PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE ENGENHARIA
PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Bruno Fernandes Chimieski

Solução de Auxílio ao Diagnóstico e à Pesquisa Médica Baseada em Mineração de Dados Utilizando Interface Android

Porto Alegre – Rio Grande do Sul
8 de janeiro de 2013

Solução de Auxílio ao Diagnóstico e à Pesquisa Médica Baseada em Mineração de Dados Utilizando Interface Android

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Pontifícia Universidade Católica do Rio Grande do Sul, como parte dos requisitos para a obtenção do título de Mestre em Engenharia Elétrica.
Área de concentração: Sinais, Sistemas e Tecnologia da Informação
Linha de Pesquisa: Sistemas de Informação

Orientador: Prof. Dr. Rubem Dutra Ribeiro Fagundes

Porto Alegre – Rio Grande do Sul

8 de janeiro de 2013

Agradecimentos

À Deus, pela Vida!

À Jesus Cristo, pelo exemplo de fé!

À São Jorge Guerreiro, pela companhia nessa árdua, porém vitoriosa jornada. Salve Jorge!

Aos meus pais, pela educação e pelo companheirismo e carinho de sempre!

Ao professor Rubem, pela orientação nesse trabalho.

Aos colegas Diego Santos, Nicolas Marroni e Tiago Noronha, pela camaradagem durante o curso de Mestrado, e pela ajuda nas questões sobre o LaTeX.

"Concedei-nos, Senhor, a Serenidade necessária
para aceitar as coisas que não podemos modificar,
Coragem para modificar aquelas que podemos,
e Sabedoria para distinguir umas das outras."
Reinhold Niebuhr

Resumo

Desde os estudos primordiais sobre as aplicações da Tecnologia da Informação objetivando agregar valor a outras áreas do conhecimento, o campo de atuação da Medicina sempre foi visto como terreno fértil para tal. Com o advento das técnicas de Inteligência Artificial, os programas de computador passaram a ter um poderio de aprendizagem mais sofisticado e, portanto, abrindo a possibilidade da sua utilização além dos processos administrativos hospitalares, chegando cada vez mais próximo da prestação de cuidados aos pacientes. Por isso, a presente dissertação propõe-se a demonstrar a viabilidade de uma solução de auxílio ao diagnóstico médico e à obtenção de conhecimento implícito em bases de dados de três doenças: tumor de mama, problemas dermatológicos e da coluna vertebral. Para tanto, aplica-se o processo de extração de conhecimento de bases de dados a fim de atingir esses objetivos. Esse processo tem como cerne o uso da Mineração de Dados, que por sua vez, apóia-se nos algoritmos de aprendizado de máquina para transformar dados em informações úteis para os negócios a que se referem. Por isso, esse trabalho apresenta um estudo, auxiliado pela ferramenta *Weka*, para a determinação de quais os algoritmos de aprendizado de máquina apresentam melhor desempenho quando aplicados às bases de dados alvo. Com esses algoritmos em mãos, implementou-se uma solução de auxilio ao diagnóstico e estudo médico fazendo uso de aplicativos *Android* como *interface* de utilização para os profissionais de saúde, com isso, utilizando o que há de mais moderno em termos de dispositivos eletrônicos móveis no mercado mundial. Os resultados foram bastante satisfatórios, dado que os objetivos traçados referentes ao estudo sobre a determinação de algoritmos de Mineração de Dados, à preparação das bases de dados para futuras pesquisas e à implementação da solução de auxílio ao diagnóstico foram atingidos e, em conjunto, comprovam que é possível aplicar ferramentas da Tecnologia da Informação para agregar valor à prática médica.

Palavras-chave: Mineração de Dados, Android, Weka, Extração de Conhecimento de Bases de Dados, classificação, associação, classificadores Bayesianos, árvores de decisão, *BayesNet*, *Functional Trees*.

