

FUNDAÇÃO CENTRO DE ANÁLISE, PESQUISA E INOVAÇÃO TECNOLÓGICA
FACULDADE FUCAPI (INSTITUTO DE ENSINO SUPERIOR FUCAPI)
COORDENAÇÃO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Uso de técnicas de *Data Mining* na classificação de tipos de dengue

Jeferson Barros Alves

Manaus - AM

2015

Jeferson Barros Alves

Uso de técnicas de *Data Mining* na classificação de tipos de dengue

Monografia apresentada ao Curso de Graduação em Ciência da Computação da Faculdade Fucapi (Instituto de Ensino Superior Fucapi), como requisito parcial para a obtenção do Título de Bacharel em Ciência da Computação. Área de concentração: Banco de Dados.

Orientador: Márcio Palheta Piedade, M.Sc.

Manaus - AM

2015

Jeferson Barros Alves

Uso de técnicas de *Data Mining* na classificação de tipos de dengue

Monografia apresentada ao Curso de Graduação em Ciência da Computação da Faculdade Fucapi (Instituto de Ensino Superior Fucapi), como requisito parcial para a obtenção do Título de Bacharel em Ciência da Computação.

Aprovada em: / / , por:

Prof. Márcio Palheta Piedade, M.Sc.
Faculdade Fucapi
Orientador

Prof. Sergio Cleger Tamayo, Dr.
Faculdade Fucapi
Examinador

Profa. Kelen Acquati Vieira, M.Sc.
Faculdade Fucapi
Examinadora

Manaus - AM
2015

Agradecimentos

Em primeiro lugar quero agradecer a Deus, meu grande amigo, companheiro, e sempre me mostrou o caminho a ser seguido.

A minha amada esposa Suzane Barros, que sempre me incetivou nessa jornada.

Ao meu orientador Prof. Márcio Palheta, por mostrar-me o caminho da mineração de dados.

Aos meus amigos, Hussama, Wellington, Patrick, Pedro e Tiago que sempre estiveram por perto nesta caminhada.

Agradecimentos também aos que contribuíram de maneira relevante à elaboração deste trabalho, direta ou indiretamente.

Resumo

A tarefa de classificação em *Data Mining* consiste na predição de classe dado um determinado conjunto de atributos de uma instância. Esta técnica pode ser útil no diagnóstico médico, como ferramenta de apoio ao profissional da saúde. Na literatura, as melhores técnicas para classificação utilizam máquinas de vetores de suporte e árvore de decisão. Tais técnicas utilizam o conceito de hiperplano para separar de uma melhor forma as classes existentes em conjuntos de dados, também pode-se verificar o uso da divisão e conquista na identificação dos melhores atributos, sendo capazes de produzir resultados mais eficientes no processo de predição. Como resultado, obtivemos modelos classificadores baseados em SMO e J48, que alcançaram resultados de 89,61% e 86,63% na classificação correta de instâncias, respectivamente nos conjuntos de dados utilizados, mostrando resultados satisfatórios em relação ao classificador ZeroR utilizado como *baseline*. Foi possível observar, também, que os modelos gerados obtiveram melhores resultados à medida que os experimentos no tamanho do conjunto de dados aumentava.

Palavras-chave: Mineração de Dados, Classificação, Árvore de Decisão, Bayes, Máquina de Vetor de Suporte, Dengue.

Abstract

The classification task in textit Data Mining is the class prediction given a certain set of attributes for an instance. This technique can be useful in medical diagnosis, as a support tool for the health professional. In the literature, the best techniques for classification using support vector machines and decision tree. Such techniques use the concept of hyperplane to separate in a better way the existing classes in data sets, you can also verify the use of divide and conquer to identify the best attributes, being able to produce more efficient results in the prediction process. As a result, we obtained classifiers based models SMO and J48, reaching results of 89.61 % and 86.63 % in the correct classification of instances respectively in the data sets used in this study, showing satisfactory results in relation to the classifier Zeror using as textit baseline. It was also observed that the generated models performed better as the experiments in the data set size increased.

Keywords: Data mining, Classifier, Decision Tree, Bayes, Support Vector Machine, Dengue.

Lista de figuras

Figura 1 – Etapas do processo KDD.	18
Figura 2 – Abordagem geral para a construção de um modelo de classifi- cação	21
Figura 3 – Núcleo do Algoritmo C4.5	23
Figura 4 – Conjunto de hiperplanos possíveis	27
Figura 5 – Data Set Original	32
Figura 6 – Erro ao Carregar Instâncias	32
Figura 7 – Erro Mencionado pelo WEKA	32
Figura 8 – Filtro <i>RemoveUseless</i>	33
Figura 9 – Filtro <i>NumericToNominal</i>	34
Figura 10 – Filtro <i>RemoveRange</i>	34
Figura 11 – Metodologia Processo dos Experimentos	35
Figura 12 – TP Rate	37
Figura 13 – FP Rate	38
Figura 14 – Gráficos de Acertos e Erros de Classificação	39
Figura 15 – Modelos de Classificação no Espaço ROC	40

Lista de tabelas

Tabela 1 – Matriz de confusão para problemas de classes binárias 28

Tabela 2 – Resultados de Classificação 37

Lista de abreviaturas e siglas

KDD	<i>Knowledge Discovery in Databases</i>
WEKA	<i>Waikato Environment for Knowledge Analysis</i>
SVM	<i>Support Vector Machine</i>
SMO	<i>Sequential Minimal Optimization</i>
SINAN	Sistema de Informação de Agravos de Notificação
SUS	Sistema Único de Saúde

Sumário

1	INTRODUÇÃO	11
1.1	Especificação do problema	12
1.2	Objetivos	12
1.2.1	Objetivo Geral	12
1.2.2	Objetivos Específicos	13
1.3	Justificativa	13
1.4	Trabalhos Relacionados	14
1.5	Metodologia de Desenvolvimento	15
1.6	Estruturação da Monografia	16
2	REFERENCIAL TEÓRICO	17
2.1	Descoberta de Conhecimento em Base de Dados	17
2.2	Data Mining	19
2.2.1	Aprendizagem Não Supervisionada	20
2.2.2	Aprendizagem Supervisionada	20
2.3	Técnicas de Classificação	20
2.3.1	Árvore de Decisão	22
2.3.1.1	Algoritmo C4.5	22
2.3.1.2	Algoritmo <i>RandomTree</i>	25
2.3.1.3	Algoritmo <i>REPTree</i>	25
2.3.2	Teorema de Bayes	25
2.3.2.1	Algoritmo Naive Bayes	26
2.3.3	Máquinas de Vetores de Suporte (SVM)	26
2.3.3.1	Algoritmo SMO	27
2.3.4	Métricas para Avaliação de Modelo	28
2.4	WEKA (<i>Waikato Environment for Knowledge Analysis</i>)	29
2.5	Dados Abertos	29
2.6	SINAN	30

3	EXPERIMENTOS	31
3.1	Base de Dados	31
3.2	Pré-processamento	31
3.3	Metodologia do Experimento	34
3.3.1	Parâmetros	36
3.4	Resultados dos Algoritmos	36
4	CONCLUSÃO E TRABALHOS FUTUROS	41
4.1	Conclusão	41
4.2	Trabalhos Futuros	42
	Referências	43

1 Introdução

Com a grande quantidade de informações geradas e armazenadas por meio do avanço tecnológico das últimas décadas, houve um acúmulo gigantesco de informações. Com isso, surgiram diversas abordagens, técnicas e ferramentas que buscam transformar dados, também conhecida por Descoberta de Conhecimento em Base de Dados (*Knowledge Discovery in Databases - KDD*). Tal abordagem foi proposta em 1989, com o objetivo de analisar os dados de uma base afim de extrair conhecimento útil (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996).

A computação tem apoiado o desenvolvimento da medicina em diversas áreas como em sistemas de apoio a coleta de dados clínicos e exames por imagens, na organização das informações obtidas, entre outras (COSTA, 2012). Desta maneira, esse grande volume de dados é uma fonte valiosa de conhecimentos que pode ser utilizada como auxílio ao diagnóstico médico. Com base nisso, é necessário o desenvolvimento de técnicas que permitam a descoberta de conhecimentos em base de dados médicos, para apoiar o médico em sua tarefa diária de tomada de decisões, aumentando a precisão, a confiabilidade e eficiência dos diagnósticos elaborados pelo especialista (COSTA, 2012).

O número de casos de dengue no Brasil quase triplicou entre 2009 e 2010 segundo dados do Ministério da Saúde. A partir desses dados, mais de um milhão de pessoas podem ter contraído a doença (SANTOS; NETO, 2011). Por esta razão observou-se a oportunidade de aplicar técnicas de classificação utilizadas em *Data Mining*, com o objetivo de gerar modelos classificadores com dados oriundos do SINAN (Sistema de Informações de Agravos de Notificação), tendo em vista que o mesmo é utilizado com meio oficial de registro em casos de dengue.

1.1 Especificação do problema

Ao longo dos últimos anos a mineração de dados tem sido cada vez mais utilizada na literatura médica. No entanto, a sua aplicação na análise de dados médicos tem sido relativamente limitada, tendo em vista que a aplicação prática pode explorar o conhecimento disponível no contexto clínico e explicar decisões propostas (BELLAZZI; ZUPAN, 2008). Faz-se importante o desenvolvimento de técnicas que permitam a descoberta de conhecimento em bases de dados médicos, que possam apoiar o profissional em sua tarefa diária de tomada de decisões, aumentando a precisão, a confiabilidade e eficiência dos diagnósticos elaborados pelo especialista (COSTA, 2012).

