode ser realizada através de tratamento para dados faltantes e a suavização de ruídos, utilizando técnicas citadas por Han (2011) como: ignorando a tupla – instância completa formada pelos atributos de um objeto; ou suprindo os valores ausentes – de forma manual, através de uma constante global, utilizando a média do atributo, ou com o valor mais provável com recursos da regressão, inferência ou outro método que possa ser utilizado para suprir a falta do valor.

6.1.2 Remoção de ruídos

O ruído pode ser definido como erros aleatórios que ocorrem durante o processo de aquisição de dados. Pode ocorrer devido a diversos fatores como mau funcionamento dos equipamentos de coleta de dados, como também de fatores ambientais.

6.1.3 Redução de dados

O volume de dados usado nas bases de dados utilizadas na AM são sempre alto. Em alguns casos, este volume é tão grande que torna o processo de análise dos dados e da própria AM impraticável. Nestes casos, as técnicas de redução de dados podem ser aplicadas para que a massa de dados original seja convertida em uma massa de dados menor, porém, sem perder a representatividade dos dados originais. Isto permite que os algoritmos de AM sejam executados com mais eficiência, mantendo a qualidade do resultado. As estratégias adotadas nesta etapa são: criação de estruturas otimizadas para os dados, seleção de um subconjunto dos atributos – foco desta dissertação, redução da dimensionalidade e discretização.

6.1.4 Transformação de dados

Várias técnicas de AM estão limitadas à manipulação de valores de determinados tipos, por exemplo, apenas valores numéricos ou apenas valores categóricos. Algumas técnicas empregadas nesta etapa são: suavização (remove valores errados dos dados), agrupamento (agrupa valores em faixas sumarizadas), generalização (converte valores muito específicos para valores mais genéricos), normalização (colocar as variáveis em uma mesma escala) e a criação de novos atributos (gerados a partir de outros já existentes). Adicionalmente, algumas técnicas têm seu desempenho influenciado pelo intervalo de variação dos valores numéricos (FACELLI *et al*., 2011).

6.2 Organização dos dados

Geralmente, a base de dados usada possui milhares de registros. Neste contexto, o uso de todos os registros do repositório para a construção do modelo de AM é inviável. Assim, utiliza-se uma amostra (mais representativa possível) que é dividida em três conjuntos:

- Conjunto de Treinamento: conjunto de registros usados no qual o modelo é desenvolvido. Onde a partir deste que o modelo irá fazer suas inferências e procurar relações entre os dados e buscar uma função matemática que poderá prever novas classes com estas que foram aprendidas.

- Conjunto de Testes: conjunto de registros usados para testar o modelo construído com o conhecimento adquirido a partir do conjunto de treinamento sendo que este conjunto de dados não possuem justamente a classe, pois a mesma será predita conforme o modelo aprendeu.

- Conjunto de Validação: conjunto de registros usados para validar o modelo construído.

Essa divisão é necessária para que o modelo não fique dependente de um conjunto de dados específico e, ao ser submetido a outros conjuntos (com valores diferentes dos usados na construção e validação do modelo), apresente resultados insatisfatórios. Este efeito é chamado de *Bias*. À medida que se aumenta a precisão do modelo para um conjunto de dados específico, perde-se a precisão para outros conjuntos.

6.3 Treinar o modelo de AM

Após toda a fase de pré-processamento, o conjunto de dados está preparado para ser induzido pelo modelo de AM, ou seja, aplicar à amostra de dados os algoritmos Árvore de Decisão, Regressão Logística e Redes Neurais Artificias, conforme exposto na Seção 4, primeiramente com os atributos selecionados com a abordagem filtro e depois com os atributos que foram selecionados após a abordagem *wrapper* para cada um dos modelos.

6.4 Avaliar o modelo de AM

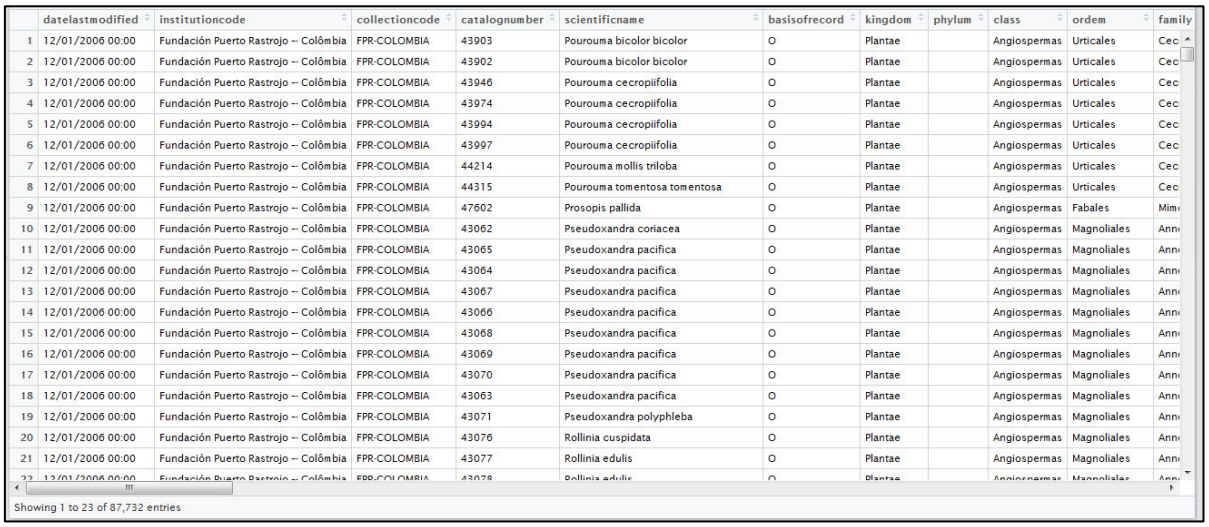
A avaliação dos modelos se dá conforme a taxa de acerto que o modelo aprendeu no conjunto de dados de treinamento que é submetido ao conjunto de teste e validação, sendo ilustrado no gráfico ROC, conforme abordado na Seção 4.4.

6.5 Otimização do modelo

A tarefa de AM é um processo interativo que sempre buscar uma taxa de acerto o mais próximo de 100%, mais é quase que improvável conseguir uma taxa de acerto tão alto, podendo esta taxa ser resultado de *overfitting*, essa otimização ocorre ajustando os parâmetros dos modelos de AM, por exemplo, diminuindo o tamanho da Árvore de Decisão, ou acrescentando camadas intermediárias nas redes neurais artificias.

7 RESULTADOS

O primeiro passo foi a importação da base de dados do INCT – Herbário Virtual da Flora e dos Fungos, para o IDE RStudio para ter o desenvolvimento para a implementação dos modelos de AM. No IDE RStudio a base de dados ficou da seguinte forma como mostrado na Figura 10.



	datelastmodified	institutioncode	collectioncode	catalognumber	scientificname	basisofrecord	kingdom	phylum	class	ordem	family
1	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43903	Pourouma bicolor bicolor	O	Plantae		Angiospermas	Urticales	Cec
2	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43902	Pourouma bicolor bicolor	O	Plantae		Angiospermas	Urticales	Cec
3	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43946	Pourouma cecropiifolia	O	Plantae		Angiospermas	Urticales	Cec
4	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43974	Pourouma cecropiifolia	O	Plantae		Angiospermas	Urticales	Cec
5	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43994	Pourouma cecropiifolia	O	Plantae		Angiospermas	Urticales	Cec
6	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43997	Pourouma cecropiifolia	O	Plantae		Angiospermas	Urticales	Cec
7	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	44214	Pourouma mollis triloba	O	Plantae		Angiospermas	Urticales	Cec
8	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	44315	Pourouma tomentosa tomentosa	O	Plantae		Angiospermas	Urticales	Cec
9	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	47602	Prosopis pallida	O	Plantae		Angiospermas	Fabales	Mim
10	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43062	Pseudoxandra coriacea	O	Plantae		Angiospermas	Magnoliales	Ann
11	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43065	Pseudoxandra pacifica	O	Plantae		Angiospermas	Magnoliales	Ann
12	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43064	Pseudoxandra pacifica	O	Plantae		Angiospermas	Magnoliales	Ann
13	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43067	Pseudoxandra pacifica	O	Plantae		Angiospermas	Magnoliales	Ann
14	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43066	Pseudoxandra pacifica	O	Plantae		Angiospermas	Magnoliales	Ann
15	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43068	Pseudoxandra pacifica	O	Plantae		Angiospermas	Magnoliales	Ann
16	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43069	Pseudoxandra pacifica	O	Plantae		Angiospermas	Magnoliales	Ann
17	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43070	Pseudoxandra pacifica	O	Plantae		Angiospermas	Magnoliales	Ann
18	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43063	Pseudoxandra pacifica	O	Plantae		Angiospermas	Magnoliales	Ann
19	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43071	Pseudoxandra polyphleba	O	Plantae		Angiospermas	Magnoliales	Ann
20	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43076	Rollinia cuspidata	O	Plantae		Angiospermas	Magnoliales	Ann
21	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43077	Rollinia edulis	O	Plantae		Angiospermas	Magnoliales	Ann
22	12/01/2006 00:00	Fundación Puerto Rastrojo -- Colômbia	FPR-COLOMBIA	43078	Rollinia edulis	O	Plantae		Angiospermas	Magnoliales	Ann

Showing 1 to 23 of 87,732 entries

Figura 10. Visualização da base de dados INCT – Herbário Virtual da Flora e dos Fungos

Após a verificação se o RStudio tinha conseguido importar a base de dados com sucesso, seus 87.732 registros e o 51 atributos, foi feito a fase de Pré-Processamento, como segue:

7.1 Pré-processamento

Com o pré-processamento verificou-se os atributos que estão com valores ausentes que em linguagem R é representado por NA (*Not Available*), ou seja, os campos dos atributos por algum motivo não foram fornecidos, e que nesta fase são excluídos da base de dados e que são mostrados no Quadro 5.

Nº	Atributo
27	julianday
28	timeofday
39	boundingbox
42	minimumdepth
43	maximumdepth
44	sex
48	relationshiptype

Quadro 5. Atributos com valores NA excluídos da base de dados

A análise de variáveis qualitativas é fundamentada na noção de frequência absoluta que é o número de observações no conjunto de dados com uma dada categoria do atributo de interesse. Considerando que * conjunto de dados tem **n** observações ***x** é um atributo qualitativo cujo valores podem assumir $\mathbf{m_x}$ categorias: $\mathbf{c_1, c_2, ..., c_{mx},}$ onde:

Frequência absoluta de $\mathbf{c_j} \rightarrow \#(x == c_j)$, j = 1, 2, ..., m_x, sendo que a notação $\#(x == c_j)$ indica o número de observações em que o atributo categórico **x** apresenta a categoria $\mathbf{c_j}$.