Abstract

Since the primary studies on the applications of Information Technology aiming to add value to other areas of knowledge, the playing field of medicine has always been seen as fertile ground for such. With the advent of Artificial Intelligence techniques, computer programs have been given a power of learning more sophisticated and thus opening the possibility of its use beyond the hospital administrative processes, drawing ever closer to the provision of patient care. Therefore, this paper proposes to demonstrate the feasibility of an aid to medical diagnosis and obtaining implicit knowledge in databases of three diseases: breast cancer, dermatology and vertebral column problems. To do so, is applied the process of extracting knowledge from databases in order to achieve these goals. This process has Data Mining as its core, which in turn relies on machine learning algorithms to transform data, sometimes not analyzed, in useful information for business referred to, in this case about health care. Therefore, this work presents a study aided by the tool Weka, to determine which machine learning algorithms perform best when applied to target databases. With these algorithms in hand, is implemented a solution to aid the diagnosis and study of medical applications making use of Android as interface for healthcare professionals, with it, utilizing what is most modern in terms of mobile electronic devices in the world market. The results were quite satisfactory, given that the objectives for the study on the determination of Data Mining algorithms, preparation of databases for future research and implementation of the solution for the diagnosis have been met and, together, prove that you can apply tools of information technology to add value to medical practice.

Keywords: Data Mining, Android, Weka, Knowledge Discovery on Databases, classification, association, Bayesian classifiers, decision trees, BayesNet, Functional Trees

Sumário

Página

Lista de Figuras

Lista de Tabelas

Lista de Acrônimos

- API - *Application Programming Interface*
- DSS - *Decision Suppport Tools*
- EHR - *Electronic Health Record*
- EMR - *Electronic Medical Record*
- FT - *Functional Trees*
- KDD - *Knowledge Discovery in Databases*
- LANs - *Local Area Networks*
- LMT - *Logistic Model Tree*
- ROC - *Receiver Operating Characteristic*

1 Introdução

1.1 Apresentação

Os hospitais foram, desde cedo, um ponto importante para a aplicação de processos computadorizados. A história dos sistemas de informação hospitalares remonta à década de 60 e teve como empresas pioneiras naquele momento a IBM e uma companhia conhecida na época como *Burroughs*, que futuramente passou por uma fusão com a *Sperry Rand* para juntas se tornarem a atualmente conhecida *Unisys*. Existiam também algumas empresas de *software* para Medicina de pequeno porte que se originaram nos anos 60 e 70, tais como a IDX, a *Shared Medical Systems* e a *Meditech*. Cabe salientar que tais companhias desenvolviam programas para o controle das operações financeiras envolvidas no atendimento aos pacientes. As possibilidades de aplicação da Computação na Medicina foram observadas ainda nos primórdios da Computação, e as aplicações de Inteligência Artificial para tal área foram desenvolvidas bem cedo. Artigos podem ser encontrados datados do final dos anos 50 tratando justamente do potencial de uso das áreas de conhecimento da Eletrônica para a Medicina [1]. Algumas instituições médicas, assim como contribuintes individuais, desenvolveram programas de computação orientados à Medicina nos anos 60 e 70, relacionados à triagem de pacientes e à avaliação de saúde dos mesmos. Estes foram desenvolvidos na empresa *Kaiser Permanente* na Califórnia, no hospital *Latter Day Saints* em *Salt Lake City* e pela *Medline*. Existiram outros projetos orientados ao ambiente clínico hospitalar, todavia o desenvolvimento corporativo de *softwares* médicos foi direcionado fortemente para o controle do faturamento derivado dos atendimentos aos pacientes [2]. Tais sistemas de faturamento foram implementados em computadores *mainframe* e em linguagens de programação projetadas para o gerenciamento de registros de pacientes, por exemplo, o COBOL. Em alguns casos, os computadores que executavam as tarefas de faturamento estavam localizados *off-site* e, dessa forma, múltiplos computadores compartilhavam o acesso à computadores centrais através de linhas telefônicas dedicadas. Informações precisas sobre o estado do hospital e