Segundo (SAÚDE, 2013), a dengue é uma das doenças com maior incidência no Brasil, atingindo a população de todos os estados, independente de classe social. Diante disto, faz-se necessário que um conjunto de ações com intuito de combater os casos de dengue sejam realizadas. Dentre elas temos a previsão de forma antecipada de possíveis novos casos. A classificação epidemiológica dos casos de dengue, que é feita habitualmente após desfecho clínico, na maioria das vezes é retrospectiva, dependendo de informações clínicas e laboratoriais, disponíveis no final do acompanhamento médico. Esses critérios não permitem o reconhecimento precoce de formas potencialmente graves, para as quais é crucial para a instituição de tratamento imediato (SAÚDE, 2013).

1.2 Objetivos

1.2.1 Objetivo Geral

Construir modelos utilizando as técnicas de classificação em *Data Mining* conhecidas como *Árvore de Decisão*, *Bayes* e *Support Vector Machine*. Após a criação dos modelos de classificação, avaliar o que obteve melhor resultado em relação ao conjunto de dados utilizados no SINAN (Sistema de Informação de Agravos de Notificação) Dengue do município de Fortaleza.

1.2.2 Objetivos Específicos

- Dividir o conjunto de dados na seguinte quantidade de instâncias: 5.000, 10.000, 15.000, 20.000 e 25.000;
- Utilizar atributos mais relevantes na tarefa de Classificação;
- Construir modelos clasificadores com os algoritmos de classificação: J48, REPTree, RandomTree, NaiveBayes e SMO;
- Identificar quais modelos gerados possuem melhor desempenho;
- Validar e interpretar os resultados obtidos no processo de classificação.

1.3 Justificativa

Transmitida pelo mosquito *Aedes aegypti*, a dengue é uma doença viral que se espalha rapidamente pelo mundo. Segundo a Organização Mundial da Saúde (OMS) por volta de 100 milhões de pessoas são infectadas por dengue anualmente, em mais de 100 países, sendo que 2,5 bilhões de pessoas estão sob o risco de infecção, tornando-se assim um dos maiores problemas de saúde pública no mundo (BHATT et al., 2013). A dengue é, hoje, uma das doenças mais frequentes no Brasil, atingindo a população em todos os estados, independente de classe social (SAÚDE, 2008).

Na região das Américas, a doença tem se disseminado com surtos cíclicos ocorrendo a cada 3/5 anos. No Brasil, a transmissão vem ocorrendo de forma continuada desde 1986, intercalando-se com a ocorrência de epidemias, geralmente associadas com a introdução de novos sorotipos em áreas anteriormente não se tinha registro ou alteração do sorotipo predominante. O maior surto no Brasil ocorreu em 2013, com aproximadamente 2 milhões de casos notificados. Atualmente, circulam no país os quatro sorotipos da doença (SAÚDE, 2015).

1.4 Trabalhos Relacionados

No trabalho de (SHAKIL; ANIS; ALAM, 2015), observamos que o foco principal do trabalho foi a classificação de dengue, onde inicialmente aplicou-se a classificação nos *datasets* iniciais para identificar qual algoritmo obteve melhor resultado. Os experimentos mostraram que os algoritmos Naive Bayes e J48 obtiveram melhores resultados, onde as maiores contribuições do trabalho foram:

- A extração da acurácia de classificação para predição do diagnóstico de dengue.
- Comparação de diferentes algoritmos de mineração no conjunto de dados de dengue.
- Identificação dos algoritmos com melhor desempenho para previsão dos diagnósticos.

No trabalho de (THITIPRAYOONWONGSE; SURIYAPHOL; SOONTHORNPHISAJ, 2012), foi utilizada a técnica de classificação conhecida como Árvore de Decisão (TAN; STEINBACH; KUMAR, 2009), aplicado a um conjunto de dados temporais contendo dados clínicos e laboratoriais com mais de 400 atributos. Nos experimentos, os dados foram divididos em: conjunto de teste e treinamento, de modo que, o seu objetivo principal foi identificar o dia zero da dengue, pois esta previsão torna-se crítica, tendo em vista que o dia zero tem forte relação com o melhor tratamento do paciente.

No trabalho (SANTOS; NETO, 2011), foi desenvolvido um aplicativo para interação dos profissionais da saúde com os dados do SINAN (Sistema de Informação de Agravos de Notificação). Tal aplicativo, escrito em *Java*(ORACLE, 2015), implementa algoritmos de classificação e regras de associação da base de dados do SINAN, utilizando o *software* WEKA (*Waikato Environment for Knowledge Analysis*) (WAIKATO, 2015). Como resultado, esta ferramenta possibilitou o auxílio no diagnóstico de pacientes.

1.5 Metodologia de Desenvolvimento

Com base na metodologia utilizada no processo de Descoberta de Conhecimento em Base de Dados (*Knowledge Discovery in Databases* - KDD) definida em (FAYYAD; PIATETSKY-SHAPIO; SMYTH, 1996), a realização desta pesquisa será composta das seguintes etapas:

- A primeira etapa deste trabalho baseado na revisão bibliográfica da literatura relacionada à mineração de dados, para determinar trabalhos relacionados, onde são aplicadas as técnicas de mineração em base de dados médicos para descoberta de conhecimento novo e relevante, que possa auxiliar em diagnósticos do médico especialista;
- Na segunda etapa, ocorrerá a extração, manutenção e pré-processamento dos dados, onde será realizada a análise de base de dados para extração de informações de atendimentos de pacientes;
- Na terceira etapa, será executado algoritmos para descoberta e extração de informações. O objetivo desta etapa será encontrar padrões na base de dados que foi pré-processada, utilizando a ferramenta de mineração WEKA (*Waikato Environment for Knowledge Analysis*), um software *open-source* (INITIATIVE, 2015) amplamente utilizado por implementar diversos algoritmos de mineração de dados (HALL et al., 2009).
- Por fim, na quarta etapa do processo, ocorrerá a análise dos resultados obtidos com a aplicação dos algoritmos de mineração de dados. Encontrar padrões de classificação que sejam relevantes para a pesquisa em questão.

1.6 Estruturação da Monografia

Além deste capítulo introdutório, este trabalho está organizado da forma descrita à seguir. No Capítulo 2, serão descritos conceitos básicos relacionados à compreensão deste trabalho, bem como o referencial teórico. No Capítulo 3, serão apresentados os experimentos juntamente com os resultados obtidos. Finalmente no Capítulo 4, serão apresentadas as conclusões observadas e sugestões de trabalhos futuros da pesquisa.

2 Referencial Teórico

Este Capítulo apresenta uma visão geral dos conceitos envolvidos nas várias técnicas de classificação utilizadas ao longo deste trabalho.

Neste capítulo, apresentamos uma visão geral dos conceitos e procedimentos envolvidos em KDD (*Knowledge Discovery in Databases*) e consequentemente em Data Mining. Apresentam-se também os trabalhos encontrados na literatura sobre assunto que são relevantes para o desenvolvimento deste trabalho.

2.1 Descoberta de Conhecimento em Base de Dados

Segundo (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996), a Descoberta de Conhecimento em Base de Dados é um processo não trivial de identificações de novos padrões, válidos e potencialmente úteis.

Em contrapartida, no trabalho de (THOMÉ, 2002), define-se KDD como sendo a busca de extração de conhecimento de bases de dados utilizando-se de técnicas e algoritmos que realizam a mineração dos dados para trabalhar e descobrir relações.

O processo de descoberta de conhecimento ocorre quando um conjunto de padrões que são semelhantes, e que podem levar a construção de um modelo. Este processo é formado por 5 etapas: seleção, pré-processamento, transformação, a mineração de dados e a interpretação dos dados (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996). Além do processo, o conhecimento que se deseja buscar deve estar de acordo com três características: deve ser correto, deve ser compreensível para o usuário e deve ter de alguma forma utilidade para o usuário (FREITAS, 2000).

A seguir serão detalhadas as etapas do processo de KDD de acordo com o apresentado na figura 1.

A etapa de seleção dos dados inicia com a definição do objetivo e mapeamento dos grupos ou conjuntos de informações que serão utilizados.

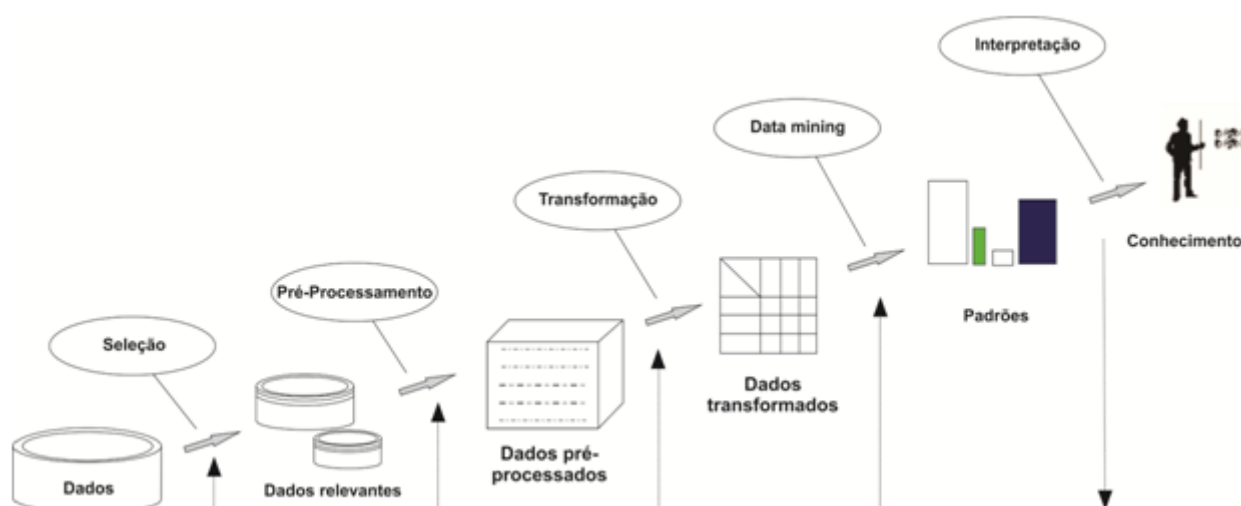
O pré-processamento é responsável pelo tratamento de ruídos e dados incompletos.