Utilizando comandos em R que retornam a tabela absoluta para verificar a frequência dos valores dos atributos foi possível notar também uma grande quantidade de valores NA em outros atributos, que para não prejudicar o poder de predição do sistema de AM, esses também foram excluídos da base, os atributos com frequência alta de valores NA são mostrados na Figura 11.

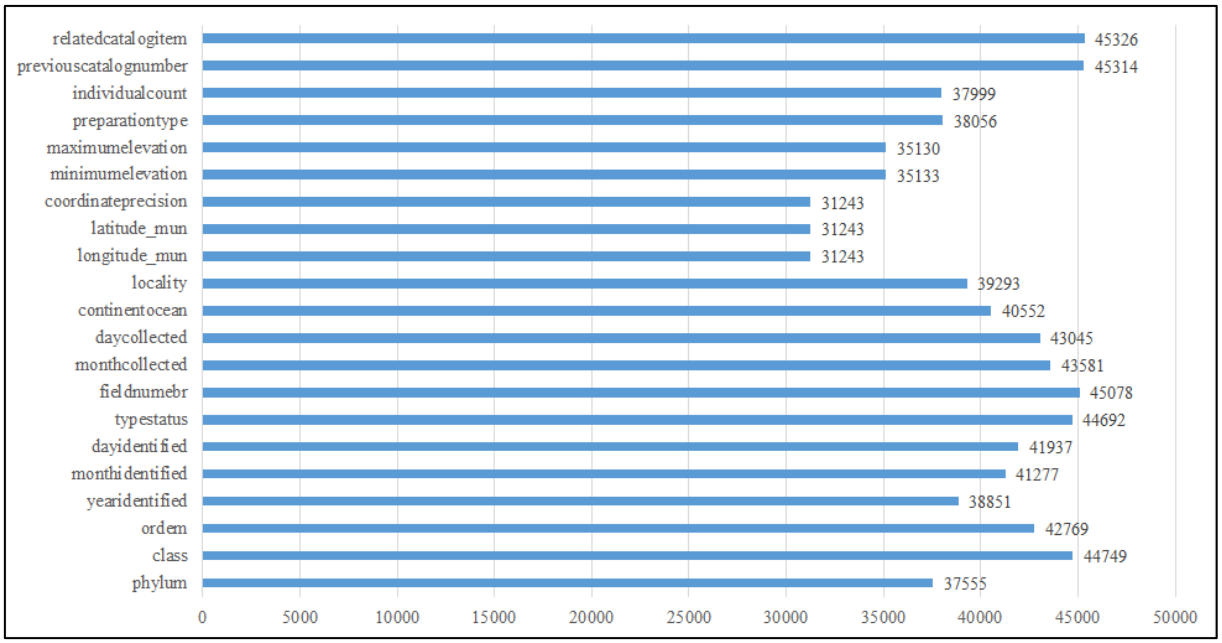


Figura 11. Gráfico de valores altos de frequência de valores NA

Para se ter um poder de precisão mais alto é necessário que os atributos tenham um poder de generalização em seus valores para que possam predizer melhor a classe, com isso foram observados alguns atributos com valores únicos para cada registro, conforme Quadro 6.

Nº	Atributo
1	datelastmodified
4	catalognumber
21	collectornumber
50	notes
51	barcode

Quadro 6. Atributos com valores únicos

Com esta etapa agora tem 18 atributos como mostra o Quadro 7:

Nº	Atributo
2	institutioncode
3	collectioncode
5	scientificname
6	basisofrecord
7	kingdom
11	family
12	genus
13	species
14	subspecies
15	scientificnameauthor
16	idenfifiedby
23	collector
24	yearcollected
30	country
31	stateprovince
32	county
34	longitude
35	latitude

Quadro 7. Atributos Selecionados com o Filtro

Desses atributos, foi escolhido o atributo *family* para ser o atributo alvo, sendo que neste atributo foi feito um filtro para reduzir a quantidade de registros e ter valores relativos e significativos que possam ser preditos pelos outros atributos já selecionados anteriormente. O filtro foi feito, pois este atributo possuía 736 categorias diferentes, com o filtro aplicado selecionando apenas as classes que tinham acima de mil registros por categoria, com isso foram selecionadas 20, conforme demonstrada na Figura 12.

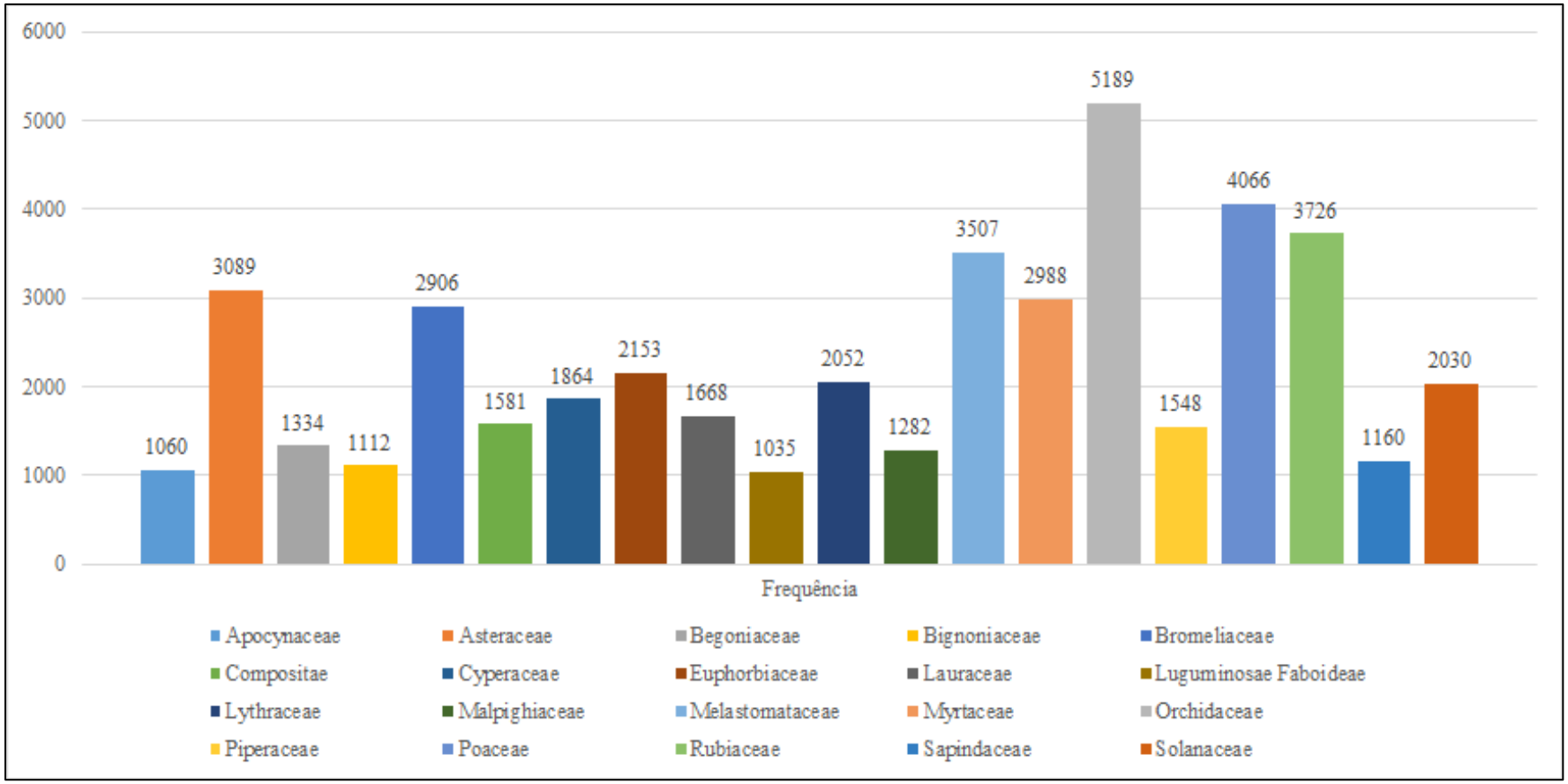


Figura 12. Frequência de ocorrências de cada classe do atributo family

Com os 18 atributos selecionados tem-se os 45.317 registros na base de dados, e agora poderá ser aplicado a partir destes metadados os modelos de AM e as outras abordagens de SA avaliando qual será o melhor modelo preditivo para a classe *family* deste conjunto de dados.

Fazendo uma nova filtragem nesses atributos, observou-se que o atributo *kingdon* tinha apenas o valor *plantae*. Com os atributos restantes pode-se notar ainda a quantidade alta de frequência de valores NA, como vemos na Figura 13. Sendo que quase todos os valores do atributo *subspecies* continham este valor, já atributo *scientificnameauthor*, com o alto índice de valores NA ainda tinha uma grande quantidade de valores únicos que não seriam relevantes para o modelo fazer a generalização, e o a tributo *basisofrecord*, mesmo com este alto valor 26.935 de NA a maioria dos demais valores eram S.