da localização dos pacientes foram um complemento necessário para a informatização do faturamento, de maneira que a função ADT (sigla em inglês para *Admission*, *Discharge*, *transference*) dos pacientes foi informatizada rapidamente. Grandes hospitais, por sua vez, desenvolveram departamentos de processamento de dados internos com pessoal dedicado à transcrição de informações dos pacientes para cartões perfurados [2]. O próximo grande desenvolvimento dos sistemas de informação hospitalar se deu com o advento dos computadores modulares dedicados às funções laboratoriais e radiológicas. Os primeiros produtos para laboratórios eram usados para reportar informações laboratoriais para os locais de análises clínicas através de terminais de vídeo conectados a computadores *mainframes* também localizados no interior do hospital. As informações relacionadas à identificação do paciente e localização eram mantidas em computadores e retransmitidas para laboratórios, setores de radiologia e farmácia através de interfaces customizadas [2]. Essa abordagem modular pode ser contrastada com a abordagem tudo-em-um, na qual uma companhia provia uma solução abrangente para todas as necessidades do hospital. A abordagem modular permitiu ao cliente selecionar a melhor solução para cada módulo, mas sofreu com a necessidade de desenvolvimento de interfaces customizadas entre os módulos, já que estas apresentaram as dificuldades inerentes à integração de sistemas de computação. Portanto, as incompatibilidades e a necessidade de negociação com dois ou mais fornecedores colocou os clientes em desvantagem devidos ao alto custo e lentidão de seus processos de desenvolvimento o que também resultou na formação de interfaces frágeis, complicadas e que utilizavam muitos recursos. Com o advento dos microcomputadores, redes primitivas começaram a se desenvolver e passou a se estabelecer a comunicação entre computadores à redes locais, que conhecemos como *LANs* atualmente. No entanto, escritórios individuais permaneceram isolados e poucos deles tinham conexões a outros computadores em rede. Esse período marca o início do que é conhecido hoje como sistema de informação clínica, no qual computadores orientados para uso clínico são conectados a provedores utilizando terminais. No final dos anos 80, os computadores pessoais começaram a substituir os terminais de vídeo nos locais de serviço e passaram a ser usados de forma generalizada como equipamentos de escritórios pelos provedores. Esses computadores dos escritórios foram conectados à sistemas de faturamento a fim de centralizar o agendamento e o próprio faturamento referentes ao pacientes [2]. Durante a primeira parte da década de 90, muitos hospitais estavam conectados via cabo, o que marcou o início das intranets hospitalares. Computadores pessoais começaram a ser empregados em locais de serviço, tais como postos de enfermagem, e as intranets passaram a prover acesso à *Internet*. Simultaneamente, subsistemas do sistema de informações hospi-

talar começaram a ser mais sofisticados. Então, a complexidade e o custo da computação médica cresceram exponencialmente depois que os computadores pessoais e as redes foram largamente empregados no ambiente hospitalar. Os textos de [2] e [3], apoiam o que foi dito anteriormente, além de corroborar com as análises da última década, na qual os sistemas de informação hospitalar tornaram-se malhas complexas de entrelaçamento e sobreposição de subcomponentes. Máquinas de *Front-End* tais como analisadores de laboratórios e *scanners* de radiografia foram informatizados. As informações dessas máquinas são repassadas para máquinas de processamento e roteamento e, posteriormente, para provedores de dados. Adicionalmente aos sistemas clínicos, sistemas de informações de saúde modernos possuem todos os componentes de um típico sistema de negócio. Ou seja, o mesmo possui servidores de arquivos nos quais os usuários guardam seus documentos, servidores de *e-mail* através dos quais os usuários enviam e recebem *e-mails*, servidores *web* que hospedam os *websites* da organização, servidores de bancos de dados que guardam informações em formato pesquisável e acesso remoto. A figura 1 mostra como exemplo a organização de um sistema de informação hospitalar integrado [4].