A transformação tem como objetivo selecionar as principais características que serão utilizadas para representar os dados, ou seja, os dados devem ser selecionados de modo que sejam os mais úteis para o modelo proposto.

A etapa de mineração de dados é o momento em que serão escolhidos os algoritmos que mais se ajustam ao objetivo que se quer extrair da base de dados. Além disso, nesta fase são escolhidos os melhores parâmetros para que, no momento do processamento, os resultados sejam os mais rápidos e precisos possíveis.

Ao final do processo teremos a etapa de interpretação e avaliação dos resultados, onde o conhecimento extraído da base de dados é representado por padrões.

Figura 1 – Etapas do processo KDD.



Fonte: Adaptado de (FAYYAD; PLATETSKY-SHAPIRO; SMYTH, 1996).

Inicialmente, é necessário definir que tipo de conhecimento se deseja extrair da base de dados, pois a técnica que será utilizada para a mineração de

dados depende do objetivo a que se quer chegar (DAMASCENO, 2005).

2.2 Data Mining

Em (BERRY; LINOFF, 1997), definiu-se *Data Mining* como um processo de exploração e análise, de uma grande quantidade de dados, por meio automático ou semiautomático, com o propósito de descobrir regras e padrões significativos.

É uma técnica que faz parte de uma das etapas da descoberta de conhecimento em banco de dados. É uma área de pesquisa multidisciplinar, incluindo principalmente as tecnologias de banco de dados, inteligência artificial, estatística, reconhecimento de padrões, sistemas baseados em conhecimento, recuperação da informação, computação de alto desempenho e visualização de dados.

Em termos gerais, a técnica de Data Mining compreende os seguintes propósitos:

- Previsão – pode mostrar como certos atributos dentro dos dados irão comportar-se no futuro;
- Identificação – padrões de dados podem ser utilizados para identificar a existência de um item, um evento ou uma atividade;
- Classificação – pode repartir os dados de modo que diferentes classes ou categorias possam ser identificadas com base em combinações de parâmetros;
- Otimização do uso de recursos limitados, como tempo, espaço, dinheiro ou matéria-prima e maximizar variáveis de resultado como vendas ou lucros sob um determinado conjunto de restrições.

2.2.1 Aprendizagem Não Supervisionada

Como mostrado por (DAMASCENO, 2005) aprendizagem não supervisionada é aquela que utiliza instâncias sem a determinação do atributo classe. Este tipo de aprendizado é utilizado geralmente para análise exploratória dos dados, utilizando técnicas de agrupamento ou regras de associação. Onde agrupamentos têm como objetivo relacionar instâncias com características em comuns.

A partir da definição de uma métrica de similaridade, os dados são agrupados, dando a possibilidade de encontrar relações interessantes entre as instâncias. Assim, o cliente do conhecimento gerado pode aplicar uma determinada ação em um subconjunto de instâncias presente nos dados.

2.2.2 Aprendizagem Supervisionada

No aprendizado supervisionado tem-se a figura de um professor externo, o qual apresenta o conhecimento do ambiente por conjuntos de exemplos na forma: entrada, saída desejada. Onde extrai-se a representação do conhecimento a partir desses exemplos. O objetivo é que a representação gerada seja capaz de produzir saídas corretas para novas entradas não apresentadas previamente (LORENA; CARVALHO, 2007).

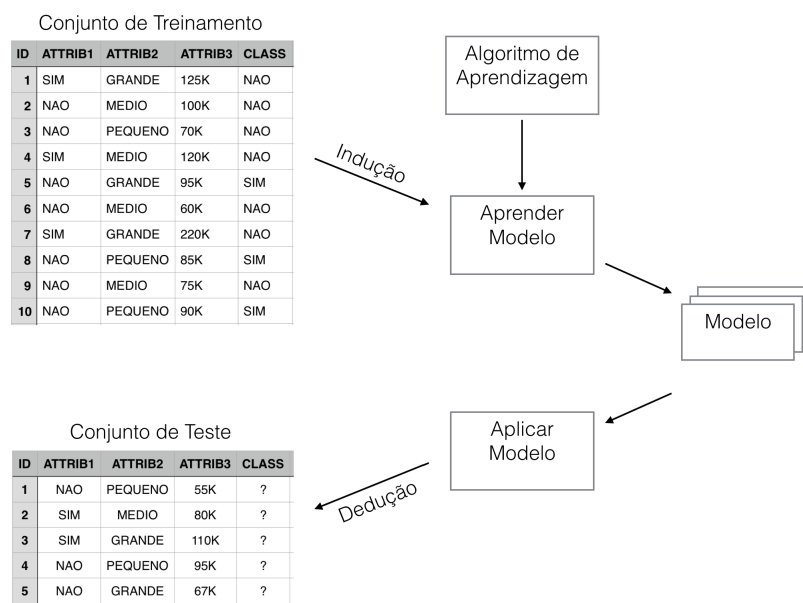
2.3 Técnicas de Classificação

Uma técnica de classificação é uma abordagem sistemática para construção de modelos de classificação a partir de um conjunto de dados de entrada. Exemplos incluem classificadores de árvores de decisão, classificadores baseados em regras, redes neurais, máquinas de vetores de suporte e classificadores Bayes simples. Cada técnica emprega um algoritmo de aprendizagem para identificar um modelo que seja mais apropriado para o relacionamento entre o conjunto de atributos e o rótulo da classe dos dados de entrada. O modelo gerado pelo algoritmo de aprendizagem deve se adaptar bem aos dados de entrada e

prever corretamente os rótulos de classes de registros que ele nunca viu antes. Portanto, o objetivo chave do algoritmo de aprendizagem é construir modelos com boa capacidade de generalização (TAN; STEINBACH; KUMAR, 2009).

A Figura 2 mostra uma abordagem geral para resolver problemas de classificação. Primeiro, um conjunto de treinamento consistindo de registros cujos rótulos sejam conhecidos e devem ser fornecidos. O conjunto de treinamento é usado para construir um modelo de classificação, que é subsequentemente aplicado ao conjunto de teste, que consiste de registros com rótulos de classes desconhecidas (TAN; STEINBACH; KUMAR, 2009).

Figura 2 – Abordagem geral para a construção de um modelo de classificação



Fonte: Adaptado de (TAN; STEINBACH; KUMAR, 2009).

2.3.1 Árvore de Decisão

Em uma árvore de decisão, cada nodo folha recebe um rótulo de classe. Os nodos não terminais, que incluem o nodo raiz e outros nodos internos, contêm condições de testes de atributos para separar registros que possuem características diferentes.

Classificar um registro de testes é direto, assim que uma árvore de decisão tenha sido construída. Começando do nodo raiz, aplicamos a condição de teste ao registro e seguimos a ramificação apropriada baseados no resultado do teste. Isto nos levará a um outro nodo interno, para o qual uma nova condição de teste é aplicada, ou a um nodo folha (TAN; STEINBACH; KUMAR, 2009).

2.3.1.1 Algoritmo C4.5

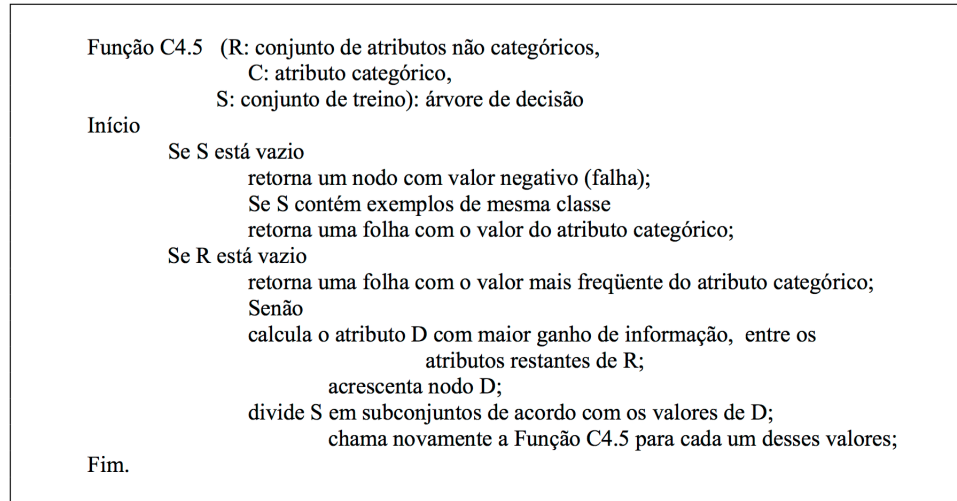
O algoritmo J48 (WAIKATO, 2015) é uma implementação em Java (ORACLE, 2015) do algoritmo C4.5 (SALZBERG, 1994), que visa gerar árvore de decisão com tratamento de atributos contínuos e discretos, construindo uma árvore com um número de partições variável e com folhas sendo indicadas pelos valores do atributo categórico.

Segundo (HALMENSCHLAGER, 2002), para evitar a geração de todas as árvores possíveis, o algoritmo C4.5 se baseia no atributo mais informativo, escolhido entre todos os atributos ainda não considerados no caminho desde a raiz. O algoritmo seleciona como sendo o atributo mais informativo aquele que possuir o maior ganho de informação, resultante da diferença do valor da informação do atributo categórico e do valor da informação (entropia) do atributo em questão.