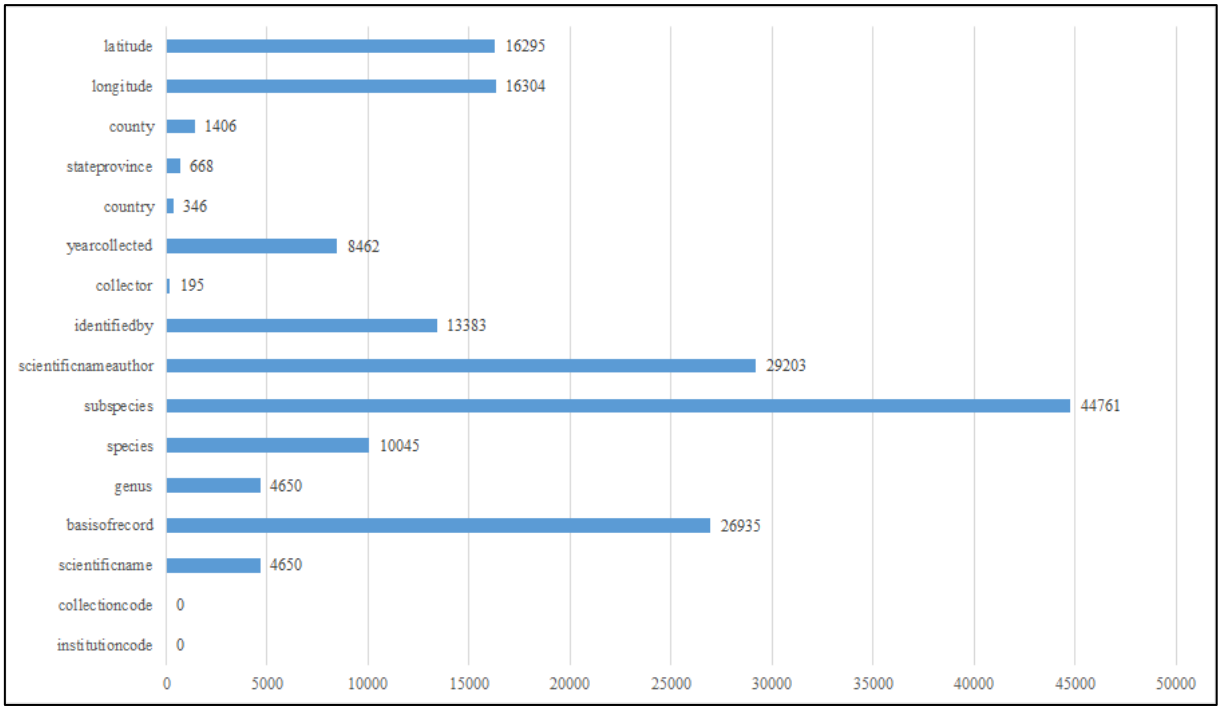


Figura 13. Valores altos de frequência de valores NA

Os atributos *collector* e *identifiedby*, continham muitos valores únicos o que prejudicaria o modelo na hora de fazer a predição da classe. E o atributo *scientificname* era a concatenação dos atributos *genus* e *species* e também foi removido da base, chegando com isso aos metadados dos quais iriam ser absorvidos pelos modelos de classificação e fazendo a limpeza de todos os valores NA da base de dados.

Observando a classe preditora o atributo family após este segundo filtro, retirando todos os valores NA da base observa-se que não há um balanceamento dos valores como mostra a Figura 14, sendo assim, este atributo passa a ficar apenas valores no intervalo entre 300 a 2.000, constando assim o atributo *family* com os valores mostrados na Figura 15.

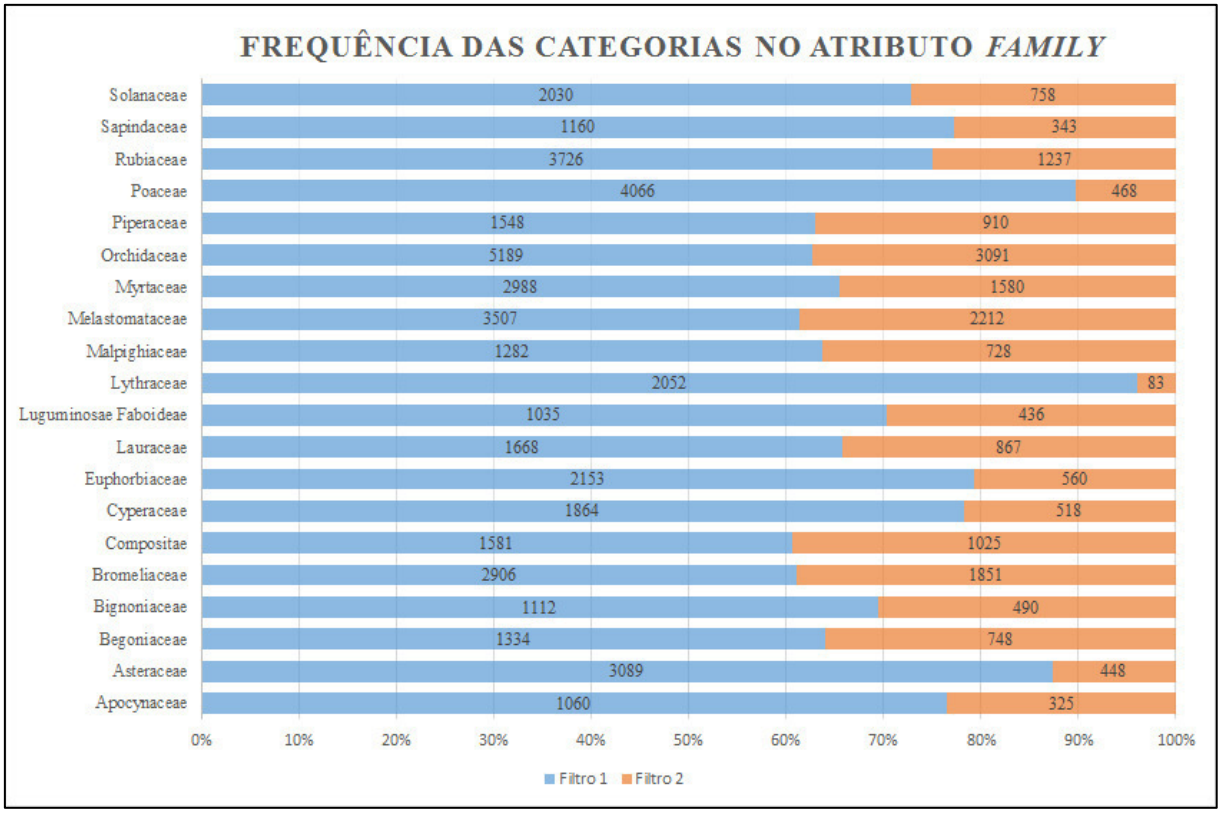


Figura 14. Frequência dos valores com os dois filtros no atributo family

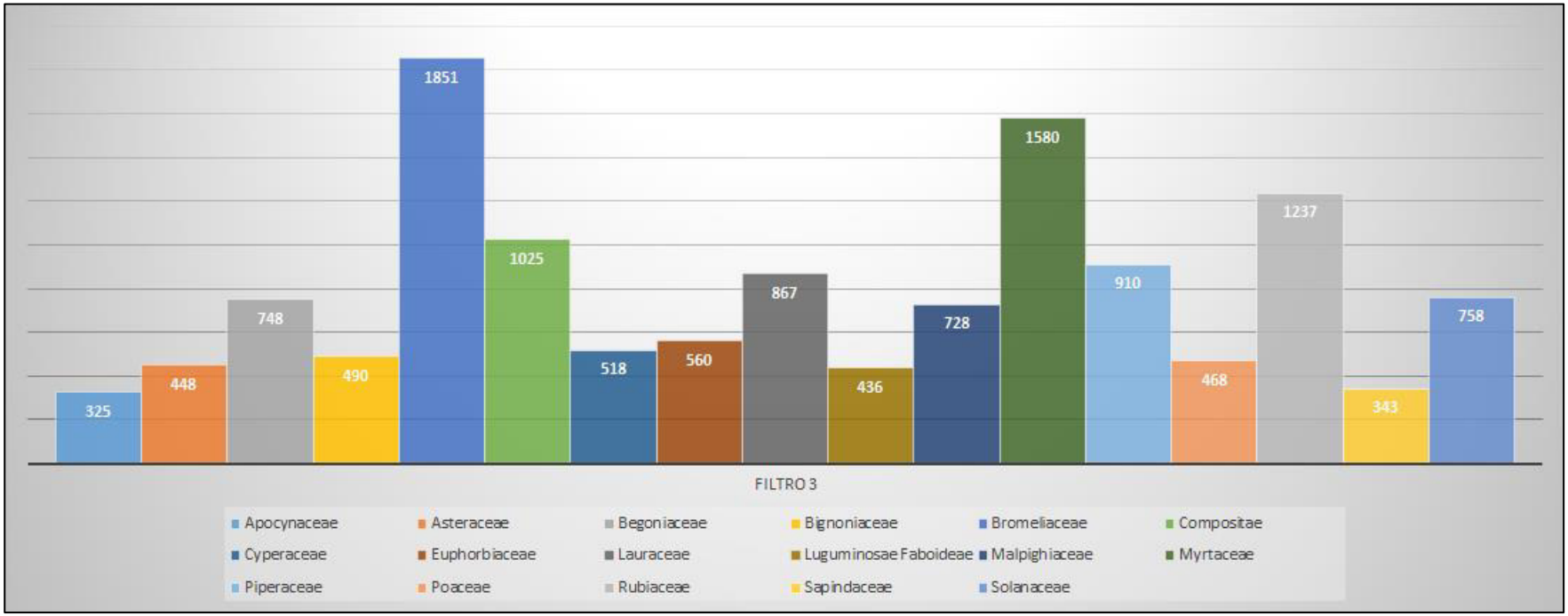


Figura 15. Frequência de valores com o terceiro filtro no atributo family

Com esse filtro a base de dados ficou com os seguintes atributos para a aplicação dos modelos preditivos como mostra o Quadro 8:

Nº	Atributo
2	institutioncode
3	collectioncode
11	family
12	genus
13	species
24	yearcollected
30	country
31	stateprovince
32	county
34	longitude
35	latitude

Quadro 8. Atributos Selecionados para a aplicação dos modelos preditivos

Como os algoritmos de AM Árvore de Decisão, Regressão Logística e Rede Neural Artificial, utilizam atributos com valores do tipo número e/ou categórico, nesta etapa foi feita a transformação dos dados, agrupando os valores iguais e atribuindo a estes valores numéricos. E o atributo *family* continuou como texto, mas foi convertido para valores do tipo categóricos, ou seja, foram criados níveis para cada uma das dezessete categorias do atributo alvo *family*.

