UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

WILLIAM CARNEIRO LIMA

ESTUDO DE CLASSIFICADORES E CONSTRUÇÃO DE DATASET PARA DIAGNÓSTICO DE DENGUE, CHIKUNGUNYA E ZIKA

TRABALHO DE CONCLUSÃO DE CURSO

PONTA GROSSA
2017

WILLIAM CARNEIRO LIMA

ESTUDO DE CLASSIFICADORES E CONSTRUÇÃO DE DATASET PARA DIAGNÓSTICO DE DENGUE, CHIKUNGUNYA E ZIKA

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas do Departamento Acadêmico de Informática, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Drª. Simone de Almeida

Co-orientador: Prof. Marcos Vinicius Fidelis

PONTA GROSSA

2017

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Ponta Grossa
Diretoria de Graduação e Educação Profissional
Departamento Acadêmico de Informática
Tecnologia em Análise e Desenvolvimento de Sistemas

TERMO DE APROVAÇÃO

ESTUDO DE CLASSIFICADORES E CONSTRUÇÃO DE DATASET PARA DIAGNÓSTICO DE DENGUE, CHIKUNGUNYA E ZIKA

por

WILLIAM CARNEIRO LIMA

Este Trabalho de Conclusão de Curso (TCC) foi apresentado em 16 de novembro de 2017 como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

_____
Simone de Almeida
Prof.(a) Orientador(a)

_____
Geraldo Ranthum
Membro titular

_____
Marcos Vinicius Fidelis
Membro titular

_____
Profª. Helyane Bronoski Borges
Responsável pelo Trabalho de Conclusão de Curso

_____
Profª. Mauren Louise Sguario
Coordenadora do curso

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso -

Dedico este trabalho a minha família, meus amigos e a todos que estiveram ao meu lado acreditando e me apoiando em todos os momentos.

AGRADECIMENTOS

A minha família, pelo apoio, compreensão em entender que a distância é necessária algumas vezes para o crescimento do indivíduo. Em especial para minha mãe Margarida, que veio a falecer nesse ano de 2017, mas me deu a melhor educação que podia ter dado e sempre com muito amor, sendo meu apoio em muitos momentos. Aos meus amigos que me apoiaram e incentivaram cada momento para eu ter forças e poder concluir está importante etapa em minha vida. A Universidade Tecnológica Federal do Paraná – Campus Ponta Grossa, e ao seu corpo docente, administração, direção, funcionários responsáveis pela limpeza e pela manutenção que auxiliam a criar um ambiente educacional prazeroso.

Agradeço a Deus por ter me dado saúde, paz, força e sabedoria para poder superar as adversidades. A minha professora Orientadora, Doutora Simone de Almeida, que sem ela não teria concluído esta etapa. Ao meu professor co-orientador Marcos Vinicius Fidelis, que auxiliou muito em meu aprendizado e despertou, através de seu ensinamento, o interesse profundo na área de análise de dados para descoberta de conhecimento.

Agradeço a alguns grandes amigos, que fazem parte de minha vida e formação pessoal, Thomaz España, que me engrandeceu com as seguintes palavras “Sejam bons se puderem, se não puderem ao menos tentem”, meus amigos do Colégio Santo Agostinho que mantemos amizade até os dias de hoje. Meu muito obrigado de coração. Agradeço a dois amigos que viraram irmãos, Daniel Lee e Daniel Keller, a minha namorada Maria Cacilda Monteiro, que entendeu toda a dificuldade e sacrifício que a graduação nos exige. As minhas amigas Késia Siqueira, Julia Yonamine, Ruth Reckziegel, Liliane Basile, Janaina Castro, Carol Akiko, Karina Dell’Àquila, Jane Sakugawa entre outras, que, mesmo distantes estão sempre presente em minha vida. Aos meus amigos da cidade de Ponta Grossa-PR, como Bruna Moscardi, Fabio Seidl, Thiago Cordeiro, Carmella Corazza, Marcela Kloth, que tornaram mais fácil e agradável meu convívio em um local totalmente novo.

Agradeço a todos que direta ou indiretamente fazem parte da minha formação, muito obrigado.

Nem mesmo o conhecimento dos estudiosos mais consagrados não pode competir com os computadores cognitivos e além disso, como a quantidade de informação é exponencialmente crescente, o uso da computação para auxiliar na tomada de decisão médica é eminente e inevitável.
(MESKÓ, Bertalan, 2014)

RESUMO

LIMA, William Carneiro. **Estudo de Classificadores e Construção de Dataset para Diagnóstico de Dengue, Chikungunya e Zika:** 2017. 79f. Trabalho de Conclusão de Curso Tecnologia em Análise e Desenvolvimento de Sistemas - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2017.

Sistemas Computacionais simulam a cognição humana utilizando uma ampla base de informações. Diferente do ser humano, a máquina não se esquece de informações e pode compartilhar seu conhecimento com diversas pessoas ao mesmo tempo, em lugares diferentes. Neste trabalho o sistema desenvolvido utilizou informações aplicadas na área de medicina diagnóstica. As doenças selecionadas para estudo são dengue, chikungunya e zika, pois a sua incidência cresce a cada ano em países tropicais e suas consequências são perigosas. O Sistema desenvolvido vai auxiliar no diagnóstico destas três doenças, e fazer com que o tratamento seja iniciado rapidamente. Fazer com que profissionais em diversos locais tenham acesso a informação e utilizem o sistema para auxilio quando estiverem em áreas mais carentes ou de difícil acesso médico.

Palavras-chave: Árvores de Decisão. Chikungunya. Dengue. Medicina Diagnóstica. Mineração de Dados. Zika.

ABSTRACT

LIMA, William Carneiro. **Study of Classifiers and Contruction of Dataset for Diagnosis of Dengue, Chikungunya and Zika.** 2017. 79p. Work of Conclusion Course Graduation in Technology in Systems Analysis and Development - Federal Technology University of Parana. Ponta Grossa, 2017.

Computing Systems simulates a human cognition by using wide bases of information. By being different than humans, machines don't forget information and may share it's knowledge with many people at the same time, even in different places. In this work, the developed system used information applied in the area of diagnosed medicine. The diseases selected to study are dengue, chikungunya and zika, since their incidence grow each passing year in tropical countries, and their consequences are dangerous. The developed system is going to help diagnosing these three diseases, which will help to begin their treatment faster. If professionals in many different places have access to this information and use the system to help them when they are in poorer areas or with harder medical assistance.

Keywords: Decision Trees. Chikungunya. Dengue. Diagnosed Medicine. Data Mining. Zika.

LISTA DE TABELAS

LISTA DE FIGURAS

LISTA DE SIGLAS E ACRÔNIMOS

AAS	Ácido Acetil Salicílico
ANS	Agência Nacional de Saúde
API	*Aplication Program Interface*
ARFF	*Atribute-Relation File Format*
CSS	*Cascade Style Sheets*
IA	Inteligência Artificial
KDD	*Knowledge Discovery Data*
LMT	*Logistic Model Trees*

SUMÁRIO

1. INTRODUÇÃO

Com foco na especialização e no menor tempo de aprendizado, a análise de classificadores se torna uma tarefa importante para a descoberta de conhecimento. A quantidade de informações gerada a cada minuto pode ser imensa, e o ser humano torna-se incapaz de assimilar e administrar tais conhecimentos (REZENDE, 2003).

A área médica tem sido uma das áreas mais beneficiadas pela tecnologia, por ser considerada detentora de problemas clássicos, possuidores de todas as peculiaridades necessárias, para serem instrumentalizados por tais sistemas (NILSON, 1982).

Neste contexto, a avaliação dos classificadores e sua implementação em um sistema que auxilie no diagnóstico de doenças como a dengue, chikungunya e zika, que hoje afetam uma grande parte da população mundial, é totalmente possível.

A dengue, chikungunya e zika, possuem similaridades em seus sintomas, mas o tratamento de cada uma é distinto, e caso um diagnóstico errôneo seja feito, pode trazer severas complicações à vida do paciente. Um sistema com um classificador confiável, que possua o conhecimento das particularidades de cada doença já citada, auxiliará no diagnóstico de maneira rápida e precisa.

Neste trabalho propõe-se analisar alguns classificadores e construir um sistema de fácil interação que os utilize, e alimentá-lo com uma extensa base de dados, com os sintomas, índice de incidência e combinação com outros fatores característicos de cada uma destas doenças para obter um diagnóstico com alto grau de assertividade.

1.1 DESCRICAO DO PROBLEMA

Na área médica, quanto mais rápido um diagnóstico for feito melhor será o resultado do tratamento. Um Sistema que faça classificação aplicado a medicina auxilia médicos a chegarem em um diagnóstico com precisão. A IBM

possui um computador denominado WATSON, que simula o aprendizado de especialistas e auxilia no diagnóstico de diversos tipos de câncer. Esse conceito pode ser aplicado em todas as áreas da medicina. Em menor escala, verificando a eficácia de um classificador pode construir um sistema com uma fração dessa ideia, e como no Brasil a incidência de dengue, chikungunya e zika é alta e vem crescendo nos últimos anos de acordo com o Ministério da Saúde (2017), um sistema que consiga diagnosticar com rapidez uma doença será muito útil para a sociedade.

1.2 OBJETIVOS

Esta Seção apresenta o objetivo geral do desenvolvimento deste trabalho, assim como os objetivos específicos a serem obtidos.

1.2.1 Objetivo Geral

Estudar classificadores de dados e desenvolver um sistema que utilize o melhor algoritmo classificador encontrado para gerar diagnóstico médico a partir de dados específicos sobre as doenças dengue, chikungunya e zika.

1.2.2 Objetivos Específicos

- Pesquisar ferramentas que permitam a criação de um *dataset*;
- Detalhar as particularidades de cada uma das doenças dengue, chikungunya e zika;
- Criar as regras de inferência na ferramenta selecionada;
- Experimentar os classificadores;
- Validar o sistema através de testes e análise de resultados.

1.3 JUSTIFICATIVA

A computação está cada vez mais presente no cotidiano. Reunir uma gama de informações que possam ser compartilhadas e processadas em pouco tempo não é uma tarefa fácil. Como ferramenta de auxilio, são desenvolvidos sistemas que, de maneira objetiva consegue concentrar a informação e deixá-la disponível a todos a qualquer momento, bastando ter acesso a ferramenta que permita seu uso. Um sistema convencional com a utilização de classificadores contribui muito para agilizar o acesso a informações que demorariam a serem processadas por humanos.

Na área médica, existe muitas doenças cujo diagnóstico é similar, podendo levar a um resultado falso positivo, tratamento errado e assim sendo, consequências graves e até a morte. Pensando nisto, um sistema convencional que utiliza classificadores serve como um guia para o profissional da saúde, não deixando que pequenos detalhes que dificilmente seriam percebidos pelo ser humano passe despercebido.

Dengue, Chikungunya e Zika são três doenças com diagnósticos parecidos, e que infecta milhares de pessoas no mundo todo. Com o objetivo de facilitar o diagnostico dessas doenças, um sistema convencional que utilize classificadores tende a ser útil, pois sua base de dados possuem todas as características de cada enfermidade. Para fins acadêmicos, o estudo será sobre estas três doenças, podendo ser estendido em muitas partes da área médica.

1.4 ORGANIZACAO DO TRABALHO

O projeto está divido em seis Capítulos principais. O segundo Capítulo apresenta os conceitos de aprendizado de máquina, mineração de dados e definições de algoritmos de classificação estruturados em árvores. O terceiro Capítulo detalha os sintomas, diagnósticos e tratamento das doenças dengue, chikungunya e dengue. O quarto Capítulo discorre sobre a composição e construção de um seguindo os padrões de um arquivo ARFF. O quinto Capítulo exibe os resultados obtidos pelo trabalho realizado e comparações entre os

algoritmos utilizados. Finalizando o trabalho, o sexto Capítulo conclui e fala sobre trabalhos futuros onde o projeto pode ser aproveitado.

2. REVISÃO BIBLIOGRÁFICA

Este Capítulo apresenta alguns conceitos sobre sistemas de classificação e sua utilização. O Capítulo está organizado em seis seções apresentando os seguintes tópicos: A Seção 2.1 apresenta conceitos e definições sobre descoberta de conhecimento, o KDD. A Seção 2.2 aprofunda na descoberta de conhecimento e também na mineração de dados e nas tarefas de classificação. Na Seção 2.3 é tratado sobre o aprendizado de máquina. A Seção 2.4 apresenta conceitos e definições da Inteligência Artificial, o IA e também sobre sua utilização na medicina. A Seção 2.5 descreve o programa Weka muito utilizado neste trabalho. A Seção 2.6 faz algumas considerações sobre o Capítulo.

2.1 KDD, DATAMINING E TAREFAS DE CLASSIFICAÇÃO

A área denominada Descoberta de Conhecimento em Base de Dados *(Knowledge Discovery in Databases – KDD)*, surgiu em 1989, referenciado ao amplo conceito de procurar conhecimento a partir de base de dados. Segundo Fayyad et al (1996), “KDD é um processo, de várias etapas, não trivial, interativo e iterativo, para a identificação de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grande conjunto de dados”.

Uma das etapas do KDD é a mineração de dados, também conhecida como *Data Mining*. Esse processo é responsável por identificação de padrões nos dados analisados e deve apresentar alguma vantagem para o analista, normalmente econômica. Mineração de dados é uma área que envolve muitas disciplinas, utilizando métodos de diversas áreas em especial de Aprendizado de Máquina e Estatística para extrair conhecimento a partir de conjunto de dados (COELHO,2011).

Na Descoberta de Conhecimento o resultado final deve ser compreensível. Definir essa compreensibilidade não é fácil. Dependendo do algoritmo usado, a simplicidade pode ajudar nessa compreensão, como por exemplo o número de nós em uma árvore de decisão.

Subcampo da ciência da computação, o aprendizado de máquina é uma evolução do estudo do reconhecimento de padrões e da teoria do aprendizado computacional em inteligência artificial. A aprendizagem de máquina é relativa a questão de como são construídos programas de computador que melhorem seu conhecimento através da experiência (Mitchell, 1997). Algoritmos de aprendizagem são úteis, e tem um grande valor quando posto em prática em aplicações, podendo ser citados:

- Uso de algoritmos na extração de dados, onde base de dados podem conter regularidades ocultas valiosas, que podem ser descobertas de maneira automática. Por exemplo, analisar uma base de dados de pacientes, correlacionando sintomas e doenças auxiliando nos tratamentos médicos;
- Conhecimento em áreas que os seres humanos podem não ter o conhecimento necessário para desenvolver regras eficientes. Por exemplo, leitura de retina para acesso de segurança;
- Facilidade de adaptação em situações que programas tem de se adaptar dinamicamente às mudanças das condições. Por exemplo, controle de venda em períodos festivos.

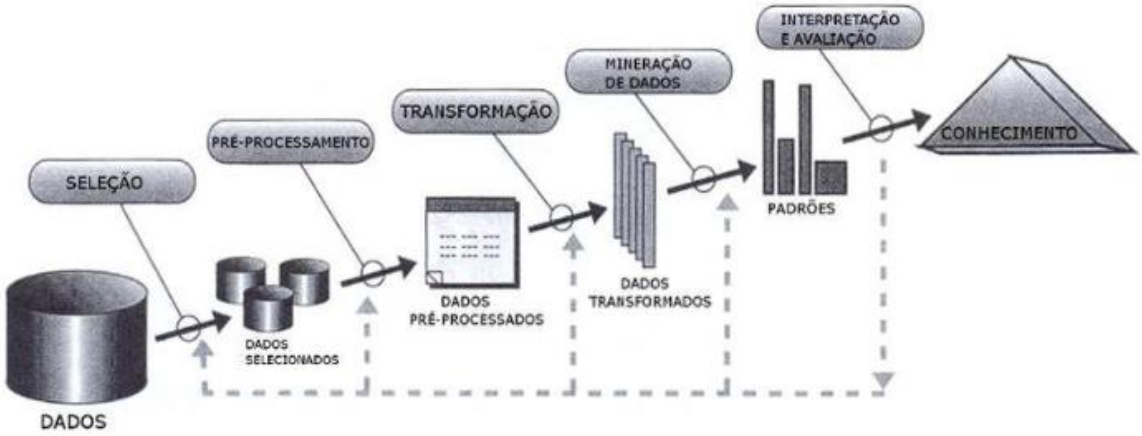
Grande quantidade de dados são armazenados diariamente. Esses dados podem conter informações valiosas implícitas, que poderiam ser utilizados para análise, diagnósticos, simulação e/ou prognóstico do processo que gerou a base de dados (HAN; KAMBER; PEI, 2012).