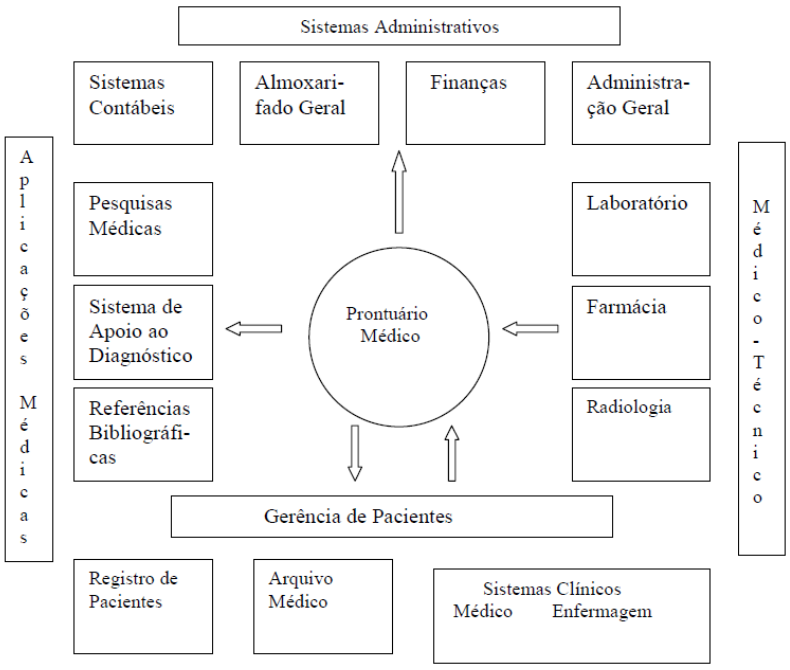

Figura 1: Sistema de Informação Hospitalar Integrado [4]

1.1.1 Registro Médico Eletrônico

Conforme os autores de [2] e [5], o registro médico eletrônico constitui-se de uma interface para o clínico através da qual o mesmo tem acesso a todas as informações do pa-

ciente para visualização, de forma agregada em um único portal. Ou seja, a partir de uma mesma tela de aplicação é possível mostrar a localização do paciente (derivada da função ADT), os resultados laboratoriais, os resultados radiológicos e as informações sobre os medicamentos do paciente. O registro médico eletrônico irá inevitavelmente num futuro próximo substituir os registros em papel utilizados hoje em dia na maior parte dos hospitais e clínicas de saúde. Por isso, esse tem sido objeto de ampla discussão e atenção por parte dos setores industriais e governamentais [2]. O registro eletrônico tem sido denominado de várias maneiras, tais como registro médico informatizado, registro informatizado do paciente, registro médico eletrônico e, mais recentemente, registro eletrônico de saúde ou EHR(do inglês, *Electronic Health Record*). Esses registros possuem dados divididos em diferentes seções tais como a administrativa, a laboratorial, a de segmento de processo e na maioria das vezes a manipulação de gráficos é feita por funcionários especializados. O estudo *Rand* de 2005, intitulado "A Difusão e o Valor da Tecnologia da Informação nos Cuidados de Saúde", resultou em várias descobertas importantes. A primeira delas é que o processo de informatização na área de cuidados médicos começou tardiamente em relação a outras indústrias, mas atualmente cresce numa taxa consistente em relação às demais. Os EHRs estão finalmente adentrando o mercado, rapidamente. O estudo também afirma que 30% dos hospitais de cuidados severos tinham adquirido EHRs até a conclusão de 2003. Por fim, numa projeção de futuro até 2016, são esperados que 80% dos hospitais tenham registros informatizados em utilização. Por isso, um significante aumento de produtividade pode ser esperado para a indústria médica, desde que essas projeções de adoção e emprego de EHRs efetivamente ocorram. Outro ponto importante a salientar é que a partir desse aumento de utilização dos EHRs, possam ser integrados aos mesmos, ferramentas de auxílio à decisão clínica, provendo conhecimento sobre doenças, tratamentos, interações medicamentosas e perfis de risco [2]. Os primeiros sistemas desenvolvidos para assistir à decisão médica datam dos primeiros anos da pesquisa de Inteligência Artificial, nos anos 70 e 80, pois desde essa época a prática da Medicina era reconhecida como um campo de aplicação rico para a área de tecnologias de Inteligência Artificial. Dois destes sistemas desenvolvidos merecem menção especial. O primeiro foi desenvolvido na *University of Pittsburgh*, pelo Dr. Jack Myers e pelo Engenheiro de Computação Harry Pople, assim como o Dr. Randall Miller, e foi chamado de INTERNIST-1. Esse sistema foi projetado em 1974 para tratar de problemas de diagnósticos nos campos da Medicina Interna e da Neurologia. Outro sistema de suporte à decisão, o MYCIN, foi desenvolvido pelo Dr. Edward Shortliffe na *Stanford University* nos anos 70. Quando devidamente integradas a outros componentes de cuidado de pacientes, as ferramentas de auxílio à decisão