family	institutioncode	collectioncode	genus	species	yearcollected	country	stateprovince	county	longitude	latitude
Apocynaceae	MBML	MBML-HERBARIO	Tabernaemontana	hystrix	2005	Brasil	Espirito Santo	Santa Teresa	-40.555833	-19.935
Apocynaceae	MBML	MBML-HERBARIO	Tabernaemontana	laeta	2001	Brasil	Espirito Santo	Santa Teresa	-40.798056	-19.899167
Apocynaceae	MBML	MBML-HERBARIO	Tabernaemontana	laeta	1998	Brasil	Espirito Santo	Santa Teresa	-40.654722	-19.823889
Apocynaceae	MBML	MBML-HERBARIO	Temnadenia	odorifera	1996	Brasil	Espirito Santo	Santa Teresa	-40.824722	-19.888889
Apocynaceae	MBML	MBML-HERBARIO	Mandevilla	scabra	2003	Brasil	Bahia	Santa Terezinha	-39.520556	-12.946389
Apocynaceae	MBML	MBML-HERBARIO	Rauvolfia	bahiensis	2000	Brasil	Bahia	Santa Terezinha	-39.520556	-12.946389
Apocynaceae	IAP	PACA-AGP	Oxypetalum	caeruleum	2006	Brasil	Rio Grande do Sul	Santana da Boa Vista	-53.633611	-30.931389
Apocynaceae	UNESC	CRI	Orthosia	dusenii	1960	Brasil	Santa Catarina	São Francisco do Sul	-48.638333	-26.243889
Apocynaceae	MBML	MBML-HERBARIO	Tabernaemontana	laeta	2009	Brasil	Espirito Santo	São Gabriel da Palha	-40.5155	-18.948139
Apocynaceae	CPQBA	CPMA	Mandevilla	velutina	1979	Brasil	Minas Gerais	São João Del Rei	-44.261667	-21.135556
Apocynaceae	UNESC	CRI	Oxypetalum	erectum	1965	Brasil	Santa Catarina	São Joaquim	-49.931944	-28.294444
Apocynaceae	UNESC	CRI	Fischeria	stellata	1960	Brasil	Santa Catarina	São José	-48.627778	-27.615833
Apocynaceae	UEFS	HUEFS	Allamanda	blanchetii	2010	Brasil	Paraíba	São José dos Cordeiros	-36.898333	-7.470556
Apocynaceae	UEFS	HUEFS	Allamanda	blanchetii	2010	Brasil	Paraíba	São José dos Cordeiros	-36.89	-7.476389
Apocynaceae	UEFS	HUEFS	Aspidosperma	pyrifolium	2009	Brasil	Paraíba	São José dos Cordeiros	-35.898333	-7.470556
Apocynaceae	UEFS	HUEFS	Mandevilla	tenuiflora	2010	Brasil	Paraíba	São José dos Cordeiros	-36.898333	-7.470556
Apocynaceae	UEFS	HUEFS	Mandevilla	tenuiflora	2010	Brasil	Paraíba	São José dos Cordeiros	-35.898333	-7.470556
Apocynaceae	UEFS	HUEFS	Marsdenia	megalantha	2010	Brasil	Paraíba	São José dos Cordeiros	-36.89	-7.476389
Apocynaceae	UEFS	HUEFS	Marsdenia	megalantha	2010	Brasil	Paraíba	São José dos Cordeiros	-36.89	-7.476389
Apocynaceae	MBML	MBML-HERBARIO	Himatanthus	phagedaenicus	1972	Brasil	Espirito Santo	São Mateus	-39.755833	-18.7375
Apocynaceae	MBML	MBML-HERBARIO	Allamanda	laevis	2003	Brasil	Espirito Santo	São Roque do Canaã	-40.739444	-19.760833
Apocynaceae	MBML	MBML-HERBARIO	Allamanda	laevis	2004	Brasil	Espirito Santo	São Roque do Canaã	-40.739444	-19.760833
Apocynaceae	MBML	MBML-HERBARIO	Allamanda	laevis	2007	Brasil	Espirito Santo	São Roque do Canaã	-40.739444	-19.760833
Apocynaceae	MBML	MBML-HERBARIO	Allamanda	laevis	2008	Brasil	Espirito Santo	São Roque do Canaã	-40.75425	-19.772056

Figura 16. Base de dados com valores antes transformação dos dados

```
Classes ‘data.table’ and 'data.frame':  14160 obs. of  11 variables:
 $ family         : chr  "Apocynaceae" "Apocynaceae" "Apocynaceae" "Apocynaceae" ...
 $ institutioncode: int  27 27 27 27 27 27 27 27 27 27 ...
 $ collectioncode : int  64 64 64 64 64 64 64 64 64 64 ...
 $ yearcollected  : int  2008 2006 2004 2007 2006 2006 2008 2008 2006 2006 ...
 $ longitude      : num  -41 -40.7 -40.7 -40.7 -40.6 ...
 $ latitude       : num  -18.6 -19 -19 -18.9 -19 ...
 $ genus          : int  706 47 237 498 706 706 706 706 706 706 ...
 $ species        : int  1659 2079 3530 2072 1431 1431 1431 1431 1659 1659 ...
 $ country        : int  9 9 9 9 9 9 9 9 9 9 ...
 $ stateprovince  : int  58 58 58 58 58 58 58 58 58 58 ...
 $ county         : int  19 28 28 28 28 28 28 28 28 28 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

Figura 17. Base de dados com valores transformados.

Pode ser observado que os valores numéricos não estão normalizados, ou seja, não estão na mesma escala, como por exemplo os atributos *institutioncode*, *yearcollected*, *longitude* e *genus* tem valores respectivamente 27, 2008, -41, 706, no algoritmo de Regressão Logística é preciso que os valores estejam na mesma escala, e o atributo *family* ainda está do tipo caractere e se faz necessário à sua transformação para categórico, então convertendo os valores para a mesma escala a base fica desta forma.

family	institutioncode	collectioncode	yearcollected	longitude	latitude	genus	species	country	stateprovince	county
Apocynaceae	-0.34391426	0.3353678	0.7661466733	0.008406148	0.008411614	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.10874156
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406949	0.008409891	-1.58637937	-0.10406503	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.3828838706	0.008406754	0.008409848	-1.07648972	1.02118305	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.6703309726	0.008406779	0.008410193	-0.37606237	-0.10949352	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406997	0.008409676	0.18213260	-0.60658795	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406609	0.008410020	0.18213260	-0.60658795	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.7661466733	0.008406876	0.008409848	0.18213260	-0.60658795	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.7661466733	0.008406682	0.008409848	0.18213260	-0.60658795	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406997	0.008409676	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406609	0.008410020	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406633	0.008410020	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.6703309726	0.008406949	0.008409891	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.6703309726	0.008406876	0.008409805	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.6703309726	0.008406609	0.008410020	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.6703309726	0.008406706	0.008409891	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.6703309726	0.008406949	0.008409891	0.18213260	-0.42977433	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406779	0.008410193	0.82352010	-1.24792507	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406779	0.008410193	0.82352010	-1.05094850	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406609	0.008409977	0.82352010	-1.05094850	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406609	0.008409977	0.82352010	-1.05094850	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406779	0.008410193	0.82352010	-1.05094850	-0.002494437	-0.3906318	-2.09442551
Apocynaceae	-0.34391426	0.3353678	0.5745152719	0.008406949	0.008410193	0.82352010	-1.05094850	-0.002494437	-0.3906318	-2.09442551

Figura 18. Base de dados após a transformação dos dados.

Com essa última fase do pré-processamento temos a base de dados, limpa, sem valores NA, e agora na mesma escala, ficando com 14.160 instâncias e 11 atributos chegando ao fim desta etapa e pronto para próxima etapa de organizar os dados para aplicar os modelos de AM.

7.2 Organização dos dados

Dividindo a base de dados em 60%, em dados de treino e o restante para os dados de teste ficando com 8.496 instâncias para os dados de treino e 5.664 instâncias para os dados de teste, conforme o Figura 19, esta divisão foi a selecionada devido a ter tido o melhor desempenho dos modelos submetidos. Sendo que os dados de teste é preciso separar o atributo classe dos demais atributos.

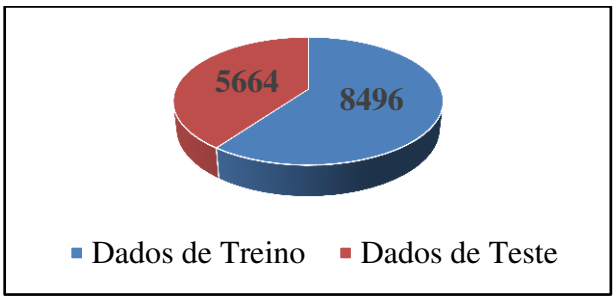


Figura 19. Organização dos dados

7.3 Aplicando os modelos de AM

Nesta fase são aplicados os modelos de AM utilizando a base de dados de treino, para que o algoritmo possa extrair conhecimento e assim criar uma função matemática com base nos valores dos atributos relacionando-os a sua respectiva classe. Com todos os dez atributos preditores e mais o atributo alvo na base de dados de treino, que foram selecionados com a abordagem de SA filtro na fase de pré-processamento e depois aplicando o modelo utilizando a abordagem *wrapper*.

7.3.1 Árvore de Decisão

Induzindo o modelo de Árvore de Decisão à base de dados de treino com todos os atributos e para fazer a avalição do modelo, após a fase treinamento é aplicado à base de dados de teste, mas desta vez sem o atributo alvo para justamente avaliar o quanto o modelo conseguiu aprender e neste caso foi alcançada uma acurácia aceitável de 70,75%, conforme mostra a Figura 20. Assim pode ser feito o plote do gráfico ROC, onde a UAC de 0,5, ou seja, gerou um classificador aleatório, mostrado na Figura 21.