Para cada atributo é calculado seu ganho de informação. O atributo que tiver o maior valor, será considerado pelo algoritmo como o próximo nodo da árvore. Assim, a partição começa pelo nodo raiz e continua pelos nodos filhos da mesma maneira, até que todos os exemplos desta partição possuam a

mesma classe, rotulando-se este nodo como folha e recebendo sua respectiva classe. Na Figura 3, podemos observar o funcionamento principal do C4.5.

Figura 3 – Núcleo do Algoritmo C4.5



Fonte: (FELDENS; CASTILHO, 1997).

O algoritmo recebe o atributo categórico C (que indica a classe), um conjunto de atributos não categóricos R (demais atributos), e um conjunto de treinamento S (que contém os exemplos). Ele verifica qual é o atributo D mais informativo de R , ou seja, o atributo com maior ganho de informação para o conjunto de treinamento S e, então, subdivide este conjunto S de acordo com cada um dos valores deste atributo mais informativo D (FELDENS; CASTILHO, 1997).

De forma recursiva o algoritmo é chamado para cada subconjunto obtido, até atingir os seguintes objetivos:

- Existindo exemplos para classificar o conjunto de treinamento S . Neste caso, a árvore de decisão é um nodo com valor negativo;
- Que os exemplos no conjunto de treinamento S pertençam todos à mesma classe C . A árvore de decisão é uma folha identificando a classe C_i ;
- Existindo atributos não categóricos R para classificação. A árvore é uma folha identificando a classe C_i mais frequente de S .

Quando o conjunto de treinamentos S contém casos pertencentes a várias classes, a ideia é refinar S em subconjuntos de casos, que tendem a permanecer em apenas uma classe. A árvore de decisão para S consiste de um nodo de decisão que identifica o teste e uma ligação de cada valor possível do atributo. O mesmo processo de construção é aplicado recursivamente em cada subconjunto de casos de treinamento até que a i -ésima ligação tenha a árvore de decisão construída do subconjunto S_i de casos de treinamento. Quando o algoritmo parar, basta percorrer os caminhos desde a raiz até as folhas para verificar as descobertas extraídas da base de dados.

Como observado em (HALMENSCHLAGER, 2002), os critérios que fazem parte do algoritmo C4.5 são:

- Eleição do melhor atributo: utiliza o critério do ganho de informação;
- Tratamento dos atributos discretos: atribui uma ligação distinta a cada valor ou forma de agrupamentos de valores em vários conjuntos;
- Tratamento de atributos contínuos: Utiliza a técnica do teste simples para partição, escolhendo como ponto de decisão o ponto médio entre os valores;
- Tratamento de valores desconhecidos: Desconsidera os atributos com valores desconhecidos nos cálculos de ganho e entropia;
- Determinação da classe associada à folha: É efetuada por atribuição da classe mais provável nesta folha;
- Método de poda: Utiliza a técnica de pós-poda baseada no erro, examinando a árvore de forma *bottom-up* e substituindo uma subárvore por uma folha;
- Complexidade do algoritmo: É dada por $O(mn^2)$, em que m é o número de atributos e n é o número de instâncias do conjunto de treinamento.

2.3.1.2 Algoritmo *RandomTree*

Segundo (OSHIRO, 2013), considerando um conjunto de treinamento T com a atributos e n exemplos, seja T_k uma amostra *bootstrap* do conjunto de treinamento a partir de T com reposição, contendo n exemplos e usando m atributos aleatórios ($m \leq a$) em cada nó das árvores.

RandomTree é uma árvore induzida aleatoriamente a partir de um conjunto de árvores possíveis, usando m atributos aleatórios em cada nó. O termo "aleatoriamente" significa que cada árvore tem uma chance igual de ser utilizada na amostrada. *Random Trees* podem ser geradas eficientemente e a combinação de grandes conjuntos de *Random Trees* geralmente leva a modelos precisos (ZHAO; ZHANG, 2008).

2.3.1.3 Algoritmo *REPTree*

O algoritmo *REPTree* constrói árvores de decisão para classificação ou regressão com base no ganho de informação/variância e poda esta árvore usando uma poda guiada por erro. Otimizado para velocidade, só classifica valores para atributos numéricos uma vez. Os valores são tratadas dividindo as instâncias correspondentes em pedaços (como no algoritmo C4.5) (WITTEN; FRANK; HALL, 2011).

2.3.2 Teorema de Bayes

É um teorema usado para calcular a probabilidade condicional, usando em estatística, probabilidade e outras aplicações

Em muitas aplicações, o relacionamento entre o conjunto de atributos e a variável classe é não determinístico. O rótulo da classe de um registro de teste não pode ser previsto com certeza embora seu conjunto de atributos seja idêntico a alguns dos exemplos de treinamento. Esta situação pode surgir por causa de dados com ruídos ou da presença de determinados fatores de

confusão que afetam a classificação mas que não são incluídos na análise.

$$P(A | B) = \frac{P(B | A) P(A)}{P(B)} \quad (2.1)$$

Sendo $P(A | B)$ a probabilidade *posteriori* condicional de A a B, tem-se que:

- $P(B | A)$ a probabilidade *posteriori* condicional de B a A.
- $P(A)$ probabilidade a *priori* de A.
- $P(B)$ probabilidade a *priori* de B.

2.3.2.1 Algoritmo Naive Bayes

Em (LUCCA et al., 2013), o algoritmo *Naive Bayes* computa a probabilidade de $P(c | d)$ de uma instância pertencer a uma determinada classe a partir da probabilidade a priori $P(c)$ de uma instância ser desta classe e das probabilidades condicionais $P(t_k | c)$ de cada termo t_k ocorrer em uma instância da mesma classe. O objetivo do algoritmo é encontrar a melhor classe C_{map} para uma instância, maximizando a probabilidade a posteriori conforme a equação 2.2, onde n_d é o número de atributos na instância d .

$$C_{map} = \arg \max_{c \in C} P(c | d) = \arg \max_{c \in C} P(c) \prod_{1 \leq k \leq n_d} P(t_k | c) \quad (2.2)$$

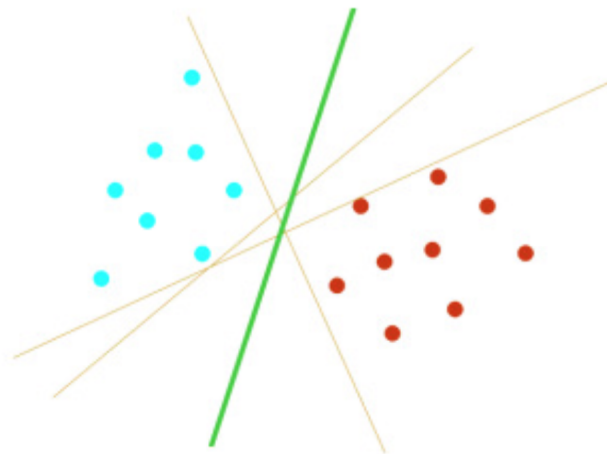
2.3.3 Máquinas de Vetores de Suporte (SVM)

Uma técnica de classificação que tem recebido considerável atenção pois esta técnica possui seus fundamentos na teoria de aprendizagem estatística e tem mostrado resultados empíricos promissores em muitas aplicações práticas, desde o reconhecimento de dígitos escritos à mão até a categorização de textos. SVM também funciona bem com dados de alta dimensionalidade e evita o

problema da dimensionalidade. Outro aspecto único desta abordagem é que ela representa o limite da decisão usando um subconjunto dos exemplos de treinamento, conhecido com vetores de suporte (TAN; STEINBACH; KUMAR, 2009).

Na Figura 4, existe um conjunto de classificadores lineares que separam duas classes, mas apenas um (em destaque) que maximiza a margem de separação (distância da instância mais próxima ao hiperplano de separação das classes). O hiperplano com margem máxima é chamado de hiperplano ótimo (JUNIOR, 2010).

Figura 4 – Conjunto de hiperplanos possíveis



Fonte: (JUNIOR, 2010).

2.3.3.1 Algoritmo SMO

O Algoritmo SMO (*Sequential Minimal Optimization*) (PLATT, 1998), é um algoritmo simples que pode resolver rapidamente o problema quadrático encontrado em SVM (*Support Vector Machines*) sem qualquer armazenamento extra de matriz e sem o uso de passos de otimização numérica.

O SMO (*Sequential Minimal Optimization*) é um algoritmo de decomposição que usa uma solução analítica para otimizar um par de Multiplicadores de

Lagrange por iteração. Em cada uma delas, três tarefas são executadas: seleção de um par de coeficientes, otimização do par de coeficientes selecionados e atualização de dados globais. A parada do algoritmo ocorre quando todos os coeficientes satisfazem o conjunto de condições. O conjunto de trabalho é formado pelo par de coeficientes a otimizar (HERNÁNDEZ, 2009).

2.3.4 Métricas para Avaliação de Modelo

A performance de um modelo de classificação pode ser medida utilizando a matriz de confusão, mostrada na Tabela 1. Nesta tabela pode ser observado que a distribuição entre as classes (proporção entre exemplos positivos e negativos) é o relacionamento entre a primeira e segunda linha. Assim, qualquer medida de desempenho que utilize valores de ambas as colunas será, necessariamente, sensível a desproporção entre as classes. Havendo mudança na distribuição das classes, os valores das métricas também mudarão, mesmo que o desempenho global do modelo não melhore (PRATI; BATISTA; MONARD, 2008).

Tabela 1 – Matriz de confusão para problemas de classes binárias

	Predição positiva	Predição negativa
Classe positiva	Verdadeiro positivo (<i>TP</i>)	Falso negativo (<i>FN</i>)
Classe negativa	Falso positivo (<i>FP</i>)	Verdadeiro negativo (<i>TN</i>)

Fonte: Adaptado de (PRATI; BATISTA; MONARD, 2008).