Sem ferramentas computacionais apropriadas, a análise de grande quantidade de dados se torna inviável. Neste contexto, torna-se imprescindível o desenvolvimento de ferramentas que auxiliem o homem, de forma automatizada e inteligente, na tarefa de análise, interpretação e relacionamento entre os dados para que se possa desenvolver e selecionar estratégias de ação em cada contexto de aplicação (GOLDSCHMIDT; PASSOS, 2005).

Além da etapa de Mineração de Dados, que faz a descoberta do conhecimento a partir de dados, o processo de mineração inclui outras etapas como a de pré-processamento, que prepara os dados, e pós processamento, que faz um refinamento do conhecimento descoberto. Conforme mostrado na

Figura 1, o objetivo do pré-processamento de dados é de transformar os dados, facilitando a aplicação de técnicas de Mineração de Dados e o objetivo dos métodos de refinamento é validar e aperfeiçoar o conhecimento adquirido (COELHO,2011).

Figura 1 – Processo de Descoberta de Conhecimento em Mineração de Dados



Fonte: Adaptado de Han, Kamber e Pei (2012)

A etapa de pré-processamento de dados é necessária na maioria das vezes, permitindo que os dados sejam utilizados adequadamente no processo de Mineração de Dados. Segundo Coelho (2011), as etapas a seguir devem ser seguidas:

- Limpeza dos dados:
  - Preenchimento de dados ausentes, identificar ou remover ruídos, resolver inconsistências.
- Integração dos dados;
  - Integração de múltiplas bases de dados, normal e agregação.
- Transformação dos dados;
  - Normalização dos dados.
- Discretização dos dados;
  - Importante para dados numéricos.
- Redução dos dados;
  - Redução no volume de dados com resultados similares.
- Seleção dos dados.

Na parte de aprendizagem de máquina, temos alguns métodos que serão tratados nas seções a seguir.

2.2 KDD – DESCOBERTA DE CONHECIMENTO EM BASE DE DADOS

A mineração de dados é o processo que descobre padrões e conhecimento que não são sabidos previamente. A fase de mineração de dados envolve a decisão de quais algoritmos serão aplicados a base de dados em estudo. Nesta fase diversos algoritmos podem ser aplicados de diferentes áreas de conhecimento como aprendizado de máquina, estatística, redes neurais e banco de dados. (REZENDE, 2003).

O objetivo dessa fase é criar um modelo preditivo, então, a decisão do melhor algoritmo não é uma tarefa fácil, pois é sabido que nenhum algoritmo é ótimo para todas as situações (KIBLER; LANGLEY, 1988).

A mineração de dados como dito anteriormente, é a etapa essencial do processo de descoberta de conhecimento onde se obtém padrões que mostrem importância para análise em um processo específico. Outras etapas importantes do KDD, conforme pode ser verificado na Figura 1 são:

- Seleção de dados, fazendo o refino dos dados utilizando os relevantes;
- Pré-processamento, utilizando os dados relevantes;
- Transformação, melhorando os dados;
- Mineração de dados, onde se descobre os padrões para adquirir o conhecimento;
- Interpretação, de onde se lê os dados e adquire-se o real conhecimento.

2.2.1 Fases do Processo KDD

A seleção de dados e o seu refino é o passo inicial para que seja cumprido o processo de descoberta de conhecimento. Neste trabalho, a etapa

de pré-processamento se iniciou com a seleção de atributos comumente encontradas nas doenças estudadas e que, os critérios utilizados são mostrados no Capítulo 3.

Com a definição dos critérios a serem utilizados, entramos na fase de pré-processamento, onde é feita a integração de dos dados, eliminação de dados faltantes e redundantes para que os dados processados assumam um único formato. Seguindo as etapas, temo o processo de mineração, no qual são extraídos padrões e posteriores análises com o fim de se tomar decisões com o conhecimento adquirido (REZENDE, 2003). Essa fase tem o objetivo de melhorar os dados selecionados, pois os mesmos podem ter informações que ainda não poderão ajudar na descoberta de conhecimento. Os dados nessa fase ainda passam pelo processo de limpeza e integração para que posteriormente sofram transformação (REZENDE, 2003).

A etapa após o pré-processamento, a transformação de dados consiste, como o nome diz, em converter os dados que possuem formatos diferentes em formatos que possam ser lidos pelo algoritmo de mineração. Os dados são padronizados em um formato único para que o algoritmo os entenda (REZENDE, 2003).

Para esse trabalho foi gerado código em linguagem *Pyton* mostrada posteriormente nos resultados que gerou um arquivo de extensão ARFF, que é lido pelo Weka onde foi testado a eficácia dos classificadores.

A mineração de dados, vindo em sequência, é a fase que minera os padrões para adquirir conhecimento, detalhado no subitem a seguir. Com os dados em mãos, pode surgir vários interesses em aprender mais sobre eles, por meio de visualizações por exemplo, ou ainda alterar as estruturas dos dados por meio. Essa fase de preparação dos dados objetiva preparar os dados para a fase seguinte, deixando a extração do conhecimento mais efetiva.

2.2.2 Teorias da Mineração

A mineração de dados é um ramo da computação que teve início nos anos 80. Nessa época, os empresas e organizações começaram a se preocupar

com os grandes volumes de dados que eram produzidos, estocados e inutilizados pelas empresas. Nesse período a mineração de dados consistia somente em extrair informação de grandes bases de dados de maneira mais automatizada possível.

Nos dias atuais, a mineração de dados consiste na análise de dados após a descoberta do conhecimento, buscando, por exemplo, criar soluções de marketing ou investimentos com base nos resultados obtidos.

Na fase de mineração de dados, é analisado o problema e decidido qual algoritmo de classificação será utilizado.

2.2.3 Mineração de dados

Mineração de dados é a exploração e a análise de forma automatizada ou semi-automatizada, de grandes quantidades de dados, com a finalidade de descobrir regras e padrões significativos (BERRY; LINOFF, 1997).

O processo de mineração de dados baseia-se na interação entre várias classes de usuários, e esse sucesso depende, em grande parte, dessa interação. Os usuários desse processo podem ser divididos em três categoria: especialista do domínio, que deve possuir grande conhecimento da aplicação e oferecer apoio para a execução do processo; analista, que deve realizar a parte de extração de conhecimento e dominar todas as etapas que fazem parte do processo; e o usuário final, que utiliza o conhecimento obtido no processo para a tomada de decisão (REZENDE, 2003).

Os objetivos principais da mineração de dados são: descobrir os relacionamentos entre dados e fornecer subsídios para que se possa fazer uma previsão de tendências futuras, baseadas no passado.

A mineração de dados é uma etapa do *Knowledge Discovery Data* (KDD). A descoberta de conhecimento auxilia as empresas na análise das informações contidas em suas bases de dados. As informações descobertas serão utilizadas no auxílio da tomada de decisão, otimizando o processo e retornando de forma eficiente a informação para que se defina a estratégia mais adequada a ser utilizada. O KDD é uma técnica que possibilita analisar grandes

conjuntos de dados, utilizando métodos aproximados (COLLAZA; BARRETO, 2003).

2.2.4 Avaliação dos Modelos dos Classificadores

Esse trabalho fez a avaliação de um *dataset* da área médica. Para isso, foi avaliado que uma boa solução de estudo seria a utilização dos algoritmos de árvores de decisão, pois a assimilação dos resultados em uma árvore de decisão é de mais fácil compreensão para o ser humano. Os algoritmos estudados nesse trabalho foram:

- DecisionStump
- HoeffdingTree
- J48 (C4.5)
- LMT
- RandomForest
- RandomTree

O Algoritmo *DecisionStump* é a representação de uma árvore de decisão “Firme”, na qual consiste em um único nó e dois nós folhas para predição.

O Algoritmo HoeffdingTree é um algoritmo de indução de árvore de decisão incremental e em qualquer momento é capaz de aprender com fluxos de dados maciços, assumindo que os exemplos que geram a distribuição não mudam ao longo do tempo. Esse algoritmo explora a ideia que uma pequena amostra pode ser satisfatória para escolher um bom atributo de divisão.

Talvez um dos mais conhecidos algoritmos de árvore de decisão o J48, também conhecido como C4.5, constrói uma árvore de decisão a partir de um conjunto de dados de treinamento, utilizando o conceito estatístico de amostras já classificadas. Em cada árvore, o J48 escolhe os atributos dos dados que mais efetivamente particiona o seu conjunto de amostras em subconjuntos, que tendem a outra subcategoria.

O Algoritmo *LMT* é um algoritmo de indução, que particiona sucessivamente o conjunto de treino original em subconjuntos menores, deixando, após a sua construção, uma fácil interpretação da árvore gerada.

O Algoritmo *RandomForest* gera várias árvores de decisão, cada uma com suas particularidades e combina o resultado da classificação de todas elas.

O Algoritmo *RandonTree* funciona semelhante ao *RandonForest*, porém ele analisa o resultado de cada árvore e apresenta o a de maior incidência.

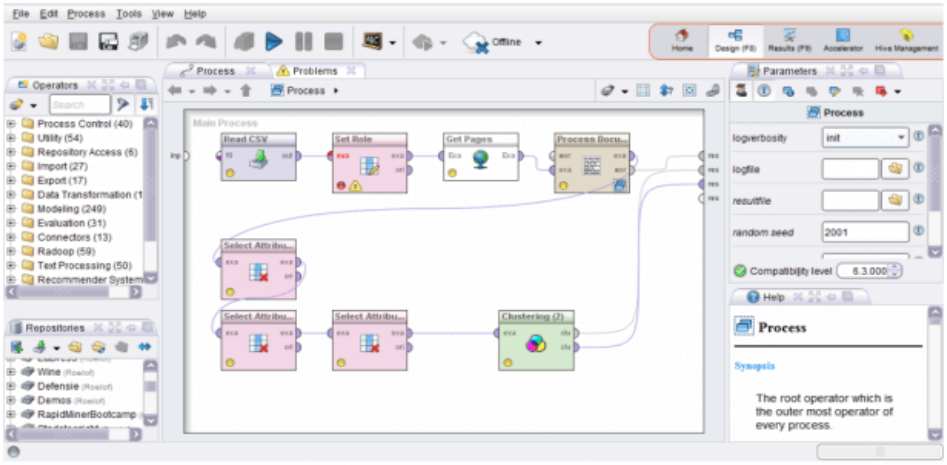
2.2.5 Ferramentas de Mineração de Dados

Existe no mercado muitas ferramentas que podem ser usadas para mineração de dados, entre elas podem ser citadas ferramentas de *Pyton*, de *R*, o *RapidMiner* e o Weka. *Pyton* é uma linguagem de programação com características como clareza, reusabilidade e uma certa simplicidade (IMASTERS, 2017). A linguagem é conhecida por oferecer uma linguagem simples e objetiva, permitindo o foco no problema ao invés da programação. Nesta linguagem, existe bibliotecas especializadas na mineração de dados, como a *Jupyter*, que faz uma aplicação entre cliente-servidor, a *Numpy*, usada para computação científica, *Matplotlib*, usada para visualização de gráficos, a biblioteca Pandas que fornece ferramentas de manipulação de estruturas entre outras. É uma ferramenta nova que vem se destacando no mercado. (IMASTERS, 2017).

R é uma outra linguagem de programação que pode ser usada na mineração de dados. Extremamente poderosa e vem se destacando na área de *Data Science* é conhecida por sua facilidade para fazer análise de dados e processar informações estatísticas e modelos gráficos.

O *RapidMiner* é uma plataforma visual, que promete uma forma rápida e simples de trabalhar na mineração de dados. Suas ferramentas oferecem interface gráfica com muitos objetos com o objetivo de simplificar a análise de resultados. Um exemplo da interface pode ser visto na Figura 2:

Figura 2 – Interface RapidMiner



Fonte: Adaptado site imaster.com - 2017

O diferencial do *RapidMiner* está na facilidade e velocidade que mostra para criar modelos preditivos, pois não é necessário o trabalho de codificação e transformação de dados (IMASTERS, 2017).

A ferramenta Weka é um projeto *open source* que tem o objetivo de disseminar técnicas de aprendizado de máquina. Essa ferramenta foi a escolhida para o desenvolvimento desse trabalho, e é explicada mais detalhadamente na Seção seguinte.

2.3 APRENDIZAGEM DE MÁQUINA

Aprendizado de máquina é um subcampo da ciência da computação que que sofreu evolução da parte que estuda o reconhecimento de padrões e da teoria do aprendizado computacional da área de inteligência artificial. Seus métodos são tratados nos subtópicos seguintes (SAS INSTITUTE INC., 2017).

2.3.1 Aprendizagem de Máquina – Reforço

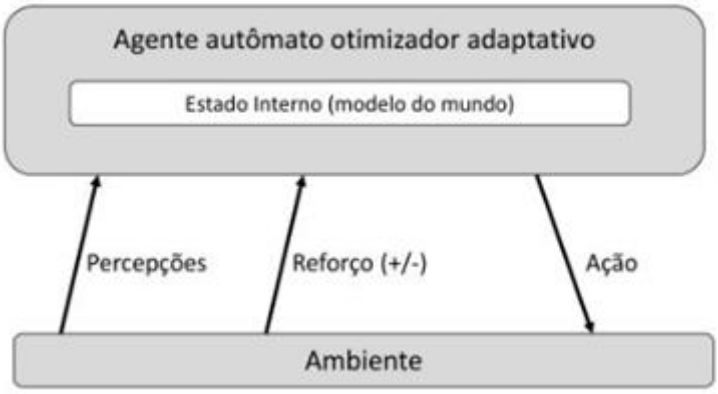
O aprendizado por reforço consiste no aprendizado através de uma política de ações, ou seja, de que maneira o agente deve agir para que suas recompensas futuras sejam maximizadas. Esse aprendizado é comumente utilizado na robótica, jogos e navegação. Aprendendo por tentativa e erro, o

algoritmo descobre quais ações geram maior recompensa. Segundo a SAS Institute Inc. (2017), a aprendizagem por reforço é composta por três componentes principais:

- O Agente (Tomador de decisões);
- O Ambiente (Tudo com o qual o agente interage);
- As Ações (O que o agente vai fazer).

A aprendizagem por reforço objetiva que o agente escolha ações que maximizem a recompensa esperada ao longo de um determinado período de tempo, podendo assim futuramente atingir o objetivo de maneira mais rápida seguindo o melhor caminho. Com essas informações, note-se que o objetivo do aprendizado por reforço é aprender a melhor política, conforme ilustrado na Figura 3.

Figura 3 – Aprendizado por reforço



Fonte: INF 1771 (2017)

2.3.2 Aprendizagem de Máquina – Supervisionado

O aprendizado de máquina supervisionado consiste no treinamento usando exemplos rotulados, como uma entrada onde a saída desejada é conhecida. Por exemplo, um pneu fabricado pode ter ponto de dados rotulados como falha e outros pontos como bons. O algoritmo em estudo recebe um conjunto de entradas junto com suas saídas correspondentes corretas, e o aprendizado é adquirido na comparação da saída real com as saídas corretas, encontrando nas diferenças os erros e, em seguida ele modifica o modelo de acordo (SAS INSTITUTE INC, 2017).

Utilizando métodos de classificação como a regressão, previsão e *boosting* do gradiente, o aprendizado supervisionado utiliza de padrões para prever os valores dos rótulos em dados adicionais não rotulados. Este tipo de aprendizado normalmente é usado em aplicações nos quais os dados históricos auxiliam na previsão de acontecimentos futuros.

2.3.3 Aprendizagem de Máquina – Não-Supervisionado

O aprendizado de máquina não supervisionado é usado nos dados que não possuem rótulos históricos, ou seja, o sistema não sabe qual a resposta certa da entrada. O Algoritmo que tem a função de fazer a descoberta do que está sendo mostrado, explorando os dados e encontrar uma estrutura neles. Esse tipo de aprendizado funciona bem em dados transacionais. Por exemplo, estudar o padrão de clientes de um mercado para que possa ser produzida uma ação de marketing direcionada ou mesmo atributos que diferenciem clientes, agrupando-os em semelhanças.

As técnicas mais populares podem incluir mapas auto organizáveis, mapeamento de vizinhança, agrupamento e decomposição em valores singulares. Esses algoritmos podem ser usados também na segmentação de tópicos de texto, recomendar itens e identificar valores com muita diferença em dados.

2.3.4 Aprendizagem de Máquina – Semi-Supervisionado

O aprendizado de máquina semi-supervisionado é utilizado nas mesmas aplicações que o aprendizado supervisionado, porém com a diferença de usar tanto dados rotulados quanto não marcados para o treinamento. Comumente é usado uma pequena quantidade de dados rotulados e uma maior parte de dados não rotulados.

Segundo a SAS Institute Inc. (2017), os dados não rotulados são mais baratos e exigem menos esforços para serem adquiridos. Esse tipo de

aprendizagem pode ser utilizado com métodos como a classificação, a previsão e a regressão. O aprendizado de máquina semi-supervisionado é útil quando levado em conta o custo associado a rotulagem, pois esses valores, dependendo dos dados, podem ser muito altos.