```
                Accuracy : 0.7075
                  95% CI : (0.6602, 0.7517)
    No Information Rate : 0.7075
    P-Value [Acc > NIR] : 0.5249

                   Kappa : 0
 Mcnemar's Test P-Value : <2e-16

              Sensitivity : 1.0000
              Specificity : 0.0000
           Pos Pred Value : 0.7075
           Neg Pred Value :    NaN
               Prevalence : 0.7075
           Detection Rate : 0.7075
     Detection Prevalence : 1.0000
        Balanced Accuracy : 0.5000
```

Figura 20. Resultado do modelo de Árvore de Decisão com os atributos da abordagem filtro

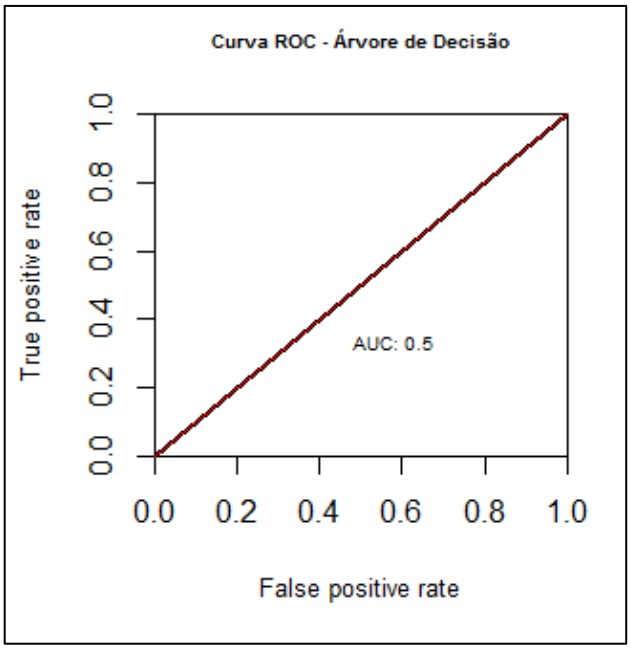


Figura 21. Gráfico ROC do modelo de Árvore de Decisão com abordagem filtro

Aplicando o mesmo modelo, mas desta vez, para fazer a abordagem *wrapper* selecionado os melhores subconjuntos de atributos como mostra a Figura 22, os cinco atributos com maior grau de importância para este modelo foram: *institutioncode, county, stateprovince, genus* e *yearcollected*.

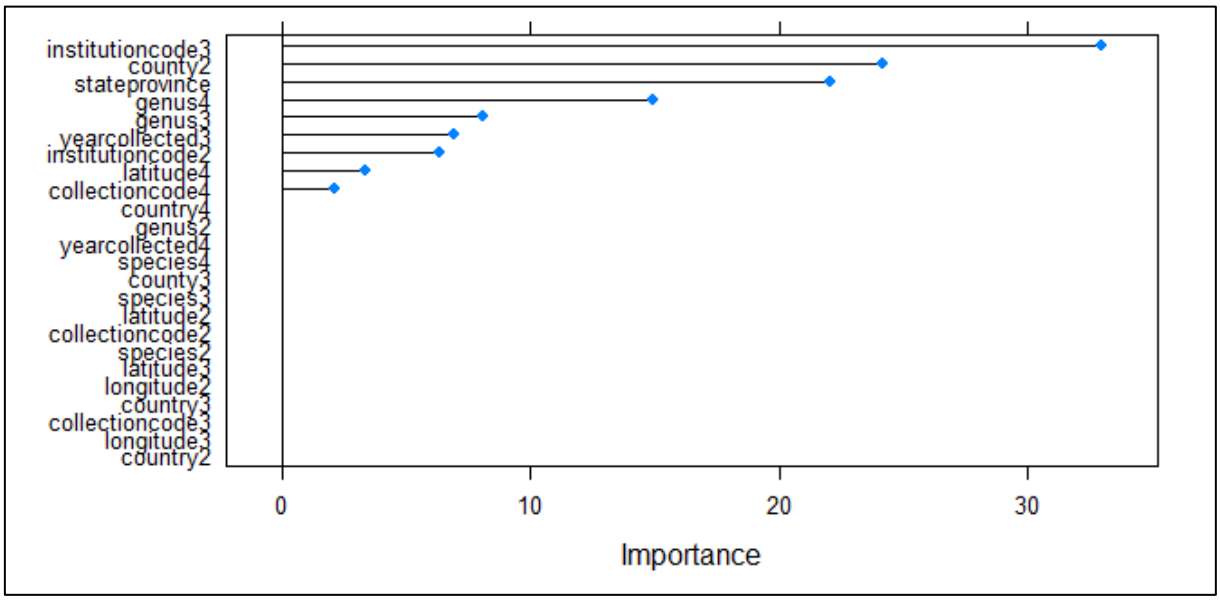


Figura 22. Grau de importância dos atributos do modelo de Árvore de Decisão com a abordagem *wrapper*

Agora aplicando o modelo apenas com os atributos mais relevantes a partir da abordagem *wrapper* observou-se que sua acurácia de 65,25% diminuiu Figura 23, mas por outro lado este modelo conseguiu um melhor nível de predição com AUC de 0.69 conforme mostra o gráfico ROC na Figura 24.

```
                Accuracy : 0.6525
                  95% CI : (0.6036, 0.6991)
    No Information Rate : 0.7075
    P-Value [Acc > NIR] : 0.9926

                   Kappa : 0.3107
 Mcnemar's Test P-Value : 2.999e-13

            Sensitivity : 0.6007
            Specificity : 0.7778
         Pos Pred Value : 0.8673
         Neg Pred Value : 0.4461
             Prevalence : 0.7075
         Detection Rate : 0.4250
   Detection Prevalence : 0.4900
      Balanced Accuracy : 0.6892
```

Figura 23. Resultado do modelo de Árvore de Decisão com os atributos da abordagem *wrapper*

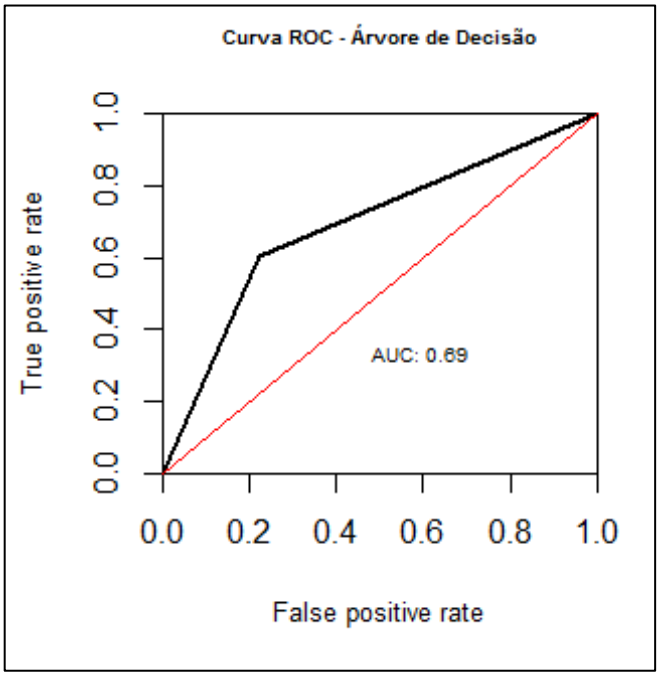


Figura 24. Gráfico ROC do modelo de Árvore de Decisão com abordagem wrapper

7.3.2 Rede Neural Artificial

Induzindo o modelo de Rede Neural Artificial à base de dados de treino com todos os atributos e para fazer a avalição do modelo, após a fase treinamento é aplicado à base de dados de teste, mas desta vez sem o atributo alvo para justamente avaliar o quanto o modelo conseguiu aprender e neste caso foi alcançada uma acurácia aceitável de 78%, conforme mostra a Figura 25. Assim pode ser feito o plote do gráfico ROC, com o classificador atingindo AUC de 0.78, com poder de classificação aceitável, conforme mostrado na Figura 26.

```
               Accuracy : 0.78
                 95% CI : (0.7362, 0.8196)
    No Information Rate : 0.7075
    P-Value [Acc > NIR] : 0.0006615

                  Kappa : 0.4314
 Mcnemar's Test P-Value : 0.0076986

            Sensitivity : 0.8905
            Specificity : 0.5128
         Pos Pred Value : 0.8155
         Neg Pred Value : 0.6593
             Prevalence : 0.7075
         Detection Rate : 0.6300
   Detection Prevalence : 0.7725
      Balanced Accuracy : 0.7016
```

Figura 25. Resultado do modelo de Rede Neural Artificial com os atributos da abordagem filtro

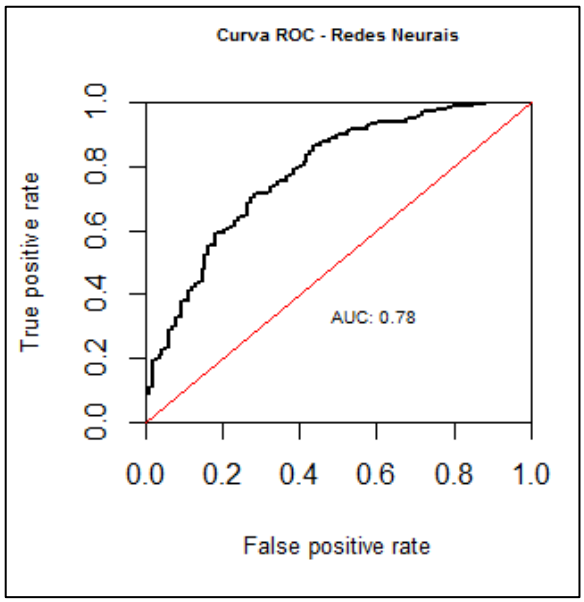


Figura 26. Gráfico ROC do modelo de Rede Neural Artificial com abordagem filtro

Aplicando o mesmo modelo para fazer a abordagem *wrapper* selecionado os melhores subconjuntos de atributos, os cinco atributos com maior grau de importância para este modelo foram: *institutioncode, genus, latitude, collectioncode, longitude*, como mostra a Figura 27.

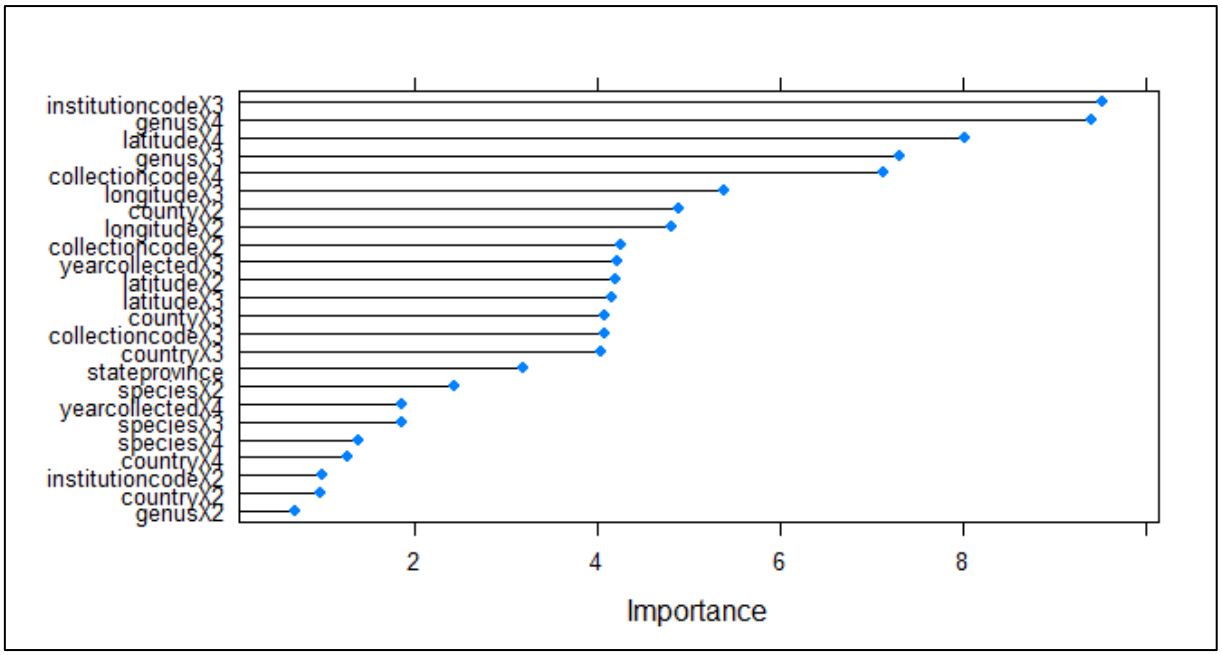


Figura 27. Grau de importância dos atributos do modelo de Rede Neural Artificial com a abordagem *wrapper*

Agora aplicando o modelo apenas com os atributos mais relevantes a partir da abordagem *wrapper*, observou-se que sua acurácia diminuiu para 75,5% (Figura 28), assim como o poder de predição conforme mostra o gráfico ROC na Figura 29, com AUC de 0.75.