Uma das medidas de desempenho mais utilizadas para avaliar a qualidade dos modelos é a *precisão na classificação*, que é definida como $Acc = \frac{TP+TN}{TP+FP+FN+TN}$, ou, equivalente, a *taxa de erro na classificação*, definida como $Err = 1 - acc$.

2.4 WEKA (*Waikato Environment for Knowledge Analysis*)

A ferramenta WEKA (*Waikato Environment for Knowledge Analysis*) conforme descrito em (HALL et al., 2009), visa proporcionar uma coleção abrangente de algoritmos de aprendizagem de máquina e ferramentas de pré-processamento de dados tanto para pesquisadores como para profissionais. Permite aos usuários experimentar rapidamente e comparar diferentes métodos de aprendizagem de máquina em diferentes conjuntos de dados. Sua arquitetura modular, extensível, permite que os processos de mineração de dados pode ser construído a partir da vasta coleção de algoritmos de aprendizado e diversas ferramentas oferecidas.

2.5 Dados Abertos

Utilizamos a base de dados disponibilizada pelo portal de dados abertos da prefeitura do Município de Fortaleza de casos notificados de dengue. O portal de dados abertos de Fortaleza é um espaço desenvolvido pela Coordenadoria de Ciência, Tecnologia e Inovação da Prefeitura de Fortaleza (CITINOVA) para que a sociedade possa encontrar e utilizar os dados e informações públicas da cidade de Fortaleza. Os dados são publicados em formatos abertos que permitem sua reutilização em aplicativos digitais desenvolvidos por e para qualquer pessoa. Além disso, o portal serve como uma ferramenta de interlocução com a sociedade fortalezense para pensar e promover a inovação e a criatividade em prol da melhoria de serviços e da vida na cidade de Fortaleza. O portal de dados abertos de Fortaleza está em conformidade com os princípios da administração pública e observâncias às recomendações aceitas internacionalmente, como as emitidas pela Open Knowledge Foundation e pelo Consórcio W3C Internacional.

2.6 SINAN

O Sistema de Informações de Agravos de Notificação (SINAN) é o principal sistema de informações que tem como objetivo os dados referentes a morbidade, sendo fundamental no processo de trabalho da Vigilância em Saúde, estando envolvido não somente nas ações de Vigilância Epidemiológica, mas também na Vigilância Ambiental em Saúde e Vigilância em Saúde do Trabalhador (CONASS, 2015). O SINAN foi desenvolvido no início da década de 90, com o objetivo de padronizar a coleta e o processamento de dados sobre agravos de notificação obrigatória em todo território nacional. Construído de maneira hierarquizada, mantendo coerência com a organização do SUS (Sistema Único de Saúde), pretende ser suficiente ágil na viabilização de análises de situações de saúde em curto espaço de tempo. O SINAN fornece dados para a análise do perfil da morbidade e contribui para a tomada de decisões nos níveis municipal, estadual e federal (SAÚDE, 2008). A dengue é uma das doenças de notificação compulsória, devendo todo caso suspeito ou confirmado ser notificado ao Serviço de Vigilância Epidemiológica, por meio do SINAN nas fichas de notificação de investigação (SAÚDE, 2008).

3 Experimentos

Neste capítulo, avaliamos os modelos gerados através de uma série de experimentos. Apresentaremos a base de dados utilizada para os experimentos, a metodologia de experimentação e os experimentos em si realizados. Serão apresentados resultados obtidos na aplicação dos classificadores em 5 conjuntos de dados e as medidas resultantes observadas. O processo de classificação utiliza o conjunto de *datasets* em ordem crescente, que foram produzidos na Seção 3.2.

3.1 Base de Dados

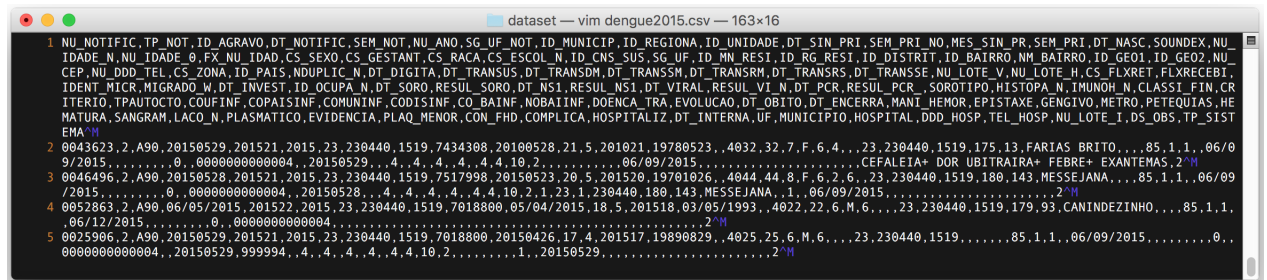
Para avaliar o objetivo geral neste trabalho, usamos a base de dados do Município de Fortaleza, disponibilizada pelo portal de dados abertos. Os dados são publicados em formatos abertos que permitem sua reutilização em aplicativos digitais desenvolvidos por e para qualquer pessoa (PREFEITURA DE FORTALEZA, 2015). Este conjunto de dados contém notificações de dengue do Município de Fortaleza no período de janeiro à junho de 2015, contendo 26.568 instâncias.

3.2 Pré-processamento

Após o download da base de dados em (PREFEITURA DE FORTALEZA, 2015), iniciamos a importação do arquivo no formato CSV (*Comma Separated Value*), conforme mostrado na Figura 5.

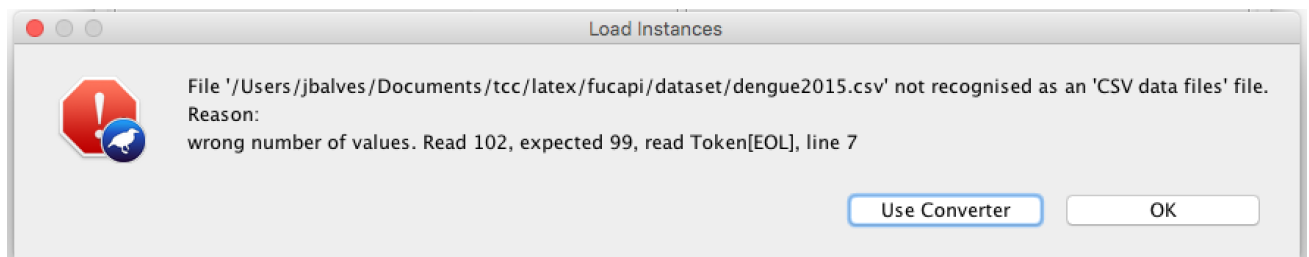
Onde nos deparamos com erros no campos DS_OBS, onde o WEKA informa que está lendo 102 atributos em vez de 99, conforme mostrado na Figura 6. Já na Figura 7, podemos observar que houve falha na inserção dos dados no atributo DS_OBS, pois o mesmo foi preenchido com vírgula em

Figura 5 – Data Set Original



Fonte: Próprio Autor.

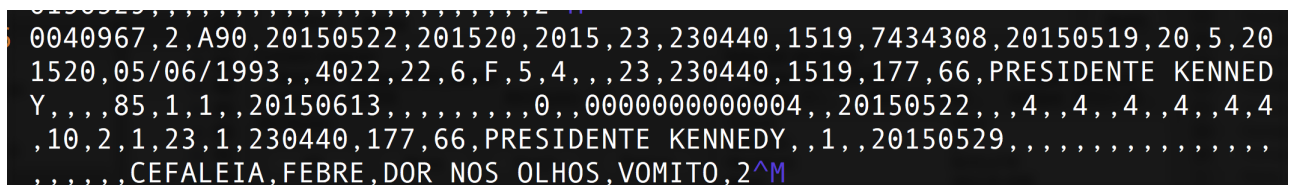
Figura 6 – Erro ao Carregar Instâncias



Fonte: Próprio Autor.

seu conteúdo da seguinte "CEFALEIA, FEBRE, DOR NOS OLHOS, VOMITO", além do caracter "M" que representa quebra de linha. Pelo fato do arquivo ser contruído no formato CSV (*Comma Separated Value*), essas vírgulas excedentes serão consideradas como início de novos atributos.

Figura 7 – Erro Mencionado pelo WEKA

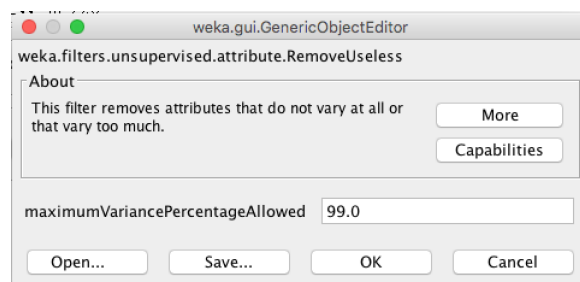


Fonte: Próprio Autor.

Para resolver este problema editamos o arquivo de tal forma que os atributos "DS_OBS" e "TP_SISTEMA" foram removidos.