Dados digitais, como identificação de rosto de uma pessoa por câmeras, podem ter um valor elevado se comparados a outras informações que são menos complexas, como dados sobre a preferência alimentar de um indivíduo. O Método semi-supervisionado utiliza de dados não rotulados em sua maioria, criando sua própria regra.

2.4 INTELIGÊNCIA ARTIFICIAL

A Inteligência Artificial (IA) é uma área que merece destaque na atualidade. Muitos desafios motivam o estudo de IA, principalmente a corrida para que máquinas simulem, com precisão, a cognição humana por meio de programas desenvolvidos para este fim.

Vista por muitos como algo futurista, a IA já está presente no dia a dia das pessoas. Algoritmos aplicados, por exemplo, em redes sociais, aprendem sobre os interesses do usuário e sugerem filmes, músicas, compras, roupas, de uma forma personalizada. A IA pode ser aplicada em áreas como robótica, jogos, visão artificial como óculos RIFT (Realidade virtual), sistemas tutores inteligentes, sistemas militares, instituições bancárias, programas de diagnósticos médicos e outros.

Segundo Rezende (2003), a pesquisa sobre IA era realizada apenas para procurar novas funcionalidades para o computador. A Segunda Guerra Mundial impulsionou o estudo da IA, pois criou a necessidade de desenvolver tecnologias para a indústria bélica assim como novas armas. IA auxiliou tanto na quebra de códigos sigilosos em mensagens criptografadas quanto na produção da bomba atômica.

Após o término da guerra, o computador não ficou limitado ao uso militar e científico. Aos poucos começaram e serem usados em universidades, empresas, indústrias e outros. O desenvolvimento da tecnologia fez com que o

computador se tornasse necessário para o auxílio do desenvolvimento principalmente em empresas. (LIMA; LABIDI, 2001).

Segundo Barreto (2001), Alan Turing propôs em 1950, um teste baseado em um jogo de salão. Neste jogo um computador demonstraria inteligência se um ser humano, ao conversar com outro ser humano e um computador, sem que pudesse vê-los, não conseguisse identificar quem era o humano e quem era o computador.

> O teste, denominado “jogo de Turing”, jogado por um homem, uma mulher e um interrogador que fica em local separado dos outros dois. Ganha o interrogador se descobrir, fazendo perguntas a cada um, quem é homem (Y), e a mulher (X). Ganha a dupla (YX) se conseguir enganar, com suas respostas, o interrogador. Turing propôs como critério de inteligência este jogo em que um dos elementos da dupla é substituído por um computador, que será considerado inteligente se conseguir ganhar o jogo, não dando ao interrogador, durante um tempo razoável, argumento convincente de quem é humano e quem é a máquina (BARRETO, 2001, p.8).

2.4.1 Inteligência Artificial na Medicina

Logo que a informática começou a se tornar acessível ao público, houve entusiasmo do seu uso aplicado para o auxílio de diagnóstico médico. A prática da medicina consiste em uma contínua tomada de decisões, como diagnósticos, prognósticos, proposta terapêutica ou até uma intervenção em nível populacional. Segundo Massad (2004), todos os procedimentos na medicina são baseados em informações, não obra do acaso. As decisões podem ser tomadas em conhecimentos médicos adquiridos de forma cumulativa ao longo do processo de formação do médico.

> Enquanto um médico pode considerar informações de alguns poucos artigos de sua memória ou talvez de algumas dezenas de artigos se ele se utilizar de tecnologias digitais, um supercomputador da IBM chamado Watson pode processar mais de 200 milhões de páginas em poucos segundos. O Watson não está sendo usado para responder a questões médicas, mas sim para avaliar os resultados possíveis mais relevantes considerando os dados de entrada; quem tem a palavra final é o médico. Watson só facilita o trabalho dos médicos, não os substitui. (BERTALAN, 2014, p. 08).

Devido ao acúmulo de informações, a organização das informações em um banco de dados auxilia no registro e acesso à informação sempre que necessário. Na medicina, a prática e experiência profissional fazem do

profissional cada vez melhor. Alguns casos clínicos são extremamente raros e complexos, com o registro acessível a estas particularidades para todos os profissionais médicos, o aprendizado pode se tornar muito mais ágil e menos penoso.

Com literatura acessível na medicina, é relativamente fácil listar quais são os principais sintomas de diversas doenças conhecidas e seus tratamentos. Organizar tais informações com o auxílio de ferramentas computacionais, agiliza o diagnóstico tanto em campo quanto em estudos dirigidos.

Casos médicos quando apresentam uma dificuldade alta e são resolvidos por especialistas no assunto são registrados, documentados e disponibilizados para que todos os demais profissionais saibam o que ocorreu e quais procedimentos foram tomados para obtenção do diagnóstico (FIOCRUZ,2013).

Os países europeus desenvolveram alguns projetos para o uso da IA na saúde. Pode-se citar como um dos grandes e competentes projetos europeus o EDUCTRA, que é um programa de informática avançada na medicina, que estuda lacunas não preenchidas por profissionais da saúde e sugere maneiras de resolve-las, construindo ferramentas que podem facilitar o ensino em diversas áreas de saúde (FERNANDES, 1996).

Segundo Hall (1990 *apud* Vaz; Raposo, 2002), os sistemas tutores são uma composição de diversas disciplinas, como psicologia, ciência cognitiva e inteligência artificial. O principal objetivo destes sistemas é realizar a modelagem e a representação do conhecimento especialista humano para auxiliar o estudante através de um processo interativo.

Programas de Sistemas Tutores quando sofrem interação, conseguem modificar suas bases de conhecimento, “aprendendo” a se adaptar as estratégias de ensino de acordo com o desenrolar do diálogo. A IA aplicada permite também construir um modelo cognitivo de quem interage com ele, facilitando a identificação de vícios e erros, da formulação e comprovação de hipóteses sobre o estilo do aluno, suas ações, o seu nível de conhecimento do assunto e suas estratégias de aprendizagem (VICARI; GIRAFFA, 2003).

2.5 FERRAMENTA WEKA

O pacote de *software* Weka (*Waikato Environment for Knowledge Analysis*) começou a ser desenvolvido em 1993 em linguagem JAVA, na universidade da Nova Zelândia sendo, em 2006 adquirida por uma empresa final. O Weka é um *software* livre, permitindo alteração em seu código fonte. (Mineração de dados com Weka, IBM 2010).

O objetivo do Weka é reunir diversos algoritmos sofisticados provenientes de diferentes abordagens da subárea de inteligência artificial dedicada ao estudo de aprendizagem de máquina. A inteligência artificial estuda o aprendizado da máquina, auxiliando no desenvolvimento de novos algoritmos e técnicas que permitam o computador obter conhecimento.

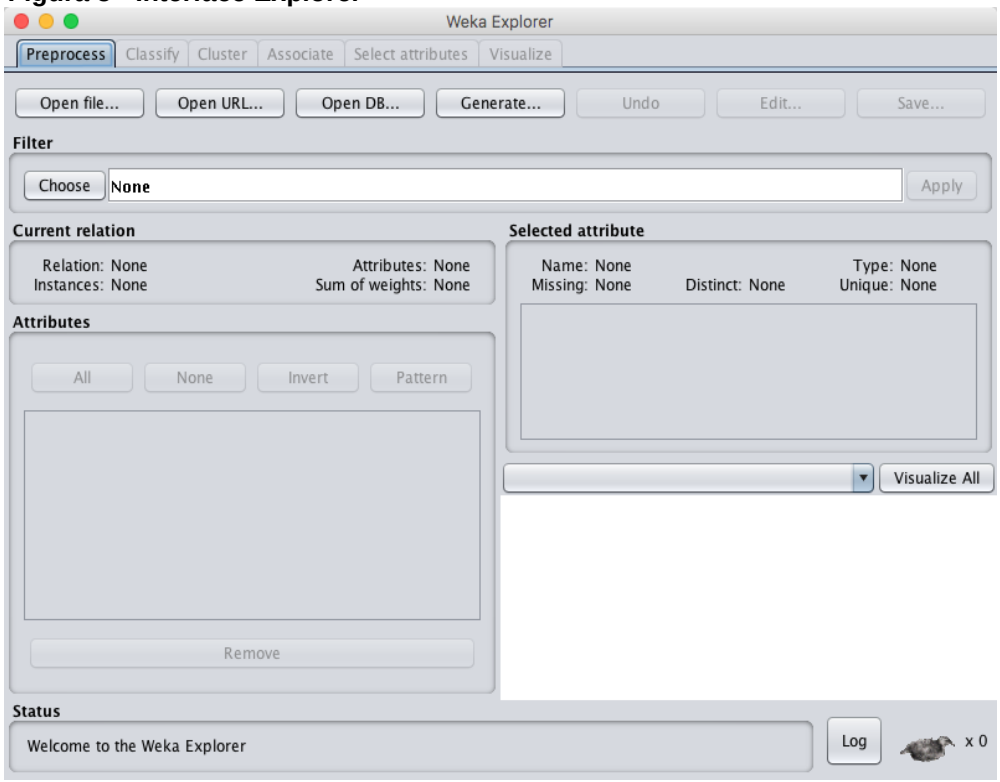
O Weka é uma ferramenta disponível para análise computacional e estatística dos dados recorrendo a técnica de mineração de dados tentando, indutivamente, a partir dos padrões encontrados, gerar soluções hipotéticas e mais profundamente formar teorias sobre os dados estudados de maneira própria. A interface do Weka possui alguns controles selecionáveis, conforme a Figura 4.

Figura 4 – Interface Inicial Software Weka

Fonte: Autoria própria (2017)

Por ser baseado em JAVA, para iniciar o Weka é necessário ter instalado no computador um JRE (*Java Run Escript*). Ao iniciar, o selecionador de GUI é exibido e permite a escolha de quatro modos de trabalho, conforme mostrado na Figura 4. Neste trabalho, a opção utilizada é a *Explorer*, conforme mostrado na Figura 5.

Figura 5 - Interface Explorer



Fonte: Autoria própria (2017)

A interface do Weka é de fácil operação e bem intuitiva. Como mostrado na Figura 5, na aba de *Preprocees* temos os ícones para *Open file*, que abre o arquivo se ele for de uma base de dados local, o *Open URL*, para abrir um arquivo endereçado por URL, o *Open DB* que abre uma *database* e o *Generate*, que auxilia na geração de um arquivo ARFF.

2.5.1 Algoritmos utilizados no trabalho

Os algoritmos utilizados nesse trabalho, que foram detalhados na Seção 2.4.3, são os presentes no Weka baseados em árvore de decisão, são eles:

- DecisionStump
- HoeffdingTree
- J48 (C4.5)
- LMT
- RandomForest
- RandomTree

A representação foi escolhida por ter uma fácil compreensão pelo ser humano na análise de resultados, pois o formato de árvore é de fácil demonstração e explicação. O formato de árvore de decisão auxilia na associação com o cotidiano.

Um dos mais utilizados, o algoritmo em árvore de decisão J48, que surgiu da necessidade de recodificar o algoritmo C4.5, que foi escrito originalmente em linguagem C para a linguagem Java (WITTEN et al., 2005). Sua finalidade é gerar uma árvore de decisão baseada em conjunto de dados de treinamento. O J48 é amplamente utilizado por especialistas em Mineração de Dados pois o mesmo se mostra adequado para procedimentos envolvendo os atributos (dados) qualitativas contínuas e discretas presentes na base de dados (ALVARENGA, M.T, 2014).

Para a montagem da árvore, o J48 utiliza a abordagem “dividir para conquistar”, onde um problema complexo é decomposto em subproblemas mais simples, utilizando de recursividade em cada subproblema, dividindo o espaço definido pelos atributos em subespaços, fazendo associação deles com uma classe (WITTEM; FRANK, 2005). Os algoritmos são aplicados para extrair padrões dos dados, ou gerar regras que descrevam o comportamento da base de dados (BERRY, 1997).

São necessários vários recursos para se promover a descoberta de conhecimento de dados, uma etapa utilizada é a Mineração de Dados. Na mineração de dados são aplicados algoritmos que tem como objetivo identificar padrões nos dados originais, sendo que tais algoritmos se baseiam em técnicas estatísticas, inteligência artificial e complexidade de algoritmos.

- Probabilística: Os algoritmos probabilísticos buscam prever a classe que maximiza a probabilidade de um evento posterior. A principal

tarefa é estimar a probabilidade de cada classe, assumindo a independência dos atributos e, mesmo assim estes classificadores são interessantes em muitas aplicações.

- Árvore de Decisão: Algoritmos baseados em árvore de decisão possuem uma hierarquia de nós que são conectados por ramos, realizando a classificação dos dados em níveis, seguindo os ramos até atingir os nós folha.
- Análise Discriminante Linear: Parte do conhecimento de que os elementos observados pertencem a diversos subgrupos e procura-se determinar funções de todas os atributos observados, que melhor permitam distinguir ou discriminar esses subgrupos ou classes.
- Máquina de Vetor e Suporte: Ocorre a classificação das entradas em duas possíveis classes, o que o torna um classificador linear binário não probabilístico.

A classificação é uma tarefa de mineração de dados que classifica objetos a determinadas classes com o objetivo de prever um novo dado automaticamente.

2.5.2 Estatística KAPPA

A estatística Kappa é uma medida de concordância usada em escalas nominais, que mostra uma ideia do quanto as observações se afastam daquela esperadas, fruto do acaso, mostrando assim o quão confiável as informações mostradas podem ser (FMP, 2017). A magnitude da estatística Kappa é uma medida de concordância mais significativa do que sua própria estatística. Os valores de interpretação do Kappa são mostrados na Tabela 1 a seguir:

Tabela 1 – Valor de concordância Kappa

Valor do Kappa	Concordância
0	Pobre
0 – 0,20	Ligeira
0,21 - 0,40	Considerável
0,41 – 0,60	Moderada
0,61 – 0,80	Substancial
0,81 - 1	Excelente

Fonte: Autoria própria (2017)

Como a Tabela 1 mostra, o valor do Kappa é aceitável quando acima de 0,60, e ele é excelente quando seu valor ultrapassa 0,80.

2.5.3 Matriz de Confusão

Matriz de confusão é uma tabela que valida o aprendizado, indicando quantos acertos e erros foram encontrados na base de dados analisada, mostrando sua acurácia (DIEGONOGARE, 2015).

Na Matriz de confusão, conforme mostrado na Tabela 2 a seguir, os resultados classificados corretamente se encontram na diagonal principal da matriz, e fora dessa diagonal, por conseguinte, as classificações que divergem do esperado.

Tabela 2 – Matriz de confusão – *Demonstração*

Dengue	Zika	Chikungunya
CERTO	errado	errado
errado	**CERTO**	errado
errado	errado	**CERTO**

Fonte: Autoria própria (2017)

Como demonstrado na Tabela 2, independentemente do tamanho da matriz, todos os valores que forem demonstrados na diagonal principal da

mesma são considerados corretos. Esses dados serão importantes na análise dos resultados desse trabalho.

2.5.4 Teoria da Probabilidade

O cálculo da probabilidade é uma ferramenta estatística que estuda os fenômenos aleatórios ou probabilísticos. Esses resultados não podem ser previstos com certeza, mas é possível correlacionar informações e prever a possibilidade da ocorrência. Um conjunto de resultados prováveis pode ser obtido de um espaço amostral, fundamentando uma tomada de decisão mais adequada.

No chamado mundo prático, a maioria das sistemáticas e dos raciocínios envolvem premissas e conclusões incertas. Isto gera, de um lado, a questão de interpretação do conceito de probabilidade e, de outro, o problema para se construir sistemas matemáticos que levem isso em consideração (SOUZA; CAMPELLO, 2004). Alguns conceitos importantes para a aplicação da probabilidade são:

- Elaborar modelos e fenômenos aleatórios,
- Projetar sistemas de inferência e de tomada de decisões,
- Tentativas de se reconhecer os mecanismos cognitivos humanos,
- Afirmar e verificar a aplicabilidade de leis cientificas.

2.6 CONSIDERAÇÕES DO CAPÍTULO

Neste Capítulo foram apresentadas as necessidades encontradas para fazer a descoberta de conhecimento, as etapas e os classificadores que foram usados para estudar os classificadores para o alcançar o objetivo proposto. Neste trabalho, será utilizado suas definições para o desenvolvimento de um sistema que auxilie no diagnóstico médico. Para isso foi selecionado três doenças de alta incidência no Brasil e em países tropicais. São elas dengue, chikungunya e zika. Para se entender um pouco sobre cada uma delas, o

Capítulo 3 mostra as particularidades e similaridades de cada uma dessas doenças.