```
                Accuracy : 0.755
                  95% CI : (0.7098, 0.7964)
    No Information Rate : 0.7075
    P-Value [Acc > NIR] : 0.01971

                   Kappa : 0.3309
 Mcnemar's Test P-Value : 5.476e-06

            Sensitivity : 0.9081
            Specificity : 0.3846
         Pos Pred Value : 0.7812
         Neg Pred Value : 0.6338
             Prevalence : 0.7075
         Detection Rate : 0.6425
   Detection Prevalence : 0.8225
      Balanced Accuracy : 0.6464
```

Figura 28. Resultado do modelo de Rede Neural Artificial com os atributos da abordagem *wrapper*

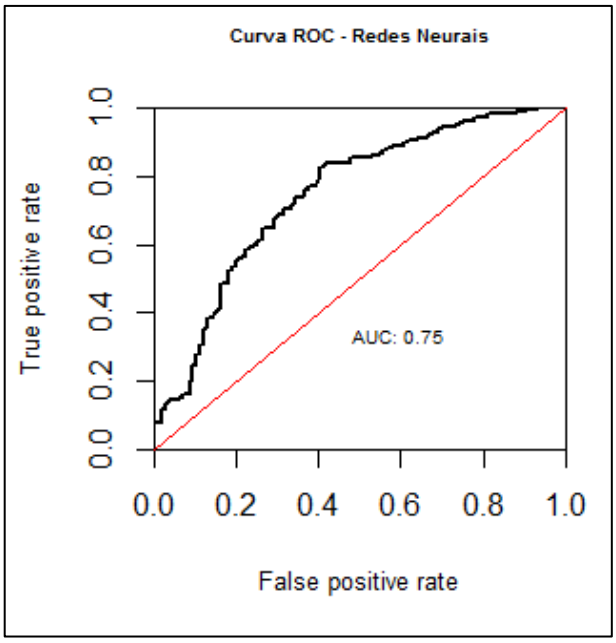


Figura 29. Gráfico ROC do modelo de Rede Neural Artificial com abordagem *wrapper*

7.3.3 Regressão Logística

Aplicando agora o terceiro modelo, o de Regressão Logística, à base de dados de treino com todos os atributos e para fazer a avalição do modelo, após a fase treinamento é aplicado à base de dados de teste, mas desta vez sem o atributo alvo para justamente avaliar o quanto o modelo conseguiu aprender e neste caso foi alcançada uma acurácia aceitável de 77,25%, conforme mostra a Figura 30. Assim pode ser gerado o gráfico ROC, mostrado na Figura 31, com AUC de 0.76.

```
               Accuracy : 0.7725
                 95% CI : (0.7282, 0.8127)
    No Information Rate : 0.7075
    P-Value [Acc > NIR] : 0.0020997

                  Kappa : 0.4008
 Mcnemar's Test P-Value : 0.0007951

            Sensitivity : 0.8975
            Specificity : 0.4701
         Pos Pred Value : 0.8038
         Neg Pred Value : 0.6548
             Prevalence : 0.7075
         Detection Rate : 0.6350
   Detection Prevalence : 0.7900
      Balanced Accuracy : 0.6838
```

Figura 30. Resultado do modelo de Regressão Logística com os atributos da abordagem filtro

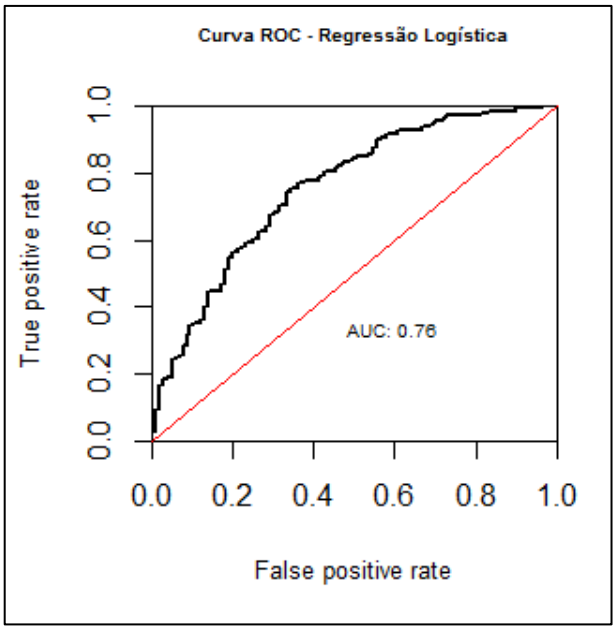


Figura 31. Gráfico ROC do modelo de Regressão Logística com abordagem filtro

Aplicando o mesmo modelo para fazer a abordagem *wrapper* selecionado os melhores subconjuntos de atributos como mostra a Figura 32, os atributos com maior grau de importância para este modelo foram: *institutioncode*, *genus*, *stateprovince*, *latitude*, *county*.

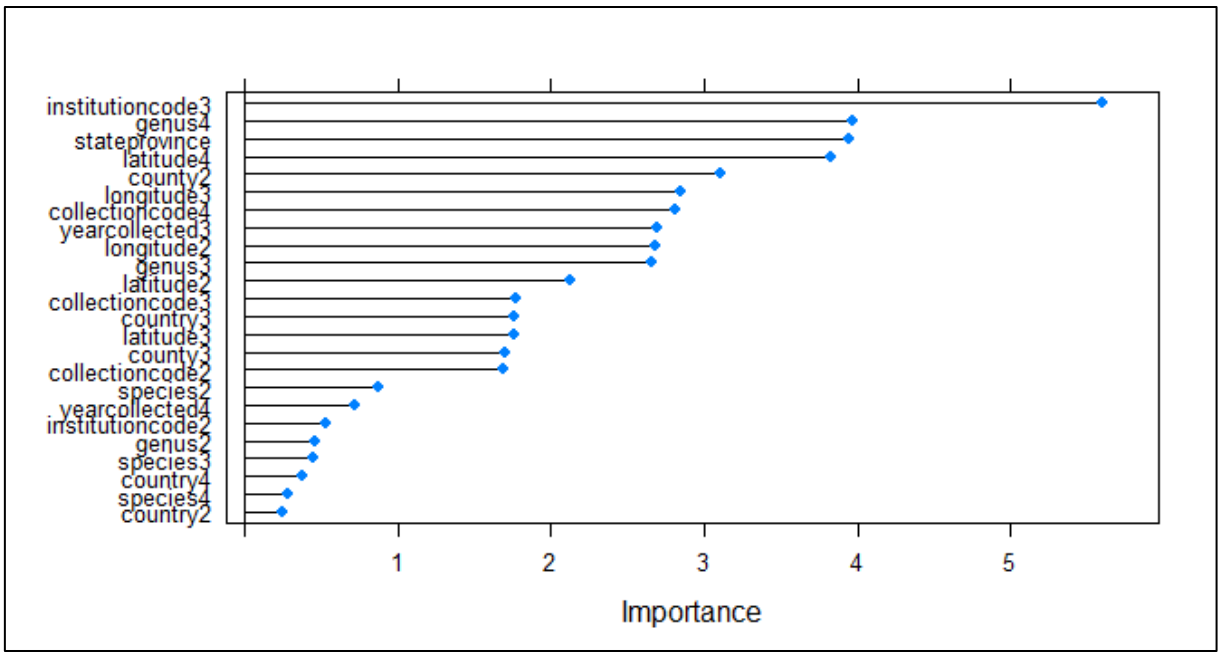


Figura 32. Grau de importância dos atributos do modelo de Regressão Logística com a abordagem *wrapper*

Aplicando o modelo apenas com os atributos mais relevantes a partir da abordagem *wrapper* observou-se que sua acurácia diminuiu para 76,25% (Figura 33), mas por outro lado este modelo conseguiu um melhor nível de predição conforme mostra o gráfico ROC com AUC de 0.78 na Figura 34.