representam um grande trunfo na entrega de uma Medicina segura, efetiva em termos de custo e baseada em evidências aos pacientes. Além disso, os EHRs podem prover acesso imediato a informações sobre a população para aplicações administrativas e de pesquisa. No entanto, existem algumas barreiras para o uso generalizado de EHRs, tal como o fato de muitos médicos terem pouco contato com as novas tecnologias da área da informática e, por isso, poderão ter seu desempenho piorado ao utilizarem processos informatizados para a realização de tarefas já rotineiras para os mesmos. Esta é a maior barreira num momento no qual a demanda por eficiência vem crescendo substancialmente. Os custos necessários para o *hardware*, o treinamento e a manutenção desses EHRs tendem a ser substanciais e o retorno de investimento leva algum tempo para ser observado. A primeira e mais óbvia categoria de EHR é o de gerenciamento de informação do paciente e de seus dados clínicos. Isso envolve o histórico do paciente, os diagnósticos, as alergias, os dados demográficos, os resultados de testes, sendo estes últimos alvos de análise no desenvolvimento dessa dissertação. Sendo assim, percebe-se que o uso desses EHRs permite que se tenha o benefício do uso de sistemas de auxílio à decisão, os quais, por exemplo, podem gerar e enviar lembretes sobre as condições atuais do paciente aos médicos a fim de que estes tomem as providências cabíveis para solucionar eventuais problemas de emergência ou de prevenção de complicações futuras. Atualmente ferramentas de auxílio à decisão têm sido usadas na prevenção, na prescrição de medicamentos, nos diagnósticos e no gerenciamento de doenças. A aderência às práticas preventivas nas áreas de vacinação, de triagem e de redução de risco de doenças cardiovasculares tem tido melhorias de desempenho quando essas ferramentas de auxílio à decisão são integradas às tarefas de cuidado dos pacientes [2]. Mais recentemente, técnicas de aprendizado de máquina, tais como Redes Neurais, vêm sendo utilizadas para detectar enfartos do miocárdio, tumor de mama, câncer de colo do útero e surtos de doenças nosocomiais, estas popularmente conhecidas como infecções hospitalares. Cabe salientar nesse ponto, que a abordagem de Redes Neurais não será foco deste trabalho de pesquisa, visto que estas apresentam uma estrutura complexa, um longo período de treinamento, e baixa interoperabilidade. Em outras palavras, embora precisas e robustas para muitos domínios, os modelos gerados pelas Redes Neurais são geralmente considerados incompreensíveis. Para se extrair regras de relacionamento a partir do modelo de Rede Neural, é preciso um tratamento bem mais complexo, para extrair a opacidade do modelo, conforme mostrado em [6], [7] e [8]. Por isso, a abordagem mais comum é a utilização de algoritmos de aprendizagem que são capazes de gerar modelos interpretáveis, por exemplo, as árvores de decisão [2].