3. DENGUE, CHIKUNGUNYA E ZIKA

Este Capítulo discorre sobre como os vírus transmitidos pelo Aedes Aegypti pode causar doenças distintas, seus sintomas, diagnósticos e complicações. Este Capítulo foi organizado em quatro Seções, sendo que a 3.1 fala sobre a dengue seus sintomas e diagnósticos. A Seção 3.2 descreve os sintomas e diagnósticos da chikungunya. A Seção 3.3 apresenta as mesmas informações a respeito da zika, seus sintomas e seus diagnósticos. A Seção 3.4 discorre sobre as correlações dos sintomas das três enfermidades estudadas. Por fim, a Seção 3.5 faz um resumo do Capítulo 3.

3.1 DENGUE

A dengue, causada por um vírus pertencente à família Flaviviridae, do gênero Flavivírus, é uma doença infecciosa febril aguda. O Vírus apresenta quatro sorotipos classificados como arbovírus (vírus exclusivamente transmitidos por mosquitos) e são transmitidos pelo mosquito fêmea do Aedes Aegypti (quando também infectada pelo vírus) e podem causar tanto a manifestação clássica da doença quanto a mais grave, hemorrágica (FIOCRUZ, 2013).

3.1.1 Sintomas e Diagnóstico

A dengue pode se manifestar de diferentes formas. Ela pode ser assintomática ou pode evoluir até quadros graves, como a dengue hemorrágica. Na dengue clássica, a primeira manifestação é uma febre alta (39˚ a 40˚C) e de início abrupto, normalmente seguido de dor de cabeça ou olhos, assim como cansaço ou dores musculares e ósseas, náuseas, falta de apetite, vômitos e erupções na pele. A doença pode durar de 5 a 7 dias, mas o período seguinte pode ser acompanhado de grande debilidade física e prolongar-se por várias semanas (FIOCRUZ, 2013).

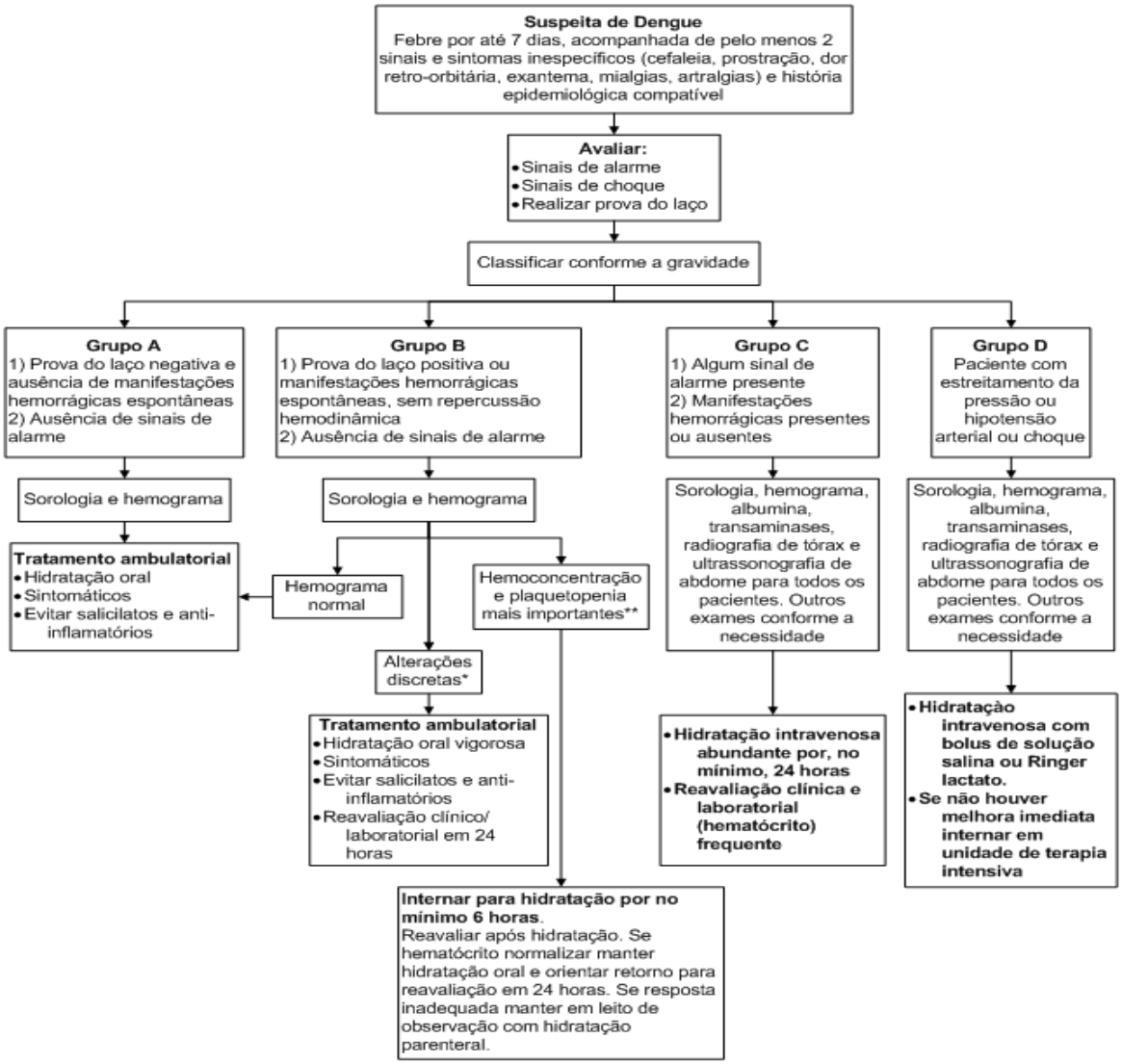
A febre hemorrágica, conhecida como a forma mais grave da doença, possui sintomas iniciais semelhantes, porém, no terceiro ou quarto dia de

evolução há um agravamento no quadro clínico, com aparecimento de manifestações de hemorragias e colapso circulatório. Nos casos mais graves, o choque ocorre geralmente entre o terceiro e sétimo dia de doença, geralmente precedido de dor abdominal (FIOCRUZ, 2013). O choque é decorrente do aumento da permeabilidade vascular, seguida de hemoconcentração, caracterizado por aumento de densidade no sangue e a falência circulatória.

Alguns casos podem apresentar manifestações neurológicas, como convulsões e irritabilidade. Além disso, condições prévias como idade avançada, hipertensão arterial, diabetes, asma brônquica, infecção anterior por outro tipo de dengue e outras doenças respiratórias crônicas graves podem constituir fatores capazes de favorecer a evolução (FIOCRUZ, 2013).

Segundo a Fiocruz (2013), a baixa imunidade do organismo não tem relação com a dengue hemorrágica. As causas mais graves poderiam estar associadas a uma excessiva resposta imunológica do organismo infectado pelo vírus, causando uma espécie de hipersensibilidade que acarretaria na produção de substâncias responsáveis pelo aumento da hipersensibilidade vascular. Tal processo leva a perda de líquidos, o que por sua vez, acarreta na queda da pressão arterial e o choque, principal causa do óbito. Na Figura 6, mostra-se uma diferenciação diagnóstica entre os tipos de dengue existentes. Ressalta-se que nesse trabalho o auxilio no diagnóstico é somente para o identificar a presença da Dengue, e não sua tipologia.

Figura 6 - Diagnósticos para tipos de Dengue



Fonte: Fiocruz (2013)

3.2 CHIKUNGUNYA

A chikungunya, assim como a dengue, é uma doença febril aguda, causada pelo vírus de mesmo nome. Seu modo de transmissão se dá pelos mosquitos Aedes Aegypt e Aedes albopictus.

Os primeiros casos humanos causados pelo vírus da chikungunya (CHIKV) foram relatados no início de 1770, mas o vírus não havia sido isolado do soro humano ou de mosquitos até a epidemia na Tanzânia. Chikungunya significa “aqueles que se dobram”, em um dos idiomas da Tanzânia. Trata-se de

uma referência à aparência curvada dos pacientes que foram atendidos durante a primeira epidemia documentada no país entre 1952-1953 (FIOCRUZ, 2013).

3.2.1 Sintomas

Os principais sintomas da chikungunya são febres altas, acima de 39 graus e de início repentino, dores intensas nas articulações de pés e mãos. Podem ocorrer, também, dor de cabeça, dores musculares e manchas vermelhas na pele. Cerca de 30% das pessoas infectadas não apesentam sinais típicos da doença (FIOCRUZ, 2013).

Os sintomas se manifestam de 2 a 10 dias após a picada do mosquito infectado, podendo chegar a 12 dias. Esse período é chamado de incubação. No caso de uma picada em uma pessoa entre um dia antes do aparecimento da doença até 5 dias após o aparecimento, o mosquito poderá ser infectado também. Casos suspeitos, com febres acima de 38,5˚c repentinas e dores nas articulações de forma intensa devem ser investigadas como possíveis casos da doença. Nos casos de dengue, existe também dores nas articulações, porem a intensidade é menor. Nos casos de chikungunya as dores nas articulações são intensas e concentradas nas mãos e pés, geralmente tornozelo e pulsos (FIOCRUZ, 2013).

3.2.2 Diagnósticos e Tratamentos

O vírus chikungunya só pode ser detectado em exames de laboratório. São três tipos de testes que podem ser realizados: sorologia, isolamento viral e PCR (reação da transcriptase reversa, seguida de reação em cadeia da polimerase) em tempo real (FIOCRUZ, 2013). Em caso de suspeita de chikungunya, deve-se procurar atendimento médico imediatamente e não se automedicar de maneira alguma, pois a automedicação pode mascarar sintomas. Até o momento não existe tratamento específico para a doença.

Os sintomas são tratados com medicação para febre e anti-inflamatórios para as dores articulares. Recomenda-se repouso absoluto ao paciente,

hidratação constante e abundante. As mortes por chikungunya são raras. Internação só em casos mais graves (FIOCRUZ, 2013).

3.3 ZIKA

Zika é uma doença viral aguda, transmitida principalmente por mosquitos, como o Aedes Aegypti, caracterizada por febre intermitente, exantema maculopapular pruriginoso (lesões eruptivas na pele), hiperemia conjuntival não purulenta e sem prurido (semelhante a conjuntivite), dor nas articulações, dor muscular e dor de cabeça. Apresenta evolução benigna e os sintomas desaparecem geralmente após 3 ou 7 dias (FIOCRUZ, 2013).

3.3.1 Sinais e Sintomas

A zika pode ser transmitida por vetores como mosquitos, mas também sua transmissão já foi registrada em ocorrência de transmissão ocupacional em laboratório de pesquisa, perinatal e sexual, além da possibilidade de transmissão transfusional.

Segundo a literatura, uma porcentagem maior que 80% das pessoas infectadas não desenvolvem manifestações clínicas, porém quando presentes suas características principais são exantema maculopapular pruriginoso, febre intermitente, hiperemia conjuntival não purulenta e sem prurido, artralgia, mialgia e dor de cabeça e, com menor frequência, edema, dor de garganta, tosse, vômitos e haematospermia (presença de sangue no esperma ejaculado).

Estudos recentes apontam uma possível correlação entre a infecção por zika e a ocorrência de síndrome de guillain-barré (SGB), uma doença autoimune que ocorre quando o sistema imunológico ataca parte do próprio sistema nervoso por engano, em locais com circulação simultânea do vírus da dengue, porém não confirmada a correlação.

3.3.2 Tratamento

Não existe tratamento específico. Casos sintomáticos são tratados com paracetamol ou dipirona, para controlar febre e manejo da dor. No caso de erupções pruriginosas, os anti-histamínicos podem ser considerados. Não é aconselhado o uso de ácido acetil salicílico (AAS) ou outras drogas anti-inflamatórias em função do devido aumento de risco de hemorragias (FIOCRUZ, 2013).

3.4 SINTOMAS EM COMUM DENGUE, ZIKA E CHIKUNGUNYA

As doenças estudadas possuem muitos sintomas em comum. Estes sintomas foram pesquisados no sitio do ministério da saúde brasileiro, e definidos nas seguintes instâncias:

- Febre repentina;
- Grau de febre;
- Duração da febre;
- Dor nas articulações;
- Inchaço nas articulações;
- Dor de cabeça;
- Dor muscular;
- Coceira na pele;
- Manchas vermelhas na pele;
- Hipertrofia ganglionar;
- Conjuntivite.

O grau da presença desses sintomas combinados pode indicar a presença de dengue, zika ou chikungunya, conforme mostrado nas Figuras 7 e 8:

Figura 7 – Grau de presenças de sintomas por doença 1

Sinais/Sintomas	Dengue	Zika	Chikungunya
Febre (duração)	Acima de 38°C (4 a 7 dias)	Sem febre ou subfebril ≤ 38°C (1-2 dias subfebril)	Febre alta > 38°C (2-3 dias)
Manchas na pele (Frequência)	Surge a partir do quarto dia 30-50% dos casos	Surge no primeiro ou segundo dia 90-100% dos casos	Surge 2-5 dia 50% dos casos
Dor nos músculos (Frequência)	+++/+++	++/+++	+/+++
Dor na articulação (frequência)	+/+++	++/+++	+++/+++
Intensidade da dor articular	Leve	Leve/Moderada	Moderada/Intensa
Edema da articulação	Raro	Frequente e leve intensidade	Frequente e de moderada a intenso
Conjuntivite	Raro	50-90% dos casos	30%
Cefaleia (Frequência e intensidade)	+++	++	++
Prurido	Leve	Moderada/Intensa	Leve
Hipertrofia ganglionar (frequência)	Leve	Intensa	Moderada
Discrasia hemorrágica (frequência)	Moderada	ausente	Leve
Acometimento Neurológico	Raro	Mais frequente que Dengue e Chikungunya	Raro (predominante em Neonatos)

Fonte: Carlos Brito – Professor da Universidade Federal de Pernambuco (atualização em dezembro/2015)

Fonte: Universidade Federal de Pernambuco (2015)

Na Figura 8 a seguir temos mais uma referência desses sintomas:

Figura 8 – Grau de presenças de sintomas por doença 2

Fonte: Ministério da saúde (2017)

Com base nesses sintomas presentes nas três enfermidades estudadas, foi definido que essas instâncias que serão utilizadas na geração do *dataset* sobre as doenças estudadas para a realização dos testes dos classificadores.

3.5 CONSIDERAÇÕES DO CAPÍTULO

Este Capítulo apresentou as principais características de cada uma das doenças selecionadas para estudo de caso nos classificadores. Com essas informações é possível registrar seus sintomas e correlações para que se chegue a um diagnóstico. Essas informações são importantes para que se forme regras relacionais, que serão implementadas. Também foi mostrado os sintomas selecionados para a criação do *dataset* que foi analisado neste trabalho. No Capítulo a seguir mostra-se como é composto o arquivo ARFF que será gerado e analisado nos classificadores.

4. COMPOSIÇAO DO ARQUIVO ARFF E GERACAO DO DATASET

Esse Capítulo apresenta toda a composição de um arquivo ARFF e como os atributos nele presente devem ser declarados. Este arquivo foi submetido a análises no Weka *software*, para a geração de conhecimento. É relatada a sua criação, seus algoritmos de análise e o algoritmo escolhido para a análise de dados. Esse Capítulo está organizado em duas Seções, a 4.1 que ela fala sobre o *arquivo ARFF*, sua composição e também exemplifica o código em linguagem *Pyton* que foi criado e executado posteriormente para formar o *dataset* a ser analisado. A Seção 4.2 discorre sobre as considerações tratadas nesse Capítulo.

4.1 ARQUIVOS ARFF

ARFF (*Attribute-Relation File Format*), é um arquivo de texto ASCII que descreve uma lista de instâncias que compartilham entre si um conjunto de atributos. Arquivos ARFF foram desenvolvidos pelo projeto de aprendizado de máquina no departamento de Ciência da Computação da Universidade de Waikato para uso com o *software* de aprendizado de máquina Weka.

Os arquivos ARFF possuem duas sessões distintas. A Primeira sessão é o *Header*, que contém informações que identificam sobre o que os dados vão tratar, geralmente um título, local onde foi obtido os dados e autor(es), seguido pelo DATA, onde contém os dados a serem analisados (WIKI, 2008). O *Header* contém o nome da relação, a lista de atributos e seus tipos.

Para este trabalho, foram utilizados os 11 atributos previsores definidos no Capítulo 3 desse trabalho e um atributo classe. Todos estão identificados a seguir, ressaltando que o atributo duração-febre está com unidade de medida em dias. Os atributos funcionam como chave e a ele pode ser atribuído um ou mais valores que formarão os dados. Um exemplo de um *Header* de um arquivo ARFF pode ser visto na Figura 9.

Figura 9 – HEADER de um arquivo ARFF

Dengue_chikungunya_Zika.arff

```
% Citation Request:
%    Casos de doenças Dengue, Chikungunya e Zika, extraídos da internet em casos
%    Simulados.
% 1. Title: Diagnóstico de Doenças, Sistema Especialista
% criado por William Lima
% análise para trabalho de conclusão de curso
%UTFPR-PG 2017 curso de ADS
%
@relation Dengue_Chikungunya_Zika
@attribute idade {'0-5', '6-14', '15-49', '50-99'}
@attribute sexo {'masc','fem'}
@attribute febre-repentina {'sim', 'nao'}
@attribute febre {'37-37.9','38-40'}
@attribute duracao-febre{'0-2','2-3','2-7'}
@attribute dor-articulacao {'leve','moderada','intensa'}
@attribute inchaco-articulacao{'sim', 'nao', 'raro'}
@attribute dor-de-cabeca {'ausente','leve','moderada','intensa'}
@attribute dor-muscular {'ausente','moderada', 'intensa'}
@attribute coceira {'ausente','leve','moderada','intensa'}
@attribute manchas-vermelhas {'sim','nao'}
@attribute hipertrofia-ganglionar {'sim','nao'}
@attribute conjuntivite {'sim', 'nao'}
@attribute 'Class' {'dengue','zika', 'chikungunya'}
@data
```

Fonte: Autoria própria (2017)

A seguir, na Figura 10, tem-se um exemplo dos dados que são analisados pelo Weka.

Figura 10 – DATA de um arquivo ARFF

```
@data
'15-49','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'0-5','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'15-49','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'0-5','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'6-14','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'6-14','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'50-99','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'15-49','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'50-99','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'6-14','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'50-99','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'6-14','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'0-5','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'0-5','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'6-14','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'50-99','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'0-5','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'6-14','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'50-99','fem','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
'6-14','masc','nao','38-40','2-7','intensa','raro','intensa','intensa','leve','sim','nao','nao','dengue'
```

Fonte: Autoria própria (2017)

4.1.1 Seção HEADER de um arquivo ARFF

A Seção *HEADER* (Cabeçalho) do arquivo deve conter a declaração de relação e a declaração de atributos. Ao declarar a relação usa-se o @ (arroba) seguido da declaração desejada, por exemplo @*RELATION* <nome-da-relação>lembrando que o <nome-da-relação> é do tipo *String*.

4.1.2 Declaração de atributos de um arquivo ARFF

As declarações de atributos devem ser feitas cada um em uma linha distinta, para a construção correta do arquivo ARFF. A declaração deve ser feita com o sinal de @ antecedendo o atributo, seguido pelo seu nome e pelo tipo do dado, como no exemplo a seguir: @*ATTRIBUTE*<nome-atributo> <tipo de dado>.

Cada atributo do conjunto de dados tem sua própria declaração @*attribute* que define de forma exclusiva o nome desse atributo e seu tipo de dado. É muito importante a ordem em que os atributos são declarados, pois isso indica a posição da coluna na seção de dados do arquivo. Por exemplo, se um atributo for o terceiro declarado, o Weka espera que todos esses valores de atributos sejam encontrados na terceira coluna delimitada por vírgulas (WIKI, 2008).

Os tipos de dado suportados pelo Weka são:

- *numeric*
- <*nominal-specification*>
- *String*
- *date* [<*date-format*>]

As palavras reservadas *number*, *string* e *date* não são *case sensitive*. Os atributos numéricos podem ser *real* ou *integer*. Os atributos nominais são definidos fornecendo uma <especificação nominal> listando os valores possíveis: {nome-nome1>, <nome-nome2>, ...}.

No exemplo a seguir, o valor de classe do conjunto de dados pode ser definido da seguinte forma: @*Class* {“dengue”, “zika”, “chikungunya”}.

Os atributos do tipo *STRING* permitem criar atributos que contenham valores de texto arbitrário. Isso é muito útil nas aplicações de mineração de texto, pois é possível criar conjunto de dados com vários atributos de *String* e em seguida fixar seu valor em um único atributo para facilitar sua busca através de *Filters* para manipular *Strings* (como *StringToWordVectorFilter*). Os atributos *String* são declarados da seguinte maneira: *@ATTRIBUTE* ABC *string*.

Os atributos de data são declarados da seguinte maneira: *@ATTRIBUTE* <nome> *date*[<d*ate-format*>].

Onde <nome> é o nome do atributo e <*date-format*> é uma *string* opcional que específica como os valores de data devem ser analisados e impressos. A *string date* aceita o formato padrão de data e hora combinados ("yyyy-MM-dd'T'HH:mm:ss").

4.1.3 Declaração de dados de um arquivo ARFF

No arquivo ARFF na seção dados, devem conter a linha de declaração de dados e as linhas de instâncias a serem analisadas. A declaração é simplesmente @data em uma única linha, indicando o início do seguimento de dados, como no exemplo a seguir:

@DATA

'15-49','fem','nao','38-40','2-7','raro','intensa','intensa','leve','nao','dengue' Cada instância é representada em uma linha. Os valores dos atributos para cada instância são delimitados por vírgula. Eles devem seguir a sequência que foram declarados na seção *Header*. Como no exemplo acima, caso não tenha as informações completas da instância, ela deve ser representada na ordem com o símbolo “?” (WIKI, 2008).

4.1.4 Gerando os dados do arquivo ARFF

Para gerar os dados que foram analisados no Weka, foi criado um código usando a linguagem *Pyton.* Ele permite gerar quantas instâncias se quer analisar. Com base na literatura médica, foi simulado sintomas mais presentes

em cada uma das doenças estudadas, gerando um código específico para cada enfermidade. Os parâmetros são os sintomas que podem aparecer em cada diagnóstico e a classe a doença ao qual o grupo pertence. Gera-se o arquivo *ARFF* que posteriormente foi analisado para obtenção dos resultados. O código completo se encontra na seção seguinte, ele foi executado no programa *PyCharm.*

Para a geração dos dados para dengue foi utilizado o código mostrado nas Figuras 11 e 12:

Figura 11 – Código geração de dados Dengue

mainDengue.py | mainChikungunya.py | mainZika.py

```
1   import random
2   import pandas as pd
3   import csv
4
5   # Pandas Configuration
6   pd.set_option('display.max_rows', 500)
7   pd.set_option('display.max_columns', 500)
8   pd.set_option('display.width', 1000)
9
10  idade = ['0-5', '6-14', '15-49', '50-99']
11  sexo = ['masc','fem']
12  febre_repentina = ['sim', 'nao']
13  febre = ['37-37.9','38-40']
14  duracao_febre= ['0-2','2-7','8-14']
15  dor_articulacao = ['leve','moderada','intensa']
16  inchaco_articulacao= ['sim', 'nao', 'raro']
17  dor_de_cabeca = ['ausente','leve','moderada','intensa']
18  dor_muscular = ['ausente','moderada', 'intensa']
19  coceira = ['ausente','leve','moderada','intensa']
20  manchas_vermelhas = ['sim','nao']
21  hipertrofia_ganglionar = ['sim','nao']
22  conjuntivite = ['nao']
23  Class = ['dengue','zika', 'chikungunya']
24
25  ficha = []
26
27  for f in range(500):
28
29      v_febre_repentina = 'nao'
30      v_febre = '38-40'
31      v_duracao_febre = '2-7'
32      v_dor_articulacao = 'intensa'
33      v_inchaco_articulacao = 'raro'
34      v_dor_de_cabeca = 'intensa'
35      v_dor_muscular = 'intensa'
36      v_coceira = 'leve'
37      v_manchas_vermelhas = 'sim'
38      v_hipertrofia_ganglionar = 'nao'
39      v_conjuntivite = 'nao'
40      v_class = 'dengue'
```

Fonte: Autoria própria (2017)

Figura 12 – Código geração de dados Dengue - Continuação

```
41
42
43      ficha.append({"idade": random.choice(idade),
44                    "sexo": random.choice(sexo),
45                    "febre_repentina": v_febre_repentina,
46                    "febre": v_febre,
47                    "duracao_febre": v_duracao_febre,
48                    "dor_articulacao": v_dor_articulacao,
49                    "inchaco_articulacao": v_inchaco_articulacao,
50                    "dor_de_cabeca": v_dor_de_cabeca,
51                    "dor_muscular": v_dor_muscular,
52                    "coceira": v_coceira,
53                    "manchas_vermelhas": v_manchas_vermelhas,
54                    "hipertrofia_ganglionar": v_hipertrofia_ganglionar,
55                    "conjuntivite": v_conjuntivite,
56                    "class": v_class})
57
58
59  ficha = pd.DataFrame(ficha)
60
61  ficha = ficha.reindex_axis(["idade", "sexo", "febre_repentina", "febre",
62                              "duracao_febre", "dor_articulacao", "inchaco_articulacao",
63                                 "dor_de_cabeca", "dor_muscular", "coceira", "manchas_vermelhas",
64                                    "hipertrofia_ganglionar","conjuntivite", "class"], axis=1)
65
66
67
68  ficha.to_csv("ds_sintomas_dengue.csv", sep=',', encoding='utf-8',
69               index=False, quoting=csv.QUOTE_NONNUMERIC, quotechar="'", header = False)
70
71
```

Fonte: Autoria própria (2017)

Para a geração dos dados para chikungunya foi utilizado o código mostrado nas Figuras 13 e 14:

Figura 13 – Código geração de dados Chikungunya

```
import random
import pandas as pd
import csv

# Pandas Configuration
pd.set_option('display.max_rows', 500)
pd.set_option('display.max_columns', 500)
pd.set_option('display.width', 1000)

idade = ['0-5', '6-14', '15-49', '50-99']
sexo = ['masc','fem']
febre_repentina = ['sim', 'nao']
febre = ['37-37.9','38-40']
duracao_febre= ['0-2','2-3','2-7','8-14']
dor_articulacao = ['moderada','intensa']
inchaco_articulacao= ['sim', 'nao', 'raro']
dor_de_cabeca = ['ausente','leve','moderada','intensa']
dor_muscular = ['ausente','moderada', 'intensa']
coceira = ['ausente','leve','moderada','intensa']
manchas_vermelhas = ['sim','nao']
hipertrofia_ganglionar = ['sim','nao']
conjuntivite = ['sim', 'nao']
Class = ['dengue','zika', 'chikungunya']

ficha = []

for f in range(500):

    v_febre_repentina = 'nao'
    v_febre = '38-40'
    v_duracao_febre = '2-3'
    v_dor_articulacao = random.choice(dor_articulacao)
    v_inchaco_articulacao = 'sim'
    v_dor_de_cabeca = 'moderada'
    v_dor_muscular = 'intensa'
    v_coceira = 'leve'
    v_manchas_vermelhas = random.choice(manchas_vermelhas)
    v_hipertrofia_ganglionar = 'nao'
    v_conjuntivite = random.choice(conjuntivite)
    v_class = 'chikungunya'
```

Fonte: Autoria própria (2017)

Figura 14 – Código geração de dados Chikungunya - Continuação

```
41
42
43      ficha.append({"idade": random.choice(idade),
44                    "sexo": random.choice(sexo),
45                    "febre_repentina": v_febre_repentina,
46                    "febre": v_febre,
47                    "duracao_febre": v_duracao_febre,
48                    "dor_articulacao": v_dor_articulacao,
49                    "inchaco_articulacao": v_inchaco_articulacao,
50                    "dor_de_cabeca": v_dor_de_cabeca,
51                    "dor_muscular": v_dor_muscular,
52                    "coceira": v_coceira,
53                    "manchas_vermelhas": v_manchas_vermelhas,
54                    "hipertrofia_ganglionar": v_hipertrofia_ganglionar,
55                    "conjuntivite": v_conjuntivite,
56                    "class": v_class})
57
58
59  ficha = pd.DataFrame(ficha)
60
61  ficha = ficha.reindex_axis(["idade", "sexo", "febre_repentina", "febre",
62                              "duracao_febre", "dor_articulacao", "inchaco_articulacao",
63                                  "dor_de_cabeca", "dor_muscular", "coceira", "manchas_vermelhas",
64                                      "hipertrofia_ganglionar", "conjuntivite", "class"], axis=1)
65
66
67
68  ficha.to_csv("ds_sintomas_chikungunya.csv", sep=',', encoding='utf-8', index=False,
69                  quoting=csv.QUOTE_NONNUMERIC, quotechar="'", header = False)
70
71
```

Fonte: Autoria própria (2017)

Para a geração dos dados para zika foi utilizado o código mostrado nas Figuras 15 e 16:

Figura 15 – Código geração de dados Zika

```
import random
import pandas as pd
import csv

# Pandas Configuration
pd.set_option('display.max_rows', 500)
pd.set_option('display.max_columns', 500)
pd.set_option('display.width', 1000)

idade = ['0-5', '6-14', '15-49', '50-99']
sexo = ['masc','fem']
febre_repentina = ['sim', 'nao']
febre = ['37-37.9','38-40']
duracao_febre= ['0-2','2-3','2-7','8-14']
dor_articulacao = ['moderada','intensa']
inchaco_articulacao= ['sim', 'nao', 'raro']
dor_de_cabeca = ['ausente','leve','moderada','intensa']
dor_muscular = ['ausente','moderada', 'intensa']
coceira = ['moderada','intensa']
manchas_vermelhas = ['sim','nao']
hipertrofia_ganglionar = ['sim','nao']
conjuntivite = ['sim', 'nao']
Class = ['dengue','zika', 'chikungunya']

ficha = []

for f in range(500):

    v_febre_repentina = 'nao'
    v_febre = '37-37.9'
    v_duracao_febre = '0-2'
    v_dor_articulacao = random.choice(dor_articulacao)
    v_inchaco_articulacao = 'sim'
    v_dor_de_cabeca = 'moderada'
    v_dor_muscular = 'intensa'
    v_coceira = random.choice(coceira)
    v_manchas_vermelhas = 'sim'
    v_hipertrofia_ganglionar = 'sim'
    v_conjuntivite = 'sim'
    v_class = 'zika'
```

Fonte: Autoria própria (2017)

Figura 16 – Código geração de dados Zika - Continuação

```
41
42
43      ficha.append({"idade": random.choice(idade),
44                    "sexo": random.choice(sexo),
45                    "febre_repentina": v_febre_repentina,
46                    "febre": v_febre,
47                    "duracao_febre": v_duracao_febre,
48                    "dor_articulacao": v_dor_articulacao,
49                    "inchaco_articulacao": v_inchaco_articulacao,
50                    "dor_de_cabeca": v_dor_de_cabeca,
51                    "dor_muscular": v_dor_muscular,
52                    "coceira": v_coceira,
53                    "manchas_vermelhas": v_manchas_vermelhas,
54                    "hipertrofia_ganglionar": v_hipertrofia_ganglionar,
55                    "conjuntivite": v_conjuntivite,
56                    "class": v_class})
57
58
59  ficha = pd.DataFrame(ficha)
60
61  ficha = ficha.reindex_axis(["idade", "sexo", "febre_repentina", "febre",
62                              "duracao_febre", "dor_articulacao", "inchaco_articulacao",
63                                 "dor_de_cabeca", "dor_muscular", "coceira", "manchas_vermelhas",
64                                    "hipertrofia_ganglionar","conjuntivite", "class"], axis=1)
65
66
67
68  ficha.to_csv("ds_sintomas_zika.csv", sep=',', encoding='utf-8', index=False,
69               quoting=csv.QUOTE_NONNUMERIC, quotechar="'", header = False)
70
71
```

Fonte: Autoria própria (2017)

Após a geração dos dados, eles foram anexados no *Header* do arquivo ARFF e analisado no *software* Weka para a obtenção dos resultados.

4.2 CONSIDERAÇÕES DO CAPÍTULO

Este Capítulo apresentou a composição do arquivo ARFF e todas as suas estruturas e declarações que formam um arquivo atingindo todos os requisitos necessários para sua leitura no Weka *software*. Utilizando um código feito em linguagem *Pyton*, foi gerado o *dataset* que foi testado para a obtenção dos resultados deste trabalho. O código exemplificado pode ser reutilizado em diversas ocasiões, mas no caso do trabalho foi feito para gerar dados que foram

analisados pelo Weka. No Capítulo 5 são mostrados os resultados obtidos após análise.

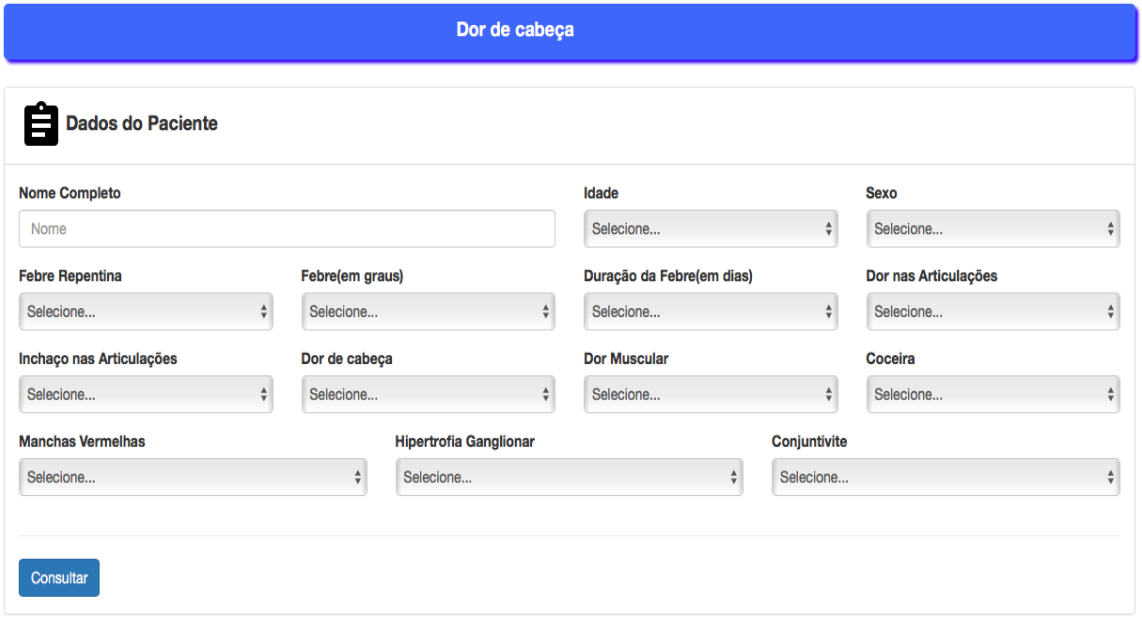
5. RESULTADOS

Este Capítulo mostra os resultados obtidos através dos dados analisados no Weka, sua precisão e confiabilidade. Organizado em 5 Seções, onde a 5.1 discorre sobre a interface de interação com o usuário, seguida pela Seção 5.2 que fala sobre a entrada de dados utilizada para análise. A Seção 5.3 mostra os resultados de análises de todos os algoritmos de árvore de decisão presentes no Weka progredindo na Seção 5.4 para um comparativo entre eles. A Seção 5.5 encerra o Capítulo discorrendo sobre o algoritmo que se mostrou mais apropriado para o uso proposto.

5.1 INTERFACE INTERATIVA

A interface que serve de interação com o usuário foi pensada para ser uma ficha de pré-atendimento clínico. Desenvolvida utilizando *bootstrap* e NodeJs que serão detalhados nas subseções seguintes. Na *interface* é necessário realizar o cadastro do paciente e, selecionando sintomas inseridos na base de dados o sistema salva essas informações, gerando mais dados para uma futura consulta e simultaneamente faz uma triagem, sugerindo a doença mais provável para os sintomas apresentados. A Figura 17 a seguir ilustra a ideia:

Figura 17 – Ficha de cadastro e pré-seleção de sintomas



Fonte: Autoria própria (2017)

Após a entrada de informações do paciente e dos sintomas definidos previamente no sistema e são acessadas através de um botão *dropdown*, é exibido um resumo dos sintomas inseridos, feita uma consulta nos dados integrados com o sistema Weka através de sua API, e retorna o mais provável caso a ser tratado, como podemos ver na Figura 18.

Figura 18 – Ficha com apresentação de provável diagnóstico

Fonte: Autoria própria (2017)

A interface gráfica serve de interação entre o usuário e o sistema que consulta a API do Weka e traz os resultados, possibilitando a agilidade na inserção de informações e na confiabilidade de um diagnóstico com índice de assertividade alto, auxiliando o usuário com a informação obtida.

5.1.1 Bootstrap

O Bootstrap é um *framework* de auxílio na construção de sites que possui padrões de códigos gerados automaticamente e que podem ser reutilizados. Na construção do *front-end*, ou seja, a parte responsável na coleta dos dados do usuário e processa-las adequando a uma forma específica que será utilizada no *back-end (*parte interna*).*

Desenvolvido por Mark Otoo e Jacob Thornton no *Twitter*, era chamado no início de *Twitter Blueprint*, como uma estrutura para incentivar a consistência entre ferramentas internas. Antes de ter o Bootstrap eram usadas várias bibliotecas distintas para o desenvolvimento do *front-end (BOOTSTRAP, 2017).*

Modular*,* o Bootstrap consiste em uma série de folhas de *Less*, que é uma linguagem de estilo de folhas dinâmico que sua compilação pode ser feita em cascatas (CSS) e é executado tanto do lado do cliente quanto do servidor, que implementam os vários componentes do conjunto de ferramentas.

5.1.2 Node JS

NodeJS é uma plataforma para desenvolvimento de aplicações do lado do servidor, baseadas em rede utilizando *JavaScript* e o *V8 JavaScript Engine*. NodeJs é uma ferramenta que foi utilizada para fazer a integração do *frontend* com o *backend.*

De acordo com a NodeJs Foundation (2017), o motor V8 aumenta o desempenho do Node consideravelmente por eliminar intermediários e compilar diretamente o código para linguagem de máquina.

O NodeJs permite atualizar a linguagem *JavaScritpt* para servidor em modelo assíncrono, e projetado para criar aplicativos escaláveis e de alta performance (NODEJS, 2017). Desde sua publicação em 2009 por Ryan Dahl, o NodeJS vem conquistando espaço como ferramenta para desenvolvimento de aplicações web (NODEJS, 2017).

Uma aplicação Node é executada como um laço contínuo em um único processo e fica aguardando por eventos, como por exemplo requisições

(NODEJS, 2017). Essa característica difere o Node de outras aplicações de modelos de programação, os quais executam cada conexão em um novo processo (NODEJS, 2017).

No *site* Weka (Weka.wikispaces.com), temos as informações necessárias para a realização da integração via *Javascript* do NodeJs com a API fornecida pelo Weka. Na Figura 19, temos um exemplo padrão de como utilizar a API do Weka.

Figura 19 – Importação API Weka

```
import weka.core.Instances;
import java.io.BufferedReader;
import java.io.FileReader;
...
BufferedReader reader = new BufferedReader(
                            new FileReader("/some/where/data.arff"));
Instances data = new Instances(reader);
reader.close();
// setting class attribute
data.setClassIndex(data.numAttributes() - 1);
```

Fonte: Adaptado Weka.Wikispace (2017)

Neste trabalho, foi criado um servidor com o NodeJS no próprio computador, acessado como *localhost*, fazendo a consulta no Weka e retornando o resultado provável.

5.2 SOBRE OS DADOS

Os dados analisados neste trabalho foram gerados utilizando parâmetros reais encontrados nas enfermidades estudadas. Utilizando um código na linguagem *Pyton e* posteriormente anexado a um *HEADER* do arquivo ARFF. Foi solicitado ao governo brasileiro, junto ao órgão da Agência Nacional de Saúde (ANS), os dados reais, porém devido a alguns problemas relacionados ao prazo não foi possível incluí-los nesse trabalho.

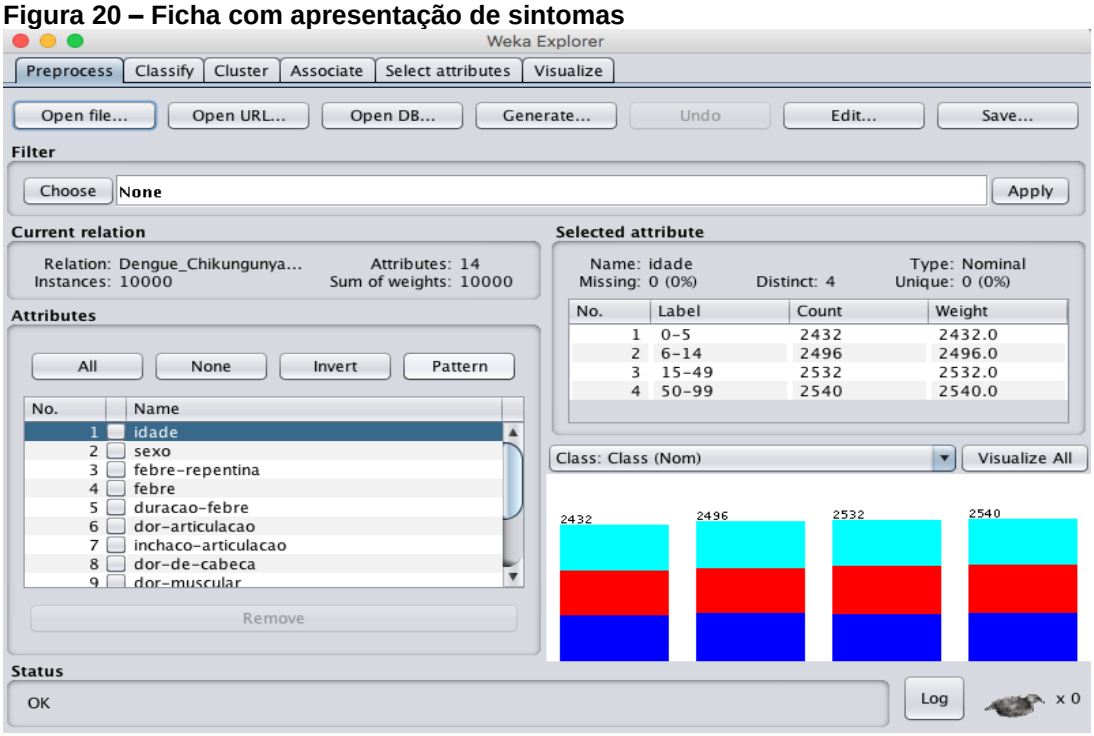
5.3 RESULTADO DOS CLASSIFICADORES DE ÁRVORES DE DECISÃO

O *software* Weka usado para esse trabalho possui 7 classificadores distintos que utilizam árvore de decisão. São eles:

- *DecisionStump*
- *HoeffdingTree*
- *J48*
- *LMT*
- *RandomForest*
- *RandonTree*
- *REPTree*

Suas particularidades não são tratadas nesse trabalho, porém a base de dados foi submetida a análise de cada um deles, utilizando *Cross-validation* com 10 *folds*. Para um algoritmo ser classificado como bom, a estatística Kappa, que é uma medida de concordância interobservador, e mede o grau de concordância além do que seria esperado tão somente pelo acaso, deve estar próximo a 0.8 ou acima desse valor. Os dados que serão analisados contêm 10.000 instâncias distintas, conforme mostrado na Figura 20 a seguir. Os resultados obtidos são exibidos em sequência.

Figura 20 – Ficha com apresentação de sintomas

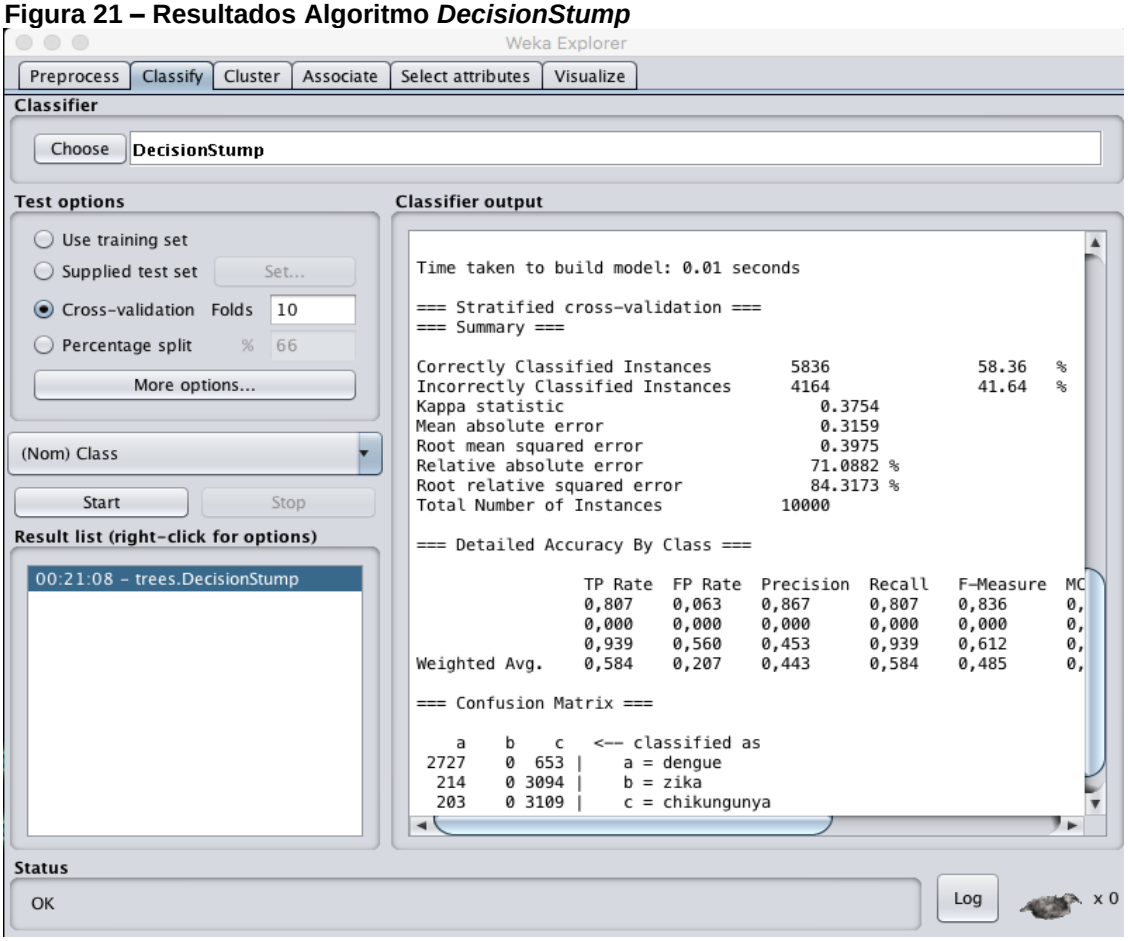


Fonte: Autoria própria (2017)

5.3.1 Algoritmo DecisionStump

O resultado do algoritmo *DecisionStump* possui uma taxa de acerto baixa e sua estatística Kappa 0.3754, não se mostrando um bom classificador para o que se espera, conforme mostrado na Figura 21.

Figura 21 – Resultados Algoritmo *DecisionStump*



Fonte: Autoria própria (2017)

A Matriz de confusão do classificador *DecisionStump,* que mostra na diagonal principal os elementos classificados corretamente e destacados em negrito, é exibida a seguir na Tabela 3.

Tabela 3 – Matriz de confusão – *DecisionStump*

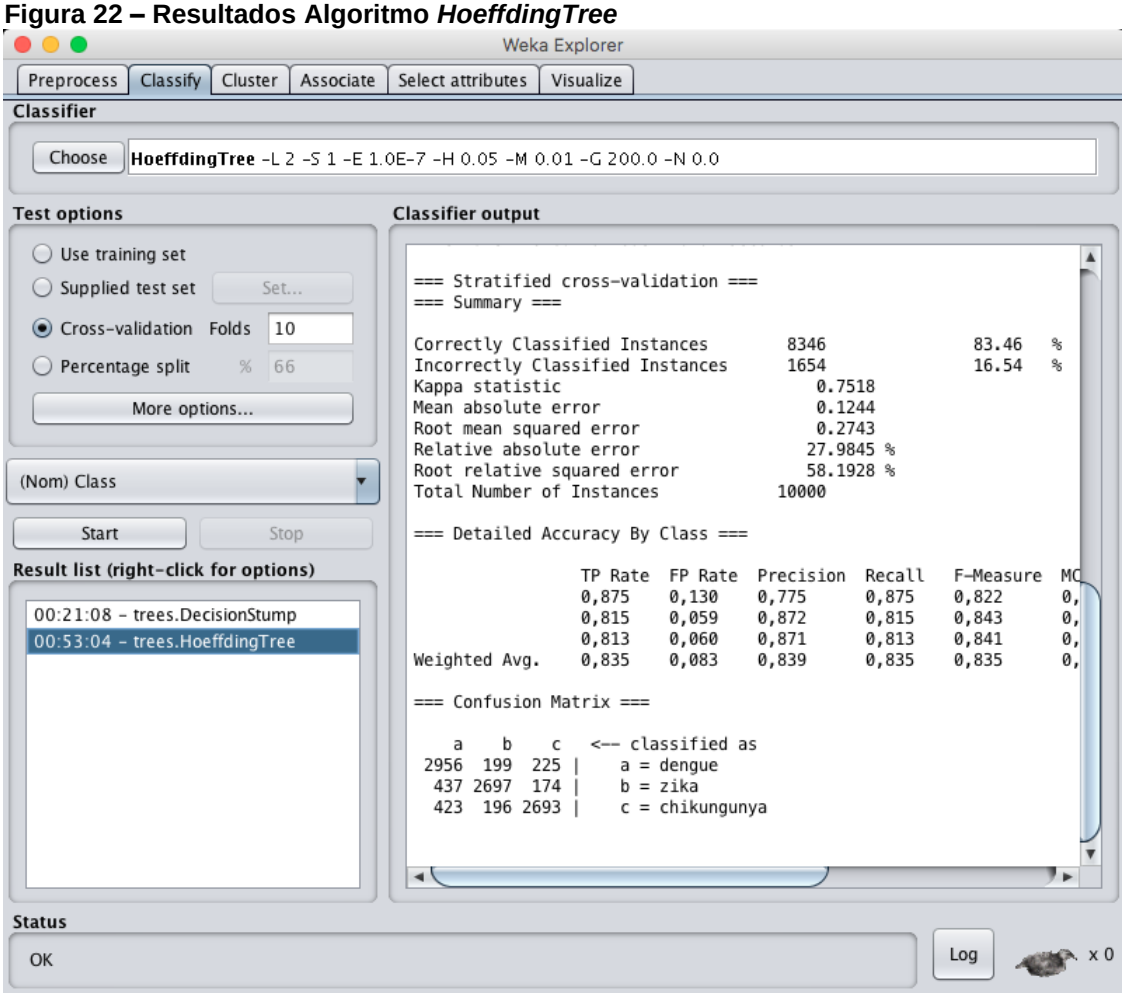
Dengue	Zika	Chikungunya
2727	0	653
214	**0**	3094
203	0	**3109**

Fonte: Autoria própria (2017)

5.3.2 Algoritmo HoeffdingTree

O resultado do algoritmo *HoeffdingTree* possui uma taxa de acerto alta, de 83.46% e sua estatística Kappa 0.7518, se mostrando um bom classificador para o que se espera, conforme mostrado na Figura 22.

Figura 22 – Resultados Algoritmo *HoeffdingTree*



Fonte: Autoria própria (2017)

A Matriz de confusão do classificador *HoeffdingTree,* que mostra na diagonal principal os elementos classificados corretamente e destacados em negrito, é exibida a seguir na Tabela 4.

Tabela 4 – Matriz de confusão – *HoeffdingTree*

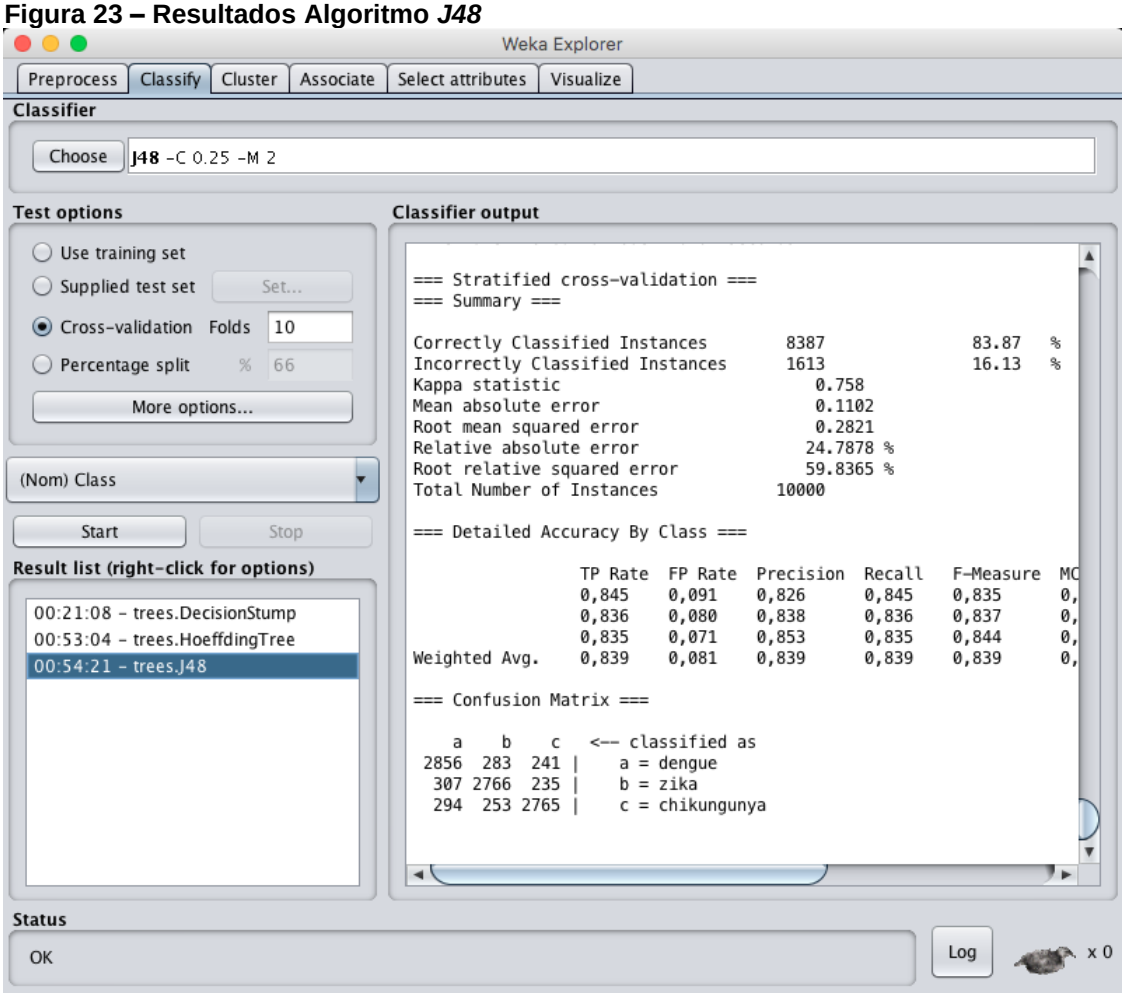
Dengue	Zika	Chikungunya
2956	199	225
437	**2697**	174
423	196	**2693**

Fonte: Autoria própria (2017)

5.3.3 Algoritmo J48

O resultado do algoritmo *J48* possui uma taxa de acerto alta, de 83,87% e sua estatística Kappa 0.758, se mostrando um bom classificador para o que se espera, conforme mostrado na Figura 23.

Figura 23 – Resultados Algoritmo *J48*



Fonte: Autoria própria (2017)

A matriz de confusão do classificador *J48* é exibida na Tabela 5.

Tabela 5 – Matriz de confusão – *J48*

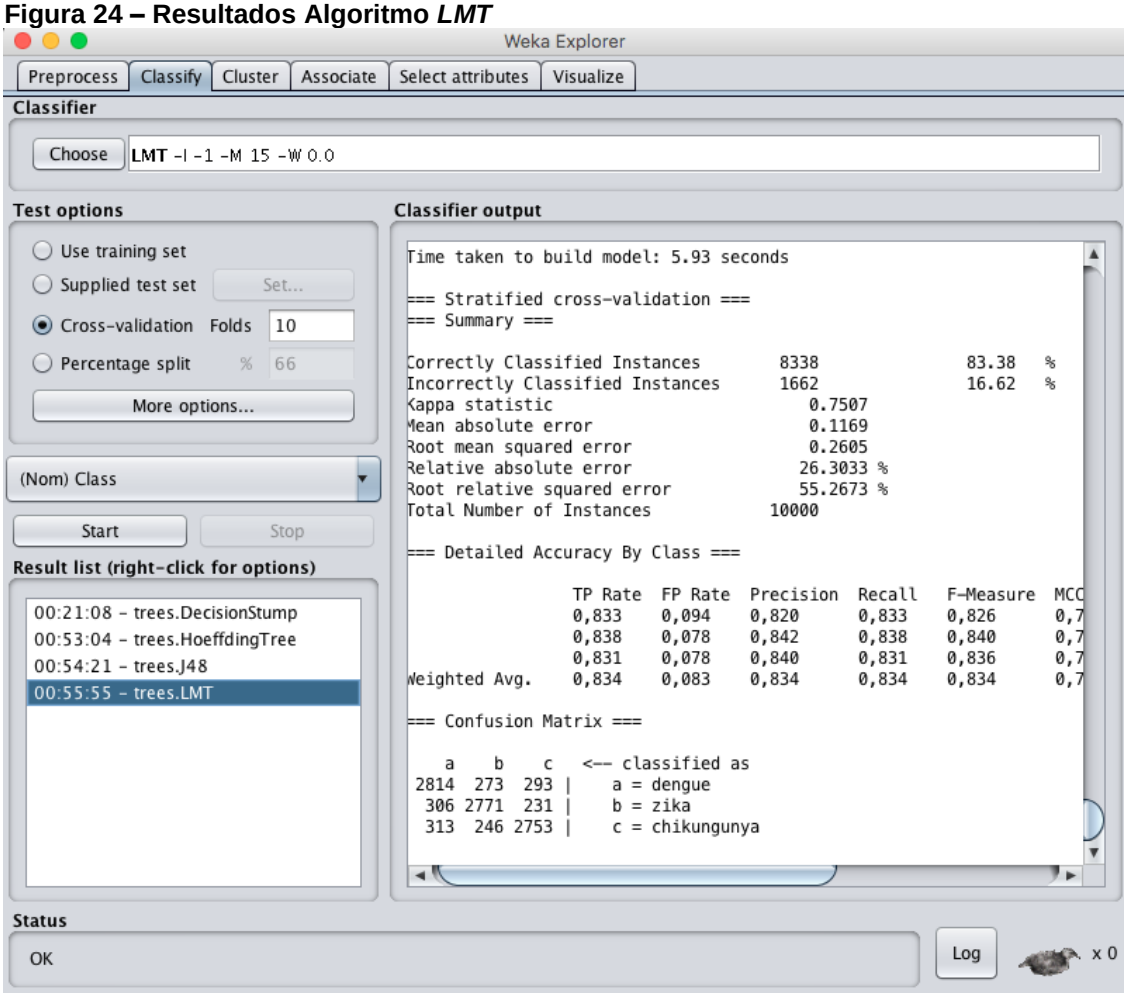
Dengue	Zika	Chikungunya
2856	283	241
307	**2766**	235
294	253	**2765**

Fonte: Autoria própria (2017)

5.3.4 Algoritmo LMT

O resultado do algoritmo *LMT* possui uma taxa de acerto alta, de 83,38% e sua estatística Kappa 0.7507, se mostrando um bom classificador para o que se espera, conforme mostrado na Figura 24.

Figura 24 – Resultados Algoritmo *LMT*



Fonte: Autoria própria (2017)

A matriz de confusão do classificador *LMT*, que mostra na diagonal principal os elementos classificados corretamente e destacados em negrito, é exibida a seguir na Tabela 6.

Tabela 6 – Matriz de confusão – *LMT*

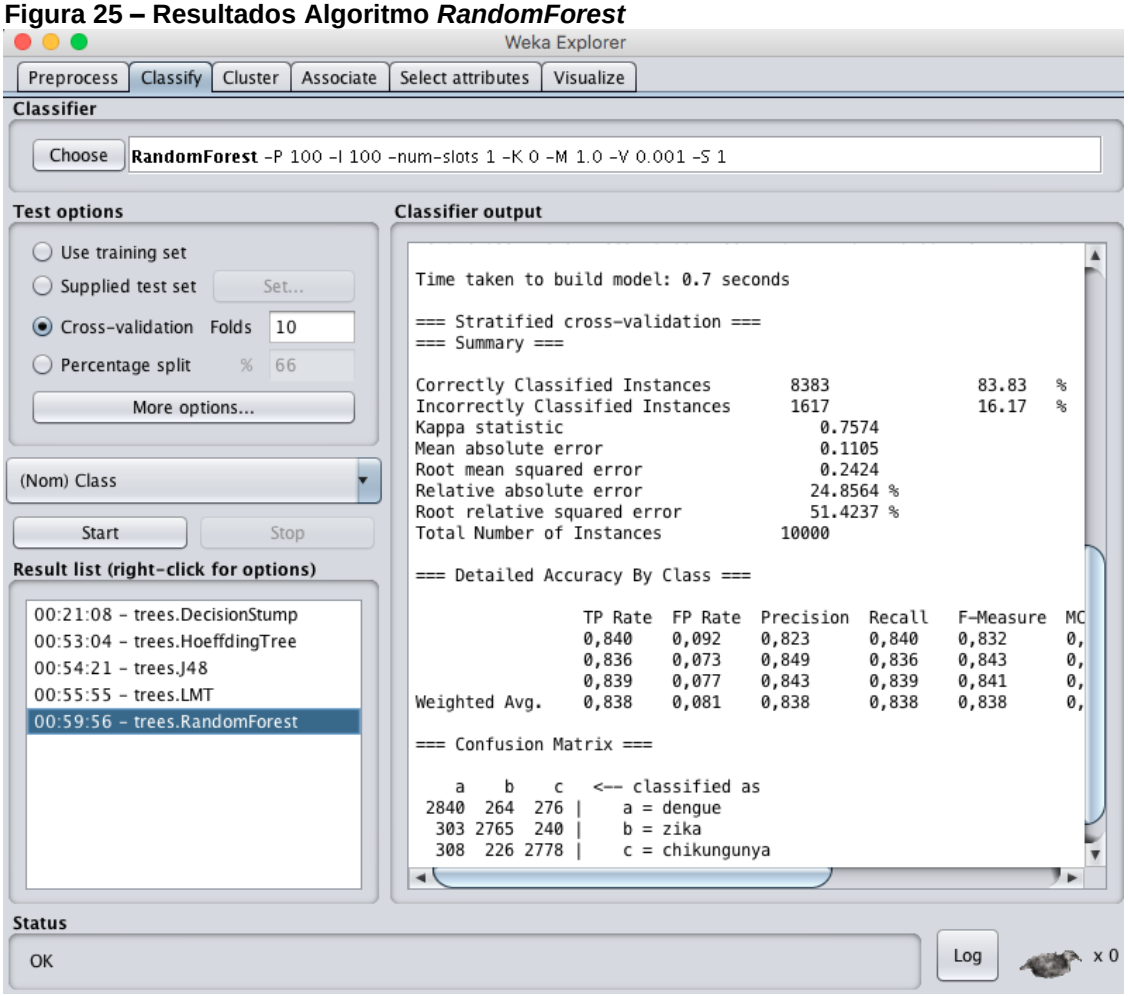
Dengue	Zika	Chikungunya
2814	273	293
306	**2771**	231
313	246	**2753**

Fonte: Autoria própria (2017)

5.3.5 Algoritmo RandomForest

O resultado do algoritmo *RandomForest* possui uma taxa de acerto alta, de 83,83% e sua estatística Kappa 0.7574, se mostrando um bom classificador para o que se espera, conforme mostrado na Figura 25.

Figura 25 – Resultados Algoritmo *RandomForest*



Fonte: Autoria própria (2017)

A matriz de confusão do classificador *RandomForest,* que mostra na diagonal principal os elementos classificados corretamente e destacados em negrito, é exibido a seguir na Tabela 7.

Tabela 7 – Matriz de confusão – *RandomForest*

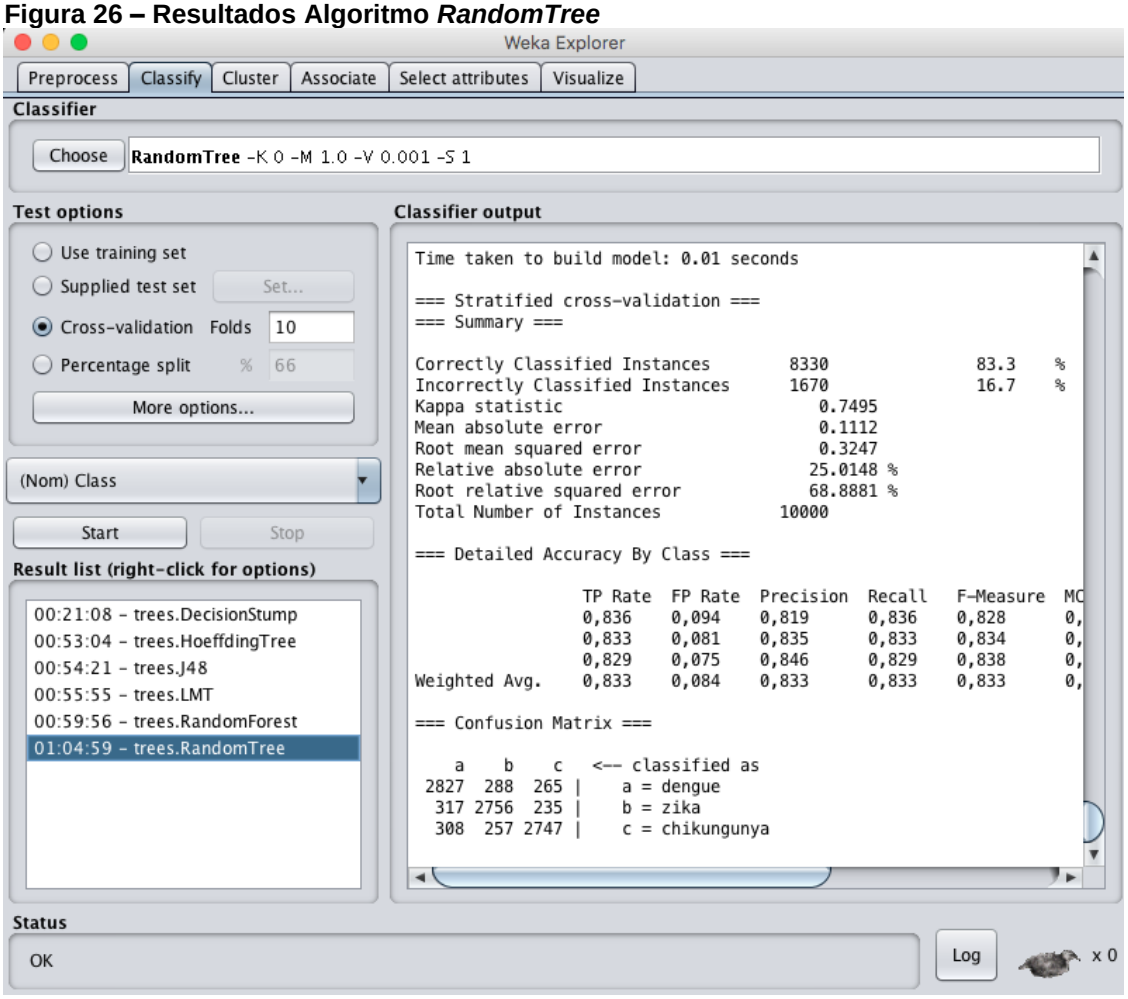
Dengue	Zika	Chikungunya
2840	264	276
303	**2765**	240
308	226	**2778**

Fonte: Autoria própria (2017)

5.3.6 Algoritmo *RandonTree*

O resultado do algoritmo *RandonTree* possui uma taxa de acerto alta, de 83,83% e sua estatística Kappa 0.7495, se mostrando um bom classificador para o que se espera, conforme mostrado na Figura 26.

Figura 26 – Resultados Algoritmo *RandomTree*



Fonte: Autoria própria (2017)

A matriz de confusão do classificador *RandomTree,* que mostra na diagonal principal os elementos classificados corretamente e destacados em negrito*,* é exibida a seguir na Tabela 8.

Tabela 8 – Matriz de confusão – *RandomTree*

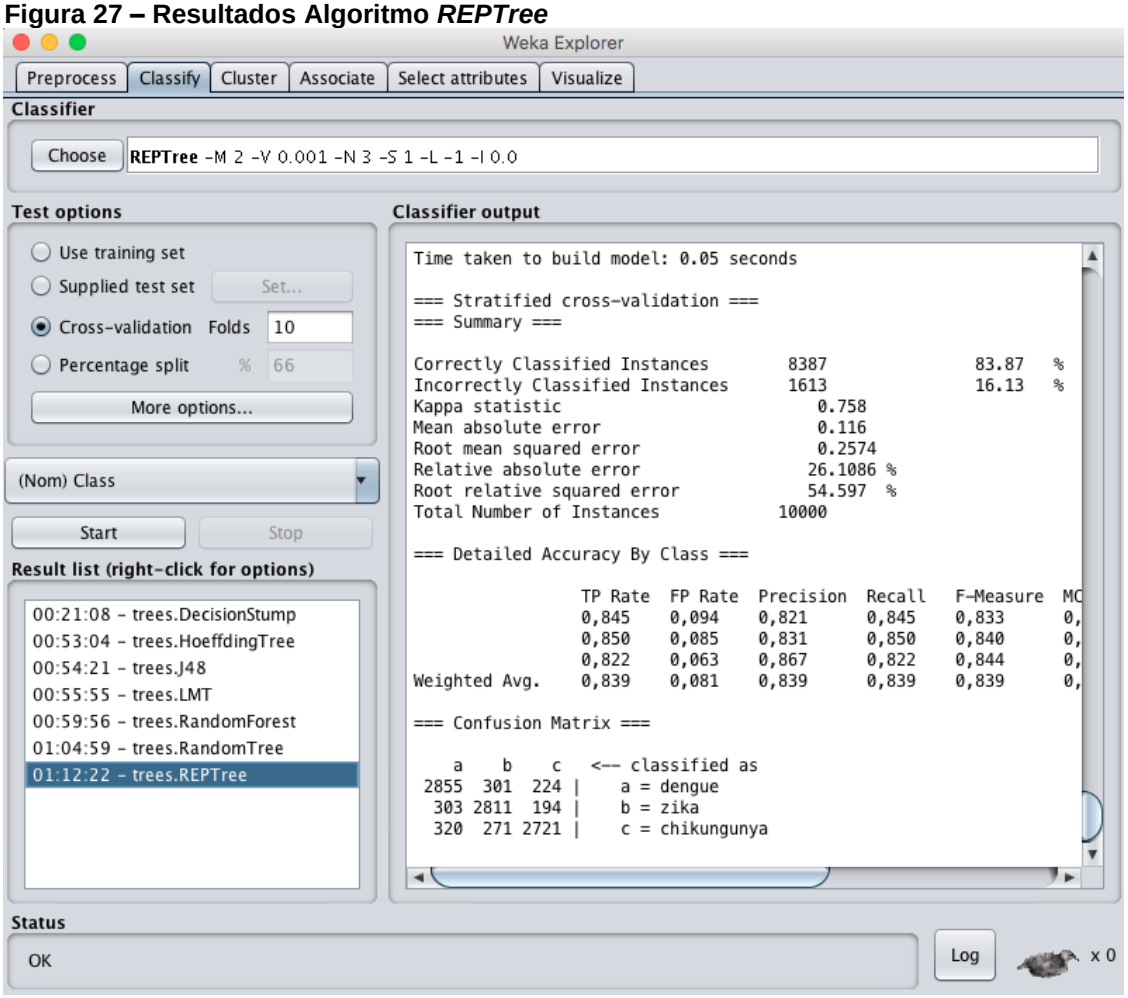
Dengue	Zika	Chikungunya
2827	288	265
317	**2756**	235
308	257	**2747**

Fonte: Autoria própria (2017)

5.3.7 Algoritmo *REPTree*

O resultado do algoritmo *REPTree* possui uma taxa de acerto alta, de 83,87% e sua estatística Kappa 0.758, se mostrando um bom classificador para o que se espera, conforme mostrado na Figura 27.

Figura 27 – Resultados Algoritmo *REPTree*



Fonte: Autoria própria (2017)

A matriz de confusão do classificador *REPTree*, que mostra na diagonal principal os elementos classificados corretamente e destacados em negrito, é exibida a seguir na Tabela 9.

Tabela 9 – Matriz de confusão – *REPTree*

Dengue	Zika	Chikungunya
2855	301	224
303	**2811**	194
320	271	**2721**

Fonte: Autoria própria (2017)

5.4 RESULTADOS OBTIDOS

Ao analisar os resultados obtidos pelo Weka, nota-se uma precisão de 83.87% de dados classificados corretamente, conforme mostrado na Figura 15. Foram analisadas 10 mil instâncias utilizando cada classificador proposto. O *J48*, que gerou uma árvore de decisão com 594 folhas de tamanho 943, juntamente com o classificador *REPTree* foram os melhores classificadores. Analisando dados coerentes, a porcentagem de assertividade do sistema mostra que o auxílio para criar diagnósticos precisos é de grande valia ao setor de medicina.

O comparativo dos resultados de todos os algoritmos utilizados nos testes pode ser visto na Tabela 10 a seguir.

Tabela 10 - Comparação de Classificadores

Classificador	Corretos	Incorretos	Kappa
DecisionStump	5836	4164	0.3754
HoeffdingTree	8346	1654	0.7518
J48 (C4.5)	8387	1613	0.758
LMT	8338	1662	0.7507
RandomForest	8383	1617	0.7574
RandonTree	8330	1670	0.7495
REPTree	8387	1613	0.758

Fonte: Autoria própria (2017)

Com exceção do Algoritmo *DecisionStump,* que teve uma taxa de erro muito alta e principalmente sua estatística Kappa muito baixa, todos os outros algoritmos são recomendados para analisar um grande conjunto de dados clínicos.

Importante ressaltar que, na área médica as classificações dadas como incorretas são na verdade falso-positivos. O paciente apresenta alguma enfermidade que deve ser tratada e, seus dados utilizados para uma melhor aferição futura.

5.5 CONSIDERAÇÕES DO CAPÍTULO

Neste Capítulo apresentou os resultados obtidos através da análise de dados pelo *software* Weka. Os resultados se mostraram consistentes após os testes, confirmando que a análise de grandes dados pode auxiliar na classificação dos mesmos e auxiliar o setor de saúde a tomar as melhores decisões baseado no histórico das enfermidades conhecidas.

6. CONCLUSÃO

A computação avançou de maneira muito rápida nas duas últimas décadas. A geração de dados diariamente produzida é uma quantidade muito grande, tornando difícil sua análise sem a presença de sistemas específicos. Esse trabalho foi focado na área médica, com ênfase em diagnósticos clínicos. A análise de um bom classificador pode auxiliar nas decisões a serem tomadas, facilitando o aprendizado e a tomar decisões de maneira rápida e objetiva.

O Sistema foi pensado para ser utilizado principalmente em comunidades carentes, onde o custo da hora médica é muito alto, e muitas pessoas precisam ser atendidas em pouco tempo, auxiliando em uma triagem de pacientes, onde casos de menor importância podem ser solucionados sem a presença de um médico.

A precisão de acerto utilizando um bom classificador, demonstra que sua utilização contribui em inúmeras áreas de estudo, sabendo organizar os dados e analisar os resultados obtidos, aplicando de melhor forma o conhecimento descoberto.

A geração de *dataset* utilizando a linguagem *Pyton* se mostrou muito efetiva. A seleção de dados úteis para o fim desejado, organizando os atributos da maneira que deixe a descoberta de conhecimento eficiente contribui para análise de dados quando não se possui o as informações necessárias.

6.1 CONSIDERAÇÕES FINAIS

O objetivo desse trabalho foi de mostrar como é possível utilizar ferramentas de descoberta de conhecimento como o Weka em uma base de dados com muita informação e organiza-los de uma maneira que sua compreensão seja fácil. Utilizando tais ferramentas e agregando conhecimento de desenvolvimento de sistemas, é possível criar um sistema que auxilie na solução de problemas específicos.

Na medicina, a quantidade de informações relacionadas é muito grande, fazendo com que o ser humano tenha dificuldade para dominar e armazenar

esses dados. Sistemas computacionais não vão se esquecer de casos específicos, podendo ser utilizado como ferramenta de apoio constantemente por profissionais da área em estudo.

6.2 TRABALHOS FUTUROS

Com esse trabalho pretende-se organizar uma base de informações ampla sobre diversas doenças que atingem principalmente a população mais carente. Disponibilizar o sistema de forma gratuita para hospitais e clínicas, fazendo com que em um primeiro momento consiga-se uma base de dados ampla e consistente de diversos tipos de enfermidades e posteriormente organizar esses dados para que se possa minerá-los e assim descobrir mais conhecimento para novos sistemas com maior complexidade.

REFERÊNCIAS

ALVARENGA, M. T. **Utilização da ferramenta j48 para descoberta do conhecimento em bases de dados fitossanitários, climáticos e espectrais**. 2014. 42 p. Monografia (Graduação em Ciência da Computação) - Universidade Federal de Lavras, Lavras, 2014.

BANSAL, K; Vadhavkar, S.; Gupta, A.: **Neural Networks Based Dataming Aplications for Medical Inventory Problems**, Data Mining and Knowledge Discovery, 2, 1998.

BATISTA, Gustavo Enrique de Almeida Prado Alves. **Pré-processamento de dados em aprendizado de máquina supervisionado**. 2003. Tese (Doutorado em Ciências de Computação e Matemática Computacional) - Instituto de Ciências Matemáticas e de Computação, University of São Paulo, São Carlos, 2003. doi:10.11606/T.55.2003.tde-06102003-160219. Acesso em: 20 nov. 2017.

BERRY, M. J. A.; LINOFF, G. **Data Mining Tehniques** – for marketing, sales, and customer support. United States: Wiley Computer Publishing, 1997.

BOOTSTRAP – The world's most popular mobile-first and responsive – Disponível em: <http://getbootstrap.com/>. Acesso em: 22 nov. 2017.

COLLAZO, K.; BARRETO, J. KDD ferramenta para análise de dados epidemiológico. **Anais do III Congresso Brasileiro de Computação** – Workshop de Informática aplicada à Saúde-CBXOMP'2003, Itajaí, p.2226, 1003. Acesso em: 22 set. 2017.

DE SOUZA, Fernando Menezes Campello. **Artigo Teorias da probabilidade**. 2004.

DIEGONOGARE – Disponível em: <http://www.diegonogare.net/2015/01/azure-machine-learning-matriz-de-confusao-parte-4/>. Acesso em: 21 nov. 2017.

FERNANDES, A. P. S. **Sistema Especialista Difuso de Apoio ao Aprendizado do Traumatismo Dento-Aveolar** Utilizando Recursos Multimídia. 1997, 96 f. (Dissertação) Florianópolis: Universidade Federal de Santa Catarina, 1997.

FIOCRUZ. **Saúde e Ciência para Todos**. 2013. Disponível em: <https://agencia.fiocruz.br>. Acesso em: 01 Dez 2016.

FMP – **Faculdade de Medicina Do Porto** <https://users.med.up.pt/~joakim/intromed/estatisticakappa.htm>. Acesso em: 22 nov. 2017.

INF1771 – **Inteligência Artificial**. Aula 19 – Aprendizado por reforço. Formas de Aprendizado. Aprendizado Supervisionado. Árvores de Decisão. K-Nearest Neighbor (KNN). Support Vector Machines (SVM). Redes Neurais.

Kibler, D. & Langley, P. **Machine Learning as an Experimental Science. Machine Learning.** 1988.

LIMA, D. R.; ROSATELLI, M. C. Um sistema tutor inteligente para um ambiente virtual de ensino aprendizagem. In CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 23., 2003, Porto Alegre. **Anais...** Porto Alegre: 2003

LINDERMANN, V. **Raciocínio baseado em casos**. In: BARONE, D. (org). Sociedades artificiais: a nova fronteira da inteligência das máquinas. Porto Alegre: Bookman, 2003. ISBN: 85-363-0124-4.

MASSAD, E. **Métodos quantitativos em Medicina**. Barueri: Manole, 2004. ISBN 85-204-1412-5.

MANCHINI, D.P.; PAPPA, G.L. **Sistemas Especialistas** Disponível em: <www.din.uem.br/ia/intelige/especialistas/especialistas/index.html> Acesso em: 18 nov. 2016.

MENDES, Raquel Dias. Inteligência Artificial: Sistemas Especialistas no Gerenciamento da Informação. **Ci. Inf.**, Brasilia , v. 26, n. 1, p. , Jan. 1997. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-19651997000100006&lng=en&nrm=iso>. Acesso em: 12 nov. 2016.

NILSON, N.S. **Principles of Artificial Inteligence**. Springer Verlag, Berlin, 1982.

NODE.JS FOUNDATION. ***NODE.JS Evented I/O for V8 JavaScript***. Disponível em: <http://nodejs.org/>. Acesso em: 22 nov. 2017.

RENAUX, D. P. B.; et al. Gestão do conhecimento de um laboratório de pesquisa: uma abordagem prática. In: SIMPÓSIO INTERNACIONAL DE GESTÃO DO CONHECIMENTO. 4., 2001, Curitiba. **Anais...** Curitiba: PUC-PR, 2001. p. 195-208.

REZENDE, S. O. **Sistema Inteligentes: fundamentos e aplicações**. Ed. Barueri, SP: Manoele. 2003.

SAS Intitute Inc., **Machine Learning**. Disponível em: <https://www.sas.com/pt_br/insights/analytics/machine-learning.html>. Acesso em: 23 set 2017.

WIKI, Disponível em: <www.weka.wikispaces.com>. Acesso em: 23 set. 2017.

WITTEN, I. H.; FRANK, E. **Data mining: practical machine learning tools and techniques**. 2 ed. San Francisco: Morgan Kaufmann Publishers, 2005.