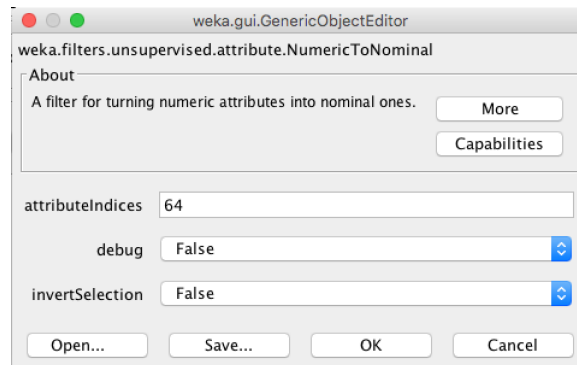
Em seguida aplicamos os seguintes filtros:

- *RemoveUseless*: Este filtro remove atributos que não variam em tudo ou que variam muito. Todos os atributos constantes são excluídos automaticamente, juntamente com qualquer que exceder o percentual máximo de parâmetro de variação. O teste de variância máxima só é aplicada aos atributos nominais: ITP_NOT, D_AGRAVO, NU_ANO, SOUNDEX, ID_GEO2, ID_PAIS, NDUPLIC_N, DT_TRANSUS, DT_TRANSDM, DT_TRANS SM, DT_TRANSRM, DT_TRANSRS, DT_TRANSSE, NU_LOTE_V, NU_LOTE_H, FLXRECEBI, MIGRADO_W, COPAISINF, MANI_HEMOR, EPISTAXE, GENGIVO, METRO, PETEQUIAS, HEMATURA, SANGRAM, LACO_N, PLASMATICO, EVIDENCIA, PLAQ_MENOR, CON_FHD e COM-PLICA. Utilizamos os parâmetros padrões conforme Figura 8.

Figura 8 – Filtro *RemoveUseless*

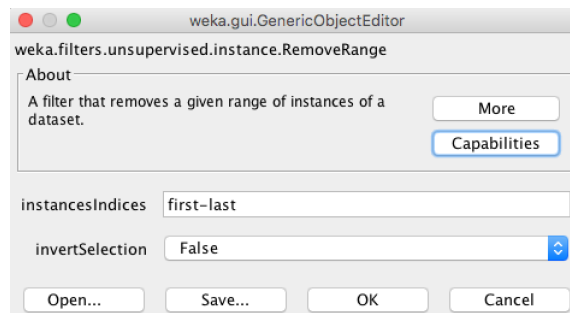
Fonte: Próprio Autor.

- Remoção manual: Pelo fato de não contribuir nos resultados os seguintes atributos foram removidos: DT_NASC, NU_DDD_TEL, DDD_HOSP, TEL_HOSP e NU_LOTE_I.
- *NumericToNominal*: Um filtro para transformar numérico atributos em uns nominais. Ao contrário da discretização, apenas toma todos os valores numéricos e os adiciona à lista de valores nominais do que atributo. Útil após a importação CSV, para impor certos atributos para se tornar nominal. Utilizamos o parâmetro mostrado na Figura 9.

Figura 9 – Filtro *NumericToNominal*

Fonte: Próprio Autor.

- *RemoveRange*: Um filtro que remove um determinado intervalo de ocorrências de um conjunto de dados. Utilizamos o parâmetro mostrado na Figura 10. O parâmetro "instancesIndices" foi utilizado cinco vezes para criarmos 05 (cinco) conjuntos de dados com as seguintes quantidades de instâncias: 5.000, 10.000, 15.000, 20.000 e 25.000. Que foram utilizados durante os experimentos.

Figura 10 – Filtro *RemoveRange*

Fonte: Próprio Autor.

3.3 Metodologia do Experimento

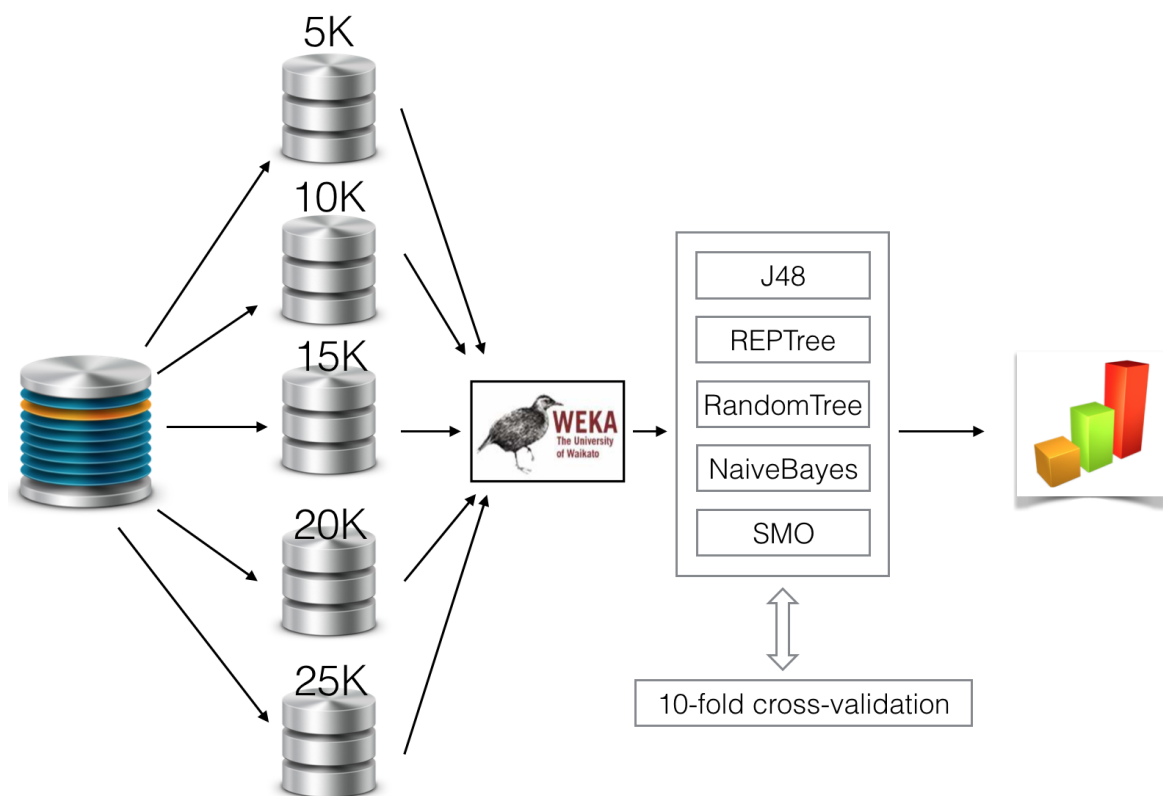
Nossos experimentos foram realizados a partir dos conjuntos de dados divididos nas seguintes quantidades de instâncias: 5.000, 10.000, 15.000, 20.000 e 25.000. A cada conjunto de instâncias, aplicamos os processos de classificação

utilizando 5 classificadores que estão contidos em 3 técnicas de classificação, conforme representado na Figura 11. Como resultado dos classificadores obtivemos 25 modelos.

Todos os testes foram feitos utilizando o processo de validação conhecido como *Cross Validation* disponibilizado pelo WEKA (*Waikato Environment for Knowledge Analysis*), com o funcionamento da seguinte forma:

- Defini-se o parâmetro *folds*, que utilizaremos com o valor 10;
- O conjunto de dados é aleatoriamente reordenado e depois divide-se em conjuntos de tamanho = *folds*;
- Em cada iteração, um conjunto é usado para testes e os outros $n - 1$ são utilizados para o treinamento do classificador;
- Os resultados do teste são coletados e calculados sobre todos os conjuntos.

Figura 11 – Metodologia Processo dos Experimentos



Fonte: Próprio Autor.

3.3.1 Parâmetros

Configuração padrão utilizada em cada algoritmo de classificação:

- J48: confidenceFactor (fator de confiança utilizado para a poda) = 0.25; minNumObj (número mínimo de instâncias por folha) = 2.
- REPTree: minNum = 2.0; minVarianceProp = 0.001; numFolds = 3; seed = 1.
- RandomTree: KValue = 0; minNum = 1.0; seed = 1;
- NaiveBayes: debug = False; displayModelInOldFormat = False; useKernelEstimator = False; useSupervisedDiscretization = False;
- SMO: c = 1.0; toleranceParameter = 0.001; epsilon = 1.0E-12; filterType = Normalize training data; kernel = PolyKernel; cacheSize = 250007; exponent = 1.0.

3.4 Resultados dos Algoritmos

Segundo (WITTEN; FRANK; HALL, 2011), o algoritmo *ZeroR* modela uma base de dados com um única regra. Onde, dado um conjunto de dados para uma nova classificação, o algoritmo prediz o valor mais frequente nos dados de treinamento. Com isso utilizou-se este algoritmo como base de referência para os demais algoritmos utilizados neste trabalho, para que seja possível compará-los entre si.

Na Tabela 2, pode-se observar que no *dataset* com 5k instâncias o melhor resultado ficou com o algoritmo *SMO* e o pior resultado no algoritmo *ZeroR*. Outro dado importante que pode ser percebido, está relação aos algoritmos: *SMO*, *J48* e *REPTree*. Pois estes algoritmos obtiveram o maior índice de classificação em todos os conjuntos de *datasets*.