1.2 Objetivos da Dissertação

Em linhas gerais, a presente dissertação tem por finalidade aplicar técnicas de Inteligência Artificial sobre bases de dados da área médica, permitindo que se possa extrair das mesmas conhecimentos implícitos que possam servir de auxílio para os profissionais da área médica na execução das suas atividades. Esse grande objetivo geral é dividido em três objetivos menores, sendo:

- Objetivo 1: Estudo e desenvolvimento de Mineração de Dados a partir de bases de dados médicas, com o objetivo de extrair novos e relevantes conhecimentos sobre os procedimentos e a prática médica. Ou seja, nesse estudo são focados os algoritmos de classificação e de associação de dados, que, respectivamente, serão aplicados na predição de diagnósticos e na extração de regras de conhecimento das bases médicas.
- Objetivo 2: Preparação das bases de dados obtidas no objetivo 1 para utilização pela comunidade científica, visando a inclusão ou atualização dos dados existentes, bem como a realização de consultas.
- Objetivo 3: Desenvolvimento de um aplicativo de auxílio ao diagnóstico em *Android*, utilizando as leis de relacionamento e, mais genericamente, o conhecimento extraído no objetivo 1 a fim de melhorar, tanto em eficiência, quanto em precisão, o atendimento médico, ou ainda a prática médica.

1.3 Justificativa

De acordo com [2], o treinamento médico tradicional requer a memorização de vastos conjuntos de informações, enquanto que a aplicação dessas informações varia conforme a demanda existente nas mais diversas situações da rotina de atendimento aos pacientes. Todavia, as informações médicas têm se tornado cada vez mais complexas e suscetíveis a mudanças tão rapidamente, que é praticamente impossível se manter atualizado. Muitos estimam que a maior parte das informações absorvidas pelos graduandos de Medicina se tornam obsoletos em aproximadamente uma década. A partir disso, para tentar minimizar esses efeitos, os sistemas informatizados de suporte à decisão podem ser empregados com sucesso. Isso porque tais ferramentas podem ser atualizadas pelas pessoas ou tecnologias dedicadas a cada tarefa, e os médicos podem se utilizar destas para atualizarem-se buscando informações por demanda sobre o conhecimento atual. Conforme dito anteriormente, a informatização do ambiente clínico tem ocorrido de forma relativamente lenta

em relação a outros setores da indústria. A base necessária para a adoção generalizada dos DSSs (*Decision Support Systems*) é o uso do *software* sobre o qual ele tipicamente é executado, que são os EHRs. Se os DSSs não forem integrados no fluxo de trabalho juntamente com esses EHRs, os médicos e os demais profissionais da saúde não se sentirão confortáveis a fazer uso deles. Um especialista sobre a difusão da tecnologia de suporte à decisão declarou sucintamente que "se uma tecnologia é facilmente assimilada na prática existente, será rapidamente adotada; se ela interrompe as atividades diárias, a organização social, ou o *status quo*, ela não será adotada". Tal declaração não significa que os DSSs devem atuar como *stand-alone software*, como foram nos primeiros sistemas de diagnóstico previamente citados, mas sim, de que é no ponto de interação entre o médico e o computador que o uso dos DSSs se torna mais efetivo e mais suscetível a ser adotado. Sistemas de suporte à decisão bem sucedidos precisam economizar tanto tempo quanto dinheiro. A economia de tempo age diretamente como incentivo aos médicos, ao passo que as reduções nos custos serão bem vistas pelos compradores desses sistemas, que podem ser médicos, hospitais ou clínicas. Existem poucos dados quanto aos benefícios dos DSSs, mas estudos têm mostrado que sistemas informatizados podem melhorar o desempenho clínico e afetar positivamente os resultados para os pacientes. De acordo com [5], por meio de entrevistas e questionários, as equipes dos hospitais enfatizam a importância de localizar colegas e recursos, que as trocas de comunicação dependem do contexto, e que acreditam que o uso de computadores portáteis é um mecanismo apropriado para acessar os dados médicos. Esses benefícios tendem a ocorrer a partir das intervenções dos DSSs relativas à dosagem e à prescrição de medicamentos, à assistência ao diagnóstico, aos lembretes sobre cuidados preventivos, e aumentaram a aderência aos protocolos clínicos ou de boas práticas. Existem também evidências de que os DSSs podem aprimorar a eficiência dos cuidados através da redução do tempo gasto pelos médicos em tarefas administrativas e do tempo entre o pedido de exames e o retorno de seus resultados. Sistemas de suporte à decisão também se mostraram capazes de reduzir os custos de cuidados médicos. Muito da economia percebida resulta da prescrição de medicamentos menos caros e também da diminuição dos erros médicos e de eventos adversos. Muitos dos estudos mostrando o custo-benefício dos DSSs têm sido realizados em instituições nas quais os sistemas foram desenvolvidos internamente e passaram por um período significativo de refinamento. Um exemplo de estudo foi o desenvolvido em [9], no qual 52 dermatologistas voluntários avaliaram a malignidade de 25 imagens de lesões e deram uma recomendação sobre cada uma das imagens. Depois de receber a avaliação do DSS, os médicos puderam rever a sua opinião inicial. Desse experimento resultaram três resultados que devem ser considerados

na implantação de DSSs:

- Com base na recomendação de um DSS, os médicos estão dispostos a mudar a sua decisão em 24% dos casos.
- O número de vezes em que uma decisão é revertida se correlaciona negativamente com o nível de experiência dos médicos que utilizam o sistema.
- Os médicos estão mais dispostos a aceitar uma recomendação de um DSS quando eles não estão confiantes do seu diagnóstico.

Este trabalho também justifica-se do ponto de vista da quantidade de dados adquiridos eletronicamente dos pacientes, que tem crescido exponencialmente na última década. As bases de dados hospitalares, incluindo os sistemas demográficos, os registros eletrônicos de pacientes, bem como de entrada de pedidos, de laboratório, de farmácia e de radiologia crescem em termos de escopo e capacidade de armazenamento de dados a cada ano. Monitores de cabeceira de leitos modernos comunicam-se com uma série de dispositivos através de barramentos de dados e outras interfaces. Esses equipamentos de cabeceira guardam dados eletrônicos, ao passo que enormes bases de dados têm sido adquiridas por seguradoras e pelos governos contendo dados demográficos, procedurais e específicos de determinadas doenças. Em sua forma bruta, tais dados são relativamente pouco informativos. Todavia, se manuseados corretamente, esse dados podem ser minerados a fim de que se extraiam dos mesmos informações novas e inesperadas. A Ciência da Computação tem desenvolvido uma série de ferramentas para extrair informações de dados e melhorar a análise dos especialistas em análises clínicas. Em muitos casos, essas ferramentas são modeladas sobre processos fisiológicos, tais como a cognição humana (rede neural, raciocínio baseado em casos) ou algoritmos genéticos. Para justificar o uso da mineração de dados nessa dissertação, parte-se da definição adequada de Inteligência Artificial que é, por si só, controversa. Alan Turing, o matemático inglês, criou o que ficou conhecido como o Teste de Turing de inteligência computacional. Ele sugeriu que um computador tinha Inteligência Artificial se conseguisse imitar um ser humano e, assim, enganar outro ser humano. Um sistema especialista é um programa de computador que simula o julgamento e comportamento de um ser humano ou uma organização com conhecimentos técnicos e experiência em um campo particular. Por sua vez, a Mineração de Dados é a análise de dados para a extração de relacionamentos que não tenham sido previamente descobertos. As técnicas utilizadas para a Mineração de Dados podem descobrir associações ocultas ou seqüências de conjuntos de dados, o agrupamento de pontos de dados, e permitir a

visualização dessas relações entre os dados ou previsões com base em padrões escondidos. A Mineração de Dados é também conhecida como a descoberta de conhecimento, e deriva das áreas da Estatística, da Inteligência Artificial, e da Aprendizagem de Máquina. O conceito de sistemas médicos especialistas passou a ser comentado a partir do advento da Inteligência Artificial, nos anos de 1960 e 1970. No entanto, os médicos têm sido relutantes para integrar as ferramentas