```
               Accuracy : 0.7625
                 95% CI : (0.7177, 0.8034)
    No Information Rate : 0.7075
    P-Value [Acc > NIR] : 0.008144

                  Kappa : 0.3383
 Mcnemar's Test P-Value : 9.55e-08

            Sensitivity : 0.9258
            Specificity : 0.3675
         Pos Pred Value : 0.7798
         Neg Pred Value : 0.6719
             Prevalence : 0.7075
         Detection Rate : 0.6550
   Detection Prevalence : 0.8400
      Balanced Accuracy : 0.6467
```

Figura 33. Resultado do modelo de Regressão Logística com os atributos da abordagem *wrapper*

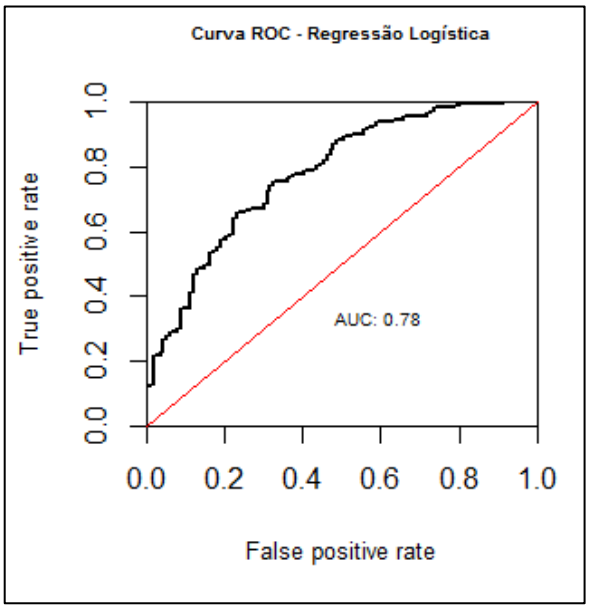


Figura 34. Gráfico ROC do modelo de Regressão Logística com abordagem *wrapper*

7.4 Considerações Finais

Portanto, a avaliação de desempenho dos modelos de AM, conforme a acurácia, os atributos selecionados com a abordagem filtro obtiveram o melhor desempenho, em relação aos atributos selecionados através da abordagem *wrapper*.

Já com a avaliação de desempenho com a área sobre a curva ROC, apenas o modelo Rede Neural Artificial obteve um melhor desempenho com a abordagem filtro em relação aos outros dois modelos, sendo que o modelo de Árvore de Decisão com os atributos: *institutioncode*, *county*, *stateprovince*, *genus* e *yearcollected*, e o modelo de Regressão Logística com os atributos: *institutioncode*, *genus*, *stateprovince*, *latitude*, *county*, foram os selecionados com a abordagem *wrapper* e obtiveram melhor desempenho em relação à abordagem filtro.

Como pode ser observado na Tabela 2, o modelo de Árvore de Decisão com a abordagem filtro obteve uma acurácia de 70,25% e já a AUC 0,5, podendo concluir que o modelo provavelmente sofreu de *overfitting*, ou seja, aprendeu demais com o modelo e não conseguiu prever corretamente na fase de teste as classes.

	ACURÁCIA		AUC	
MODELOS	FILTRO	WRAPPER	FILTRO	WRAPPER
Árvore de Decisão	**70,75%**	65,25%	0,5	**0,69**
Rede Neural Artificial	**78%**	75,5%	**0,78**	0,75
Regressão Logística	**77,25%**	76,25%	0,76	**0,78**

Tabela 2. Resumo dos resultados dos modelos de classificação, conforme as abordagens e a forma de avaliação.

8 CONCLUSÃO

Conforme mencionado na introdução, o objetivo deste trabalho foi aplicar os processos de seleção de atributos na base de dados do INCT – Herbário Virtual da Flora e dos Fungos utilizando técnicas de AM, analisando os processos de seleção de subconjuntos de atributos com as abordagens filtro, *wrapper* e embutido, induzindo os modelos de classificação: Árvore de Decisão, Rede Neural Artificial e Regressão Logística, tanto com os subconjuntos da abordagem filtro quanto da abordagem *wrapper*, sendo que, estes modelos já possuem a abordagem embutida na própria estrutura do algoritmo selecionando internamente os melhores atributos e depois foi avaliado o desempenho desses modelos através da matriz de confusão para se extrair a acurácia do modelo e assim gerar o gráfico ROC.

Partindo desse ponto, o trabalho conseguiu cumprir o que foi proposto e mostrou que a combinação das técnicas de seleção de atributos é vantajosa, já que elas têm funções complementares entre si, obtendo-se modelos coesos e com boa capacidade de generalização.

Do ponto de vista teórico, vale destacar que durante o levantamento bibliográfico, não foi identificado nenhum outro trabalho de seleção de atributos utilizando base de dados de herbário, assim sendo esse trabalho traz uma contribuição para seu campo de pesquisa acadêmica ao mostrar que pode ser utilizada as bases de dados da rede *speciesLink* para a disseminação das informações dos recursos biológicos brasileiros.

Do ponto de vista prático, o trabalho demonstrou sua viabilidade técnica nas fases de pré-processamento até a modelagem utilizando a linguagem R com o IDE RStudio, uma das linguagens mais utilizadas atualmente na área de AM.

8.1 Sugestões para pesquisas futuras

Ao longo do trabalho, foi possível identificar algumas frentes de trabalho que podem ser abordadas no futuro.

Integrar outras bases: nesse trabalho foi utilizada a base de dados do Herbário Virtual da Flora e dos Fungos, sendo que na rede *speciesLink* existem diversas bases, e incorporar outras utilizando esta metodologia para fazer novos casos de uso seria de fundamental importância para ampliar os conhecimentos sobre a biodiversidade brasileira com conhecimentos da AM.

Modelos de Aprendizagem de Máquina: Desenvolver modelos de AM com outros algoritmos e avaliar o seu desempenho com as abordagens de seleção de subconjuntos de atributos, sendo que nesta pesquisa foram utilizados os modelos de Árvore de Decisão, Rede Neural Artificial e Regressão Logística.

Avaliação de desempenho dos modelos: nesta pesquisa foi utilizada a matriz de confusão e a medida de desempenho foi avaliada por meio da acurácia, com a possibilidade de fazer ajustes paramétricos nos modelos podendo alcançar resultados diferentes dos obtidos, e até mesmo ter novas métricas de avaliação, como confiança negativa, sensitividade, cobertura e suporte.

Classes desbalanceadas: outro tralho interessante é verificar o comportamento dessas abordagens de seleção de subconjuntos de atributos no problema de classes desbalanceadas, com o objetivo de ordenar os casos de classes minoritárias à frente das classes majoritárias.

REFERÊNCIAS

AHA, D. W.; KIBLER, D.; ALBERT, M. K. Instance-based learning algorithms. Machine learning, v. 6, n. 1, p. 37-66, 1991.

ALLISON, P. Logistic Regression Using SAS: Theory and Application. 1 ed. SAS Publishing, 1999.

BLUM, A. L.; LANGLEY, P. Selection of relevant features and examples in machine learning. Artificial Intelligence, p. 245-271, 2007.

BREVE, F. A.; ZHAO, L. Aprendizado de Máquina em Redes Complexas. In: VIII Best MSc Dissertation/PhD Thesis Contest in Artificial Intelligence-CTDIA, The Brazilian Conference on Intelligent System-BRACIS, 2012, Curitiba.

BREIMAN, L.; FRIEDMAN, J. H. Estimating optimal transformations for multiple regression and correlation. Journal of the American statistical Association, v. 80, n. 391, p. 580-598, 1985.

CANHOS, D. A. et al. The importance of biodiversity E-infrastructures for megadiverse countries. PLoS Biol, v. 13, n. 7, p. 202-204, 2015.

CLARK, P.; NIBLETT, T. The CN2 induction algorithm. Machine learning, v. 3, n. 4, p. 261-283, 1989.

CLARK, P.; BOSWELL, R. Rule induction with CN2: Some recent improvements. In: European Working Session on Learning. Springer Berlin Heidelberg, 1991. p. 151-163.

DIETTERICH, T. G. An experimental comparison of three methods for constructing esembles of decision trees: Bagging, boostins and randomization. Machine Learning, v. 40, p. 139-157, 2000.

DIETTERICH, T. G. Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation. USA, Oregon, v. 10, p. 1895-1923, 1998.

DUDA, R. O.; HART, P. E.; STORK, D. G. Pattern classification. 2000.

FACELLI, K. et al. Inteligência Artificial: Uma abordagem de aprendizado de máquina. Rio de Janeiro: LTC, 2011.

FLACH, P. A. e WU, S. Repairing concavities in ROC curves. In Kaelbling, L. P. e SAFFIOTTI, A., editors, IJCAI' 05: Proceeding of the Nineteenth International Joint Conference on Artificial Intelligence, p. 702-707. 2005

HAN, J., KAMBER, M., e PEI, J. Data Mining: Concepts and Techniques. 3 ed. Morgan Kaufmann, 2011.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. The Elements if Statistical Learning. Springer, 2001.

Herbário Alexandre Leal Costa (ALCB), *Herbário da Universidade Federal de Sergipe* (ASE), *Arizona State University Vascular Plant Herbarium* (ASU-Plants), *Herbarium Berolinense* (B), *Herbário Antônio Nonato Marques* (BAH), *Xiloteca Calvino Mainieri* (BCTw), *Herbário da Universidade Federal de Minas Gerais* (BHCB), *Herbário UFMG - Samambaias e Licófitas* (BHCB-SL), *Herbário do Jardim Botânico da Fundação Zoo-Botânica de Belo Horizonte* (BHZB), *Herbário Irina Delanova Gemtchújnicov* (BOTU), *Herbário da Embrapa Recursos Genéticos e Biotecnologia* (CEN), *Herbário do Centro de Pesquisas do Cacau* (CEPEC), *Herbário Leopoldo Krieger* (CESJ), *Herbário da Fundação Universidade Federal de Mato Grosso do Sul* (CGMS), *Herbário Centro Norte Mato Grossense* (CNMT), *Herbário da Universidade Federal de Mato Grosso do Sul, Campus Pantanal* (COR), *Coleção de plantas medicinais e aromáticas* (CPMA), *Herbário Pe. Dr. Raulino Reitz* (CRI), *Herbário da Reserva Natural Vale* (CVRD), *Royal Botanic Garden Edinburgh Herbarium* (E), *Herbário Prisco Bezerra* (EAC), *Herbário do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas* (EAFM), *Herbário Jaime Coelho de Moraes* (EAN), *Herbário Escola de Florestas Curitiba* (EFC), *Herbário da Escola Superior de Agricultura Luiz de Queiroz* (ESA), *Herbário Friburguense* (FCAB), *Herbário do Departamento de Botânica da Universidade Federal de Santa Catarina* (FLOR), *Fundación Puerto Rastrojo* (FPR-Colombia), *Herbário da Universidade Estadual de Londrina* (FUEL), *Coleção de lâminas de grãos de pólen* (Funed-Pol), *Herbário Dr. Roberto Miguel Klein* (FURB), *Herbário Alarich Rudolf Holger Schultz* (HAS), *Herbarium Hamburgense* (HBG), *Herbário do Instituto de Estudos Costeiros da Universidade Federal do Pará* (HBRA), *Herbário Virtual Flora Brasiliensis* (HbVirtFlBras), *Herbário Caririense Dárdano de Andrade-Lima* (HCDAL), *Herbário da Universidade Tecnológica Federal do Paraná Campus Campo Mourão* (HCF), *Herbário do Departamento de Ciências Florestais* (HDCF), *Herbário Dendrológico Jeanine Felfili* (HDJF), *Herbário Ezechias Paulo Heringer* (HEPH), *Herbário do Pantanal "Vali Joana Pott"* (HPAN), *Herbário Padre Balduino Rambo* (HPBR), *Herbário do Jardim Botânico Plantarum* (HPL), *Herbário do Museu de Ciências Naturais da PUC-Minas* (HPUC-MG), *Herbário Rioclarense* (HRCB), *Herbário Sérgio Tavares* (HST), *Herbário do Trópico Semiárido* (HTSA), *Herbário da Pontifícia Universidade Católica do Paraná* (HUCP), *Herbário da Universidade de Caxias do Sul* (HUCS), *Herbario da Universidade Estadual de Feira de Santana* (HUEFS), *Herbário da Universidade Estadual de Goiás* (HUEG), *Herbário UEM* (HUEM), *Herbário da Universidade Estadual do Sudoeste da Bahia* (HUESB), *Herbarium Uberlandense* (HUFU), *Herbário da Universidade Estadual de Ponta Grossa* (HUPG), *Herbário do Recôncavo da Bahia* (HURB), *Herbário Unisanta* (HUSC), *Herbário Vale do São Francisco* (HVASF), *Herbário do Vale do Taquari* (HVAT), *Herbário do Instituto Agronômico de Campinas* (IAC), *Herbário do Instituto de Ciências Naturais* (ICN), *Herbário INPA* (INPA), *Carpoteca INPA* (INPA-Carpoteca), *Coleção de Fungos INPA* (INPA-Fungos), *Herbário - IPA Dárdano de Andrade Lima* (IPA), *Herbário do Parque da Ciência Newton Freire Maia* (IRAI), *Herbário Joinvillea* (JOI), *Xiloteca Joinvillea* (JOIw), *Herbário Lauro Pires Xavier* (JPB), *Laboratório de Botânica e Ecologia Vegetal* (LABEV), *Herbário de Lages da Universidade do Estado de Santa Catarina* (LUSC), *Herbário do Instituto do Meio Ambiente do Estado de Alagoas* (MAC), *Herbário MACK* (MACK), *Herbário do Maranhão* (MAR), *Herbário do Museu Botânico Municipal* (MBM), *Coleção de Aves MBML* (MBML-Aves), *Herbário Mello Leitão* (MBML-Herbario), *Missouri Botanical Garden - Brazilian records* (MOBOT_BR), *Herbário Dárdano de Andrade Lima* (MOSS), *Herbário do Museu da Pontifícia Universidade Católica do Rio Grande do Sul* (MPUC), *Coleção de Herpetofauna do Museu de Zoologia* (MZUEL-Herpeto), *Museu de Zoologia da Universidade Estadual de Londrina - Coleção de Peixes* (MZUEL-Peixes), *Botanical Collections* (NHM-London-BOT), *Smithsonian Department of Botany - Brazilian records* (NMNH-Botany_BR), *Herbário Nova Xavantina* (NX), *The New*

York Botanical Garden - Brazilian records (NY), *Herbário "Professor José Badini"* (OUPR), *MNHN - Herbário Virtual A. de Saint-Hilaire* (P), *Herbarium Anchieta* (PACA-AGP), *Herbarium Anchieta - Aloysio Sehnem* (PACA-Bryophytes), *Herbarium Anchieta - Fungi Rickiani* (PACA-Fungi), *Herbário Professor Vasconcelos Sobrinho* (PEUFR), *Herbário do Museu Nacional* (R), *RECOLNAT - Herbário Virtual A. Glaziou* (RECOLNAT_Glaziou), *Herbário Rondoniense* (RON), *Herbário do Museu Nacional - Tipos* (R-Tipos), *Herbário de São José do Rio Preto* (SJRP), *Herbário de algas de São José do Rio Preto* (SJRP-Algae), *Herbário de Pteridophyta de São José do Rio Preto* (SJRP-Pteridophyta), *Solanaceae Source - a taxonomic resource for the nightshade family* (Solanaceae_Source_BR), *Herbário do Centro de Ciências e Tecnologias para a Sustentabilidade* (SORO), *Herbário do Estado "Maria Eneyda P. Kaufmann Fidalgo" - Coleção de Fanerógamas* (SP), *Herbário do Estado "Maria Eneyda P. Kaufmann Fidalgo - Coleção de Algas* (SP-Algae), *Maria Eneyda P. Kauffman Fidalgo* (SP-Bryophyta), *Herbário da Universidade de São Paulo* (SPF), *Maria Eneyda Pacheco Kauffmann Fidalgo* (SP-Fungi), *Xiloteca do Instituto de Biociências da Universidade de São Paulo* (SPFw), *Herbário Dom Bento José Pickel* (SPSF), *Herbário Tangará* (TANG), *Herbário Graziela Barroso* (TEPB), *Herbário da Universidade de Brasília* (UB), *Herbário da Universidade Estadual de Campinas* (UEC), *Herbário Universidade Estadual de Santa Cruz* (UESC), *Coleção Entomológica da UFES* (UFES-Entomologia), *Herbário da Universidade Federal de Goiás* (UFG), *Herbário UFMT* (UFMT), *Coleção Zoológica da UFMT- Setor Herpetologia-Répteis* (UFMT-R), *Herbário UFP - Geraldo Mariz* (UFP), *Herbário UFRN* (UFRN), *Herbário da Universidade Estadual do Oeste do Paraná* (UNOP), *Herbário do Departamento de Botânica* (UPCB), *Herbário Pe. Camille Torrand* (URM), *Herbário da Universidade Federal de Viçosa* (VIC), *Herbário Central da Universidade Federal do Espírito Santo VIES* (VIES) disponível na rede *species*Link (http://www.splink.org.br) em 08 de outubro de 2016.

HAYKIN, S.; LIPPMANN, R. Neural Networks, a comprehensive foundation. International Journal of Neural Systems. v. 5, n. 4, p. 363-364, 1994.

HOLLAND, P. W.; THAYER, D. T. Differential item performance and the Mantel-Haenszel procedure. Test validity, p. 129-145, 1988.

HOSMER, D. W.; LEMESHOW, S. Applied logistic regression (Willey Series in probability and statistics). 2 ed. Wiley-Interscience Publication, 2000.

KAUFMAN, K. A. MICHALSKI, R. S. From data mining to knoeledge mining. In Handbook in Statistics, volume 24: Data Mining and Data Visualization, p. 47- 75. Elsevier, 2005

KIRA, K.; RENDELL, L. A practical approach to feature selection. International Conference on Machine Learning. p 249- 256, 1992.

KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. Artificial Intelligence, 1997.

LEK, S.; GUÉGAN, J. F. Artificial neural networks as a tool in ecological modelling, na introduction. Ecological modelling, v. 120, n. 2, p. 65-73, 1999.

LIU, H.; MOTODA, H. Computational Methods of Feature Selection. Chapman & Hall/CRC, 2008.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, v. 5, n. 4, p. 115-133, 1943.

MITCHELL, T. M. Machine Learning. New York: McGraw-Hill, 1997.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre Aprendizado de Máquina. In: Solange Oliveira Rezende. (Org). Sistemas Inteligentes - Fundamentos e Aplicações. Barueri, 1 ed. Editora Manole, 2003.

OLIVEIRA, P. H. M. A. Detecção de fraudes em cartões: um classificador baseado em regras de associação e Regressão Logística. 2015. 100f. Dissertação (Mestrado em Ciências da Computação) – Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo.

PEREIRA, G. H. A.; CENTENO, J. A. S. Avaliação do tamanho de amostras de treinamento para redes neurais artificiais na classificação supervisionada de imagens utilizando dados espectrais e *laser scanner*. Boletim de Ciências Geodésias, v.23, n. 2, p. 268-283, 2017.

REZENDE, S. O. Sistemas Inteligentes: Fundamentos e aplicações, Baurueri – SP: [s.n.], São Paulo, 2003.

ROSENBLATT, F. Principles of Neurodynamics: Perceptrons and the theory of Brain Mechanisms. Spartan, New York. 1962.

RUSSELL, S.; NORVIG, P. Artificial Inteligence – A Modern Approach, 2 ed. Elsevier Prentice Hall, 2003.

QUINLAN, J. Ross. Induction of decision trees. Machine learning, v. 1, n. 1, p. 81-106, 1986.

QUINLAN, J. Ross. Simplifying decision trees. International journal of man-machine studies, v. 27, n. 3, p. 221-234, 1987.

QUINLAN, J. R. Combining instance-based and model-based learning. In: Proceedings of the Tenth International Conference on Machine Learning. 1993. p. 236-243.

SHANNON, C. E. A mathematical theory of communication. Bell System Technical Journal, v. 27. p. 237-423, 1948.

STANFILL, C.; WALTZ, D. Toward memory-based reasoning. Communications of the ACM, v. 29, n. 12, p. 1213-1228, 1986.

TAKASHI, E. M. Relações entre ranking, análise ROC e calibração em aprendizado de máquina. 2008. 149f. Tese (Doutorado em Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo. São Paulo.

TEAM, R. C. R: A language and environment for statistical computing [Internet]. Vienna, Austria: R Foundation for Statistical Computing; 2013. Documento disponível na internet em: http://www. r-project.org, acessado em 28 de setembro de 2015.

TEAM, RStudio. RStudio: integrated development for R. RStudio, Inc., Boston, MA; 2014. Documento disponível na internet em: http://www. RStudio.com/ide, acessado em 12 de outubro de 2015.

WEISS, S. M.; KULIKOWSKI, C. A. Computer Systems that Learn. Classification and Prediction Methods from Statistics, Neural Nets, Machine, and Expert Systems. Morgan Kaufmann, San Mateo, 1990.

YU, L.; LIU, H. Feature selection for high-dimensional data: A fast correlation-based filter solution. In: ICML. 2003. p. 856-863.

ZAMONER, F. W. Técnica de aprendizado semi-supervisionado para detecção de outliers. 2013. Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.

ZHANG, K.; HUTTER, M.; JIN, H. A new local distance-based outlier detection approach for scattered real-world data. 13th Pacific-Asia Conference, PAKDD. Bangkok, Thailand, v. 5476, p. 813-822, 2009.