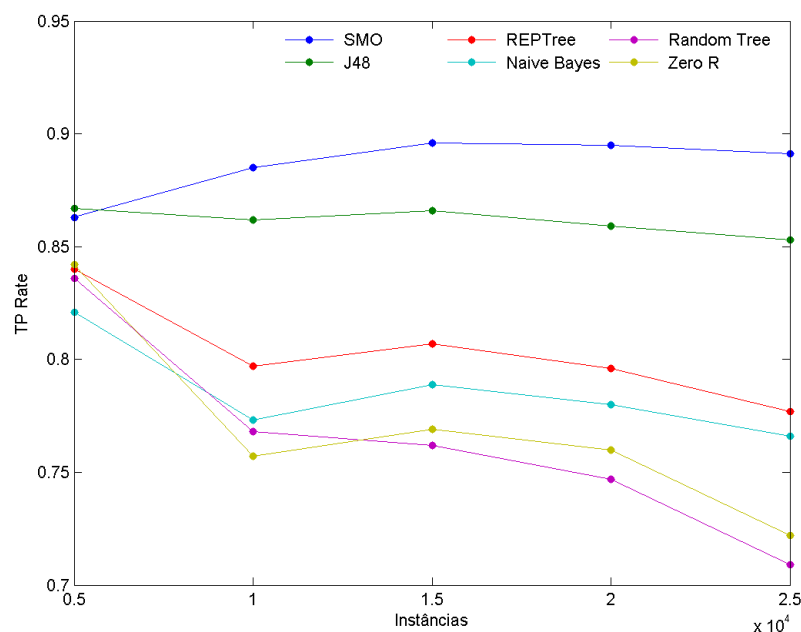
Tabela 2 – Resultados de Classificação

Classificador	Correctly Classified Instances				
	5k instâncias	10k instâncias	15k instâncias	20k instâncias	25k instâncias
SMO	89,2866%	88,5354%	89,6185%	89,4714%	89,1407%
J48	86,6530%	86,2315%	86,6322%	85,9331%	85,3096%
REPTree	83,9683%	79,7414%	80,7076%	79,6344%	77,7052%
NaiveBayes	83,5592%	76,8103%	76,1682%	74,9666%	70,8689%
RandomTree	82,0762%	77,2906%	78,8551%	78,0289%	76,6155%
ZeroR	84,2496%	75,6773%	76,8608%	75,9664%	72,1555%

Fonte: Próprio Autor.

Na Figura 12, pode-se visualizar estas informações observando a *taxa de verdadeiros positivos* de cada modelo gerado. Observa-se que houve uma certa redução na precisão de classificação dos *verdadeiros positivos* conforme o tamanho do *dataset* aumenta nos algoritmos: *REPTree*, *NaiveBayes*, *ZeroR* e *RandomTree*.

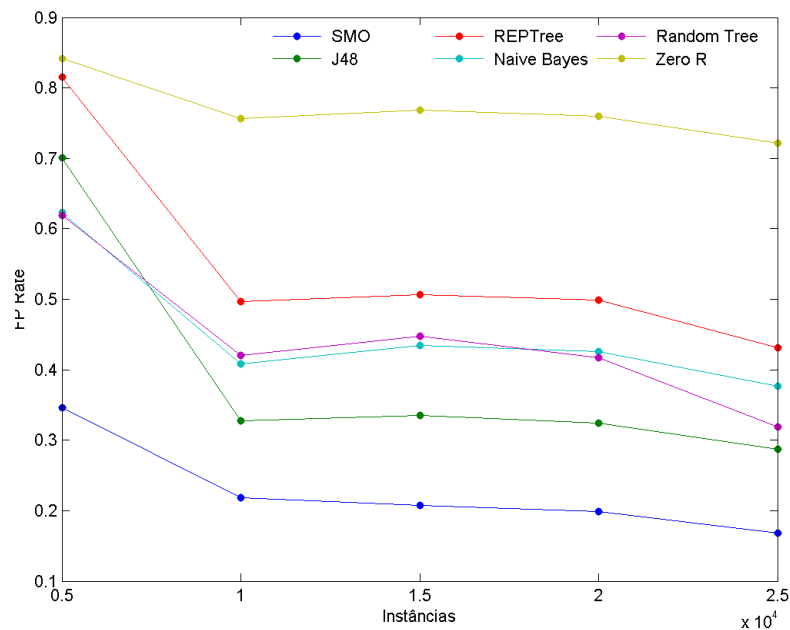
Figura 12 – TP Rate



Fonte: Próprio Autor.

Na Figura 13, pode confirmar que os algoritmos com melhor desempenho *SMO* e *J48* mostrados na Figura 12, apresentam menor *taxa de falso positivo* e com isso um menor custo no processo de classificação de novas instâncias, pois estes algoritmos obtiveram menor quantidade de erros nos *datasets* utilizados.

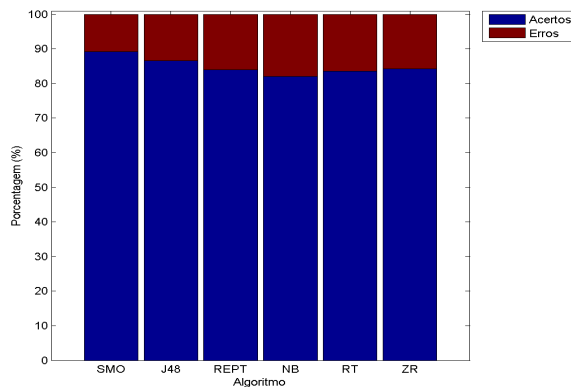
Figura 13 – FP Rate



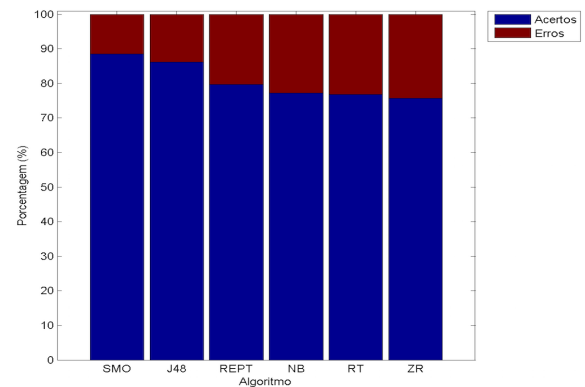
Fonte: Próprio Autor.

Observando a Figura 14, pode-se verificar que em todos os conjuntos de dados utilizados nos experimentos, os algoritmos *SMO* e *J48* se destacam em relação aos demais. Com isso os modelos de classificação gerados por esses algoritmos se desempenharam com índices próximos a 90% no quesito predição de novas instâncias.

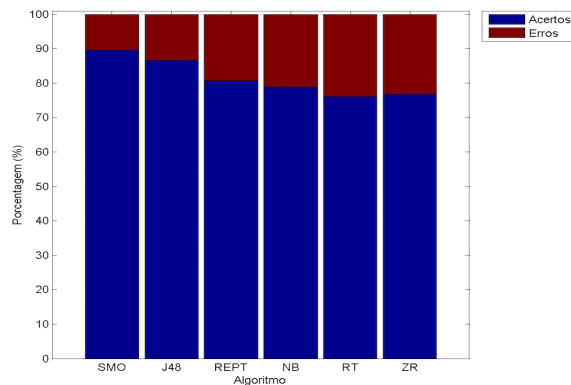
Figura 14 – Gráficos de Acertos e Erros de Classificação



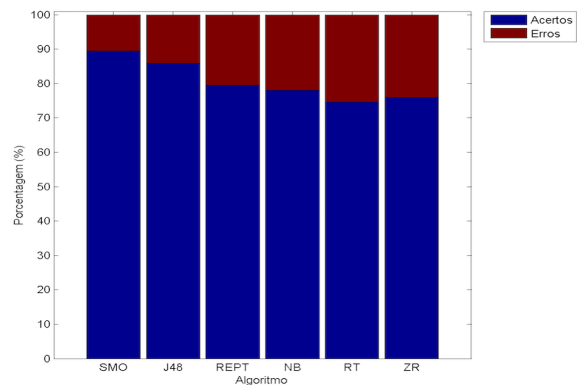
(a) 5k Instâncias



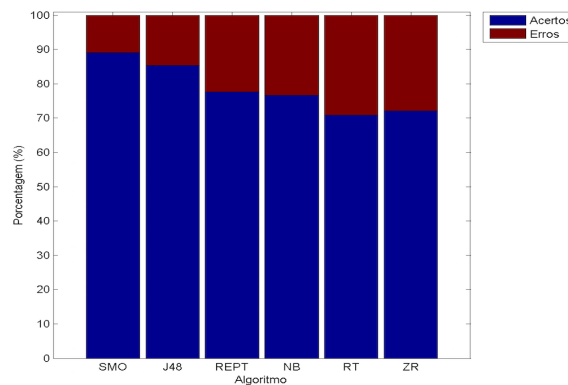
(b) 10k Instâncias



(c) 15k Instâncias



(d) 20k Instâncias



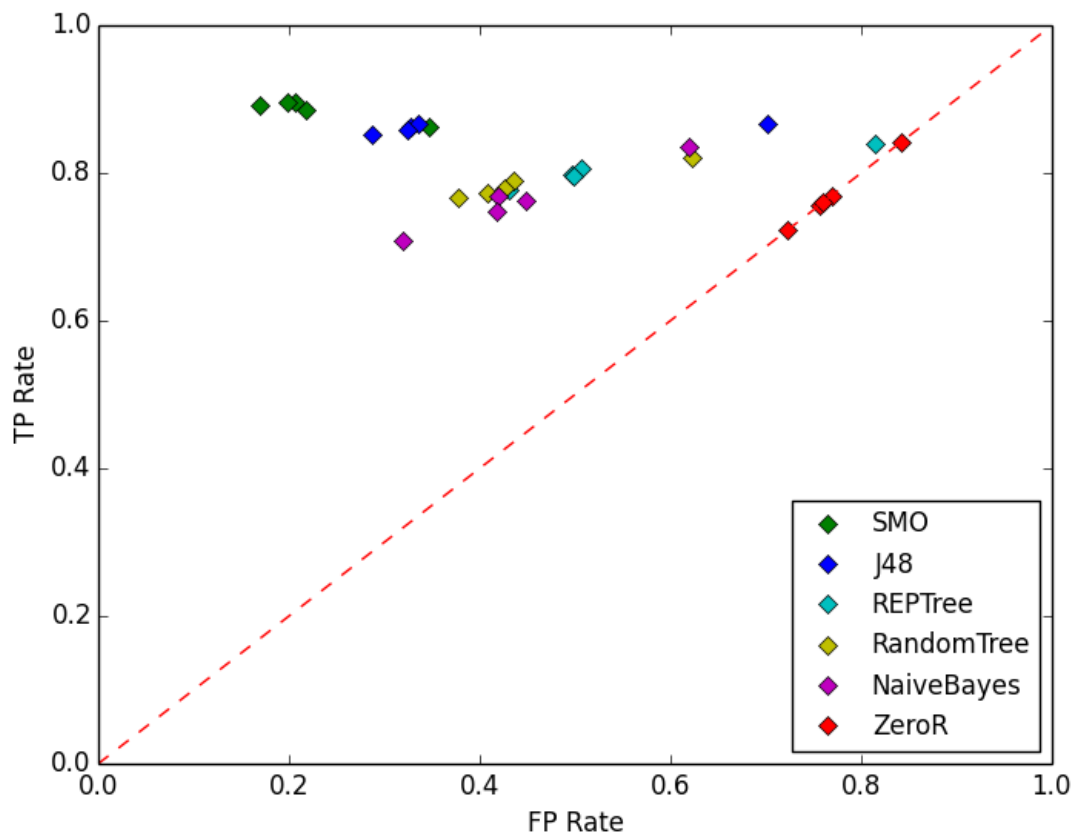
(e) 25k Instâncias

Fonte: Próprio Autor.

Em (PRATI; BATISTA; MONARD, 2008), gráfico ROC (*Receiver Operating Characteristic*) é um método para avaliação e seleção de sistemas de diagnóstico e/ou predição. O gráfico ROC é baseado na probabilidade de detecção ou *taxa de verdadeiros positivos*, e na probabilidade de falsos alarmes, ou *taxa de falsos positivos*.

Como pode-se observar na Figura 15, os modelos gerados pelo algoritmos *SMO* e *J48* obtiveram melhor qualidade, pois se aproximam do ponto $(0, 1)$. Em (PRATI; BATISTA; MONARD, 2008), resultados pertencentes ao canto superior esquerdo representam classificadores que possuem melhor desempenho.

Figura 15 – Modelos de Classificação no Espaço ROC



Fonte: Próprio Autor.

4 Conclusão e Trabalhos Futuros

Neste Capítulo, apresentado as conclusões do nosso trabalho, incluindo suas limitações e futuros estudos que possam ser explorados.

4.1 Conclusão

O Objetivo deste estudo foi construir modelos de classificação por meio do uso de técnicas em *Data Mining* conhecidas como: *Árvore de Decisão*, *Teorema de Bayes* e *Máquinas de Vetores de Suporte*. Estas técnicas foram aplicadas por meio dos algoritmos: *SMO*, *J48*, *REPTree*, *RandomTree* e *NaiveBayes*.

As três técnicas apresentadas e testadas neste trabalho: Árvores de Decisão, Bayes e SVM (*Support Vector Machine*), mostraram ser técnicas aplicáveis em bases de dados oriundas do SINAN (Sistema de Informações de Agravos de Notificação), obtendo resultados entre 70% e 90% na classificação.

Apesar da técnica SVM (*Support Vector Machine*) em sua formulação original tem o objetivo de lidar apenas com problemas de classificação binários, na aplicação do algoritmo SMO (*Sequential Minimal Optimization*) neste trabalho observou-se os melhores resultados em todas os conjuntos de *datasets* utilizados nos experimentos.

O Algoritmo REPTree mostrou maior custo na classificação, pois sua *taxa de falsos positivos* foi a maior em comparação aos algoritmos RandomTree e J48, apesar de realizarem a mesma técnica de classificação Árvores de Decisão.

Com isso foi possível observar a viabilidade da aplicação de técnicas de *Data Mining* em conjuntos de dados com informações de agravos de Dengue.

4.2 Trabalhos Futuros

Como trabalhos futuros, podem ser explorados na continuação desta pesquisa as oportunidades descritas a seguir:

- Implementar remoção de instâncias que influenciam na qualidade do classificador.
- Adaptar a construção de modelos em tempo real.
- Integrar informações de precipitação de chuva da região onde os dados foram coletados.
- Testar modelos gerados em bases de outros estados.
- Usar outros algoritmos de classificação.

Referências

- BELLAZZI, R.; ZUPAN, B. Predictive data mining in clinical medicine: current issues and guidelines. *International journal of medical informatics*, Elsevier, v. 77, n. 2, p. 81–97, 2008.
- BERRY, M. J.; LINOFF, G. *Data mining techniques: for marketing, sales, and customer support*. [S.l.]: John Wiley & Sons, Inc., 1997.
- BHATT, S. et al. The global distribution and burden of dengue. *Nature*, Nature Publishing Group, a division of Macmillan Publishers Limited. All Rights Reserved., v. 496, n. 7446, p. 504–507, 04 2013. Disponível em: <<http://dx.doi.org/10.1038/nature12060>>.
- CONASS. *Nota Técnica - 45 - 2013*. 2015. Disponível em: <<http://www.conass.org.br>>.
- COSTA, A. F. *Mineração de imagens médicas utilizando características de forma*. Tese (Doutorado) — Universidade de São Paulo, 2012.
- DAMASCENO, M. Introdução a mineração de dados utilizando o weka. *Instituto AAAFederal de Educação, Ciência e Tecnologia do Rio Grande do Norte*, 2005.
- FAYYAD, U.; PIATETSKY-SHAPIO, G.; SMYTH, P. From data mining to knowledge discovery in databases. *AI magazine*, v. 17, n. 3, p. 37, 1996.
- FELDENS, M. A.; CASTILHO, J. M. V. d. Engenharia da descoberta de conhecimento em bases de dados: estudo e aplicação na área de saúde. *Porto Alegre*, 1997.
- FREITAS, A. Uma introdução a data mining. *Informática Brasileira em Análise, Centro de Estudos e Sistemas Avançados do Recife (CESAR), Recife, Pe, ano II*, n. 32, 2000.
- HALL, M. et al. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, ACM, v. 11, n. 1, p. 10–18, 2009.
- HALMENSCHLAGER, C. *Um algoritmo para indução de árvores e regras de decisão*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, 2002.
- HERNÁNDEZ, R. A. *MP-SMO: um algoritmo para a implementação VLSI do treinamento de máquinas de vetores de suporte*. Tese (Doutorado) — Universidade de São Paulo, 2009.
- INITIATIVE, O. S. 2015. Disponível em: <<https://opensource.org/definition>>.
- JUNIOR, G. M. d. O. Máquina de vetores suporte: estudo e análise de parâmetros para otimização de resultado, 2010. *Trabalho de Graduação em Ciência da Computação*, 2010.

LORENA, A. C.; CARVALHO, A. C. de. Uma introdução às support vector machines. *Revista de Informática Teórica e Aplicada*, v. 14, n. 2, p. 43–67, 2007.

LUCCA, G. et al. Uma implementação do algoritmo naïve bayes para classificação de texto. 2013.

ORACLE. 2015. Disponível em: <https://www.java.com/pt_BR/download/faq/whatis_java.xml>.

OSHIRO, T. M. *Mestre em Bioinformática*. Tese (Doutorado) — Universidade de São Paulo, 2013.

PLATT, J. *Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines*. [S.l.], 1998. 21 p. Disponível em: <<http://research.microsoft.com/apps/pubs/default.aspx?id=69644>>.

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

PREFEITURA DE FORTALEZA. Notificações de dengue - ref. 01 a 06/2015. 2015. Disponível em: <<http://dados.fortaleza.ce.gov.br/catalogo/dataset/notificacoes-de-dengue-ref-01-06-2015>>. Acesso em: 19 out. 2015.

SALZBERG, S. C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Machine Learning*, Kluwer Academic Publishers, v. 16, n. 3, p. 235–240, 1994. ISSN 0885-6125. Disponível em: <<http://dx.doi.org/10.1007/BF00993309>>.

SANTOS, M. S.; NETO, J. C. da C. Amagodis: Algoritmos de mineração para apoio à gerência de ocorrências de dengue a partir de informações presentes na base dados do sinan. 2011.

SAÚDE, M. da. Vigilância em saúde: Dengue, esquistossomose, hanseníase, malária, tracoma e tuberculose. *Departamento de Atenção Básica*, 2008.

SAÚDE, M. da. Dengue: diagnóstico e manejo clínico: adulto e criança. *Departamento de Atenção Básica*, 2013.

SAÚDE, M. da. 2015. Disponível em: <<http://portalsaude.saude.gov.br/index.php/o-ministerio/principal/secretarias/svs/dengue>>.

SHAKIL, K. A.; ANIS, S.; ALAM, M. Dengue disease prediction using weka data mining tool. *arXiv preprint arXiv:1502.05167*, 2015.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. *Introdução ao datamining: mineração de dados*. [S.l.]: Ciência Moderna, 2009.

THITIPRAYOONWONGSE, D.; SURIYAPHOL, P.; SOONTHORNPHISAJ, N. Data mining of dengue infection using decision tree. *Entropy*, v. 2, p. 2, 2012.

THOMÉ, A. C. G. Redes neurais: uma ferramenta para kdd e data mining. *Material Didático* http://equipe.nce.ufri.br/thome/grad/nn/mat_didatico/apostila_kdd_mbi.pdf, Outubro, 2002.

WAIKATO, T. U. of. 2015. Disponível em: <<http://www.cs.waikato.ac.nz/ml/weka/>>.

WITTEN, I. H.; FRANK, E.; HALL, M. A. *Data Mining: Practical Machine Learning Tools and Techniques*. 3rd. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011. ISBN 0123748569, 9780123748560.

ZHAO, Y.; ZHANG, Y. Comparison of decision tree methods for finding active objects. *Advances in Space Research*, Elsevier, v. 41, n. 12, p. 1955–1959, 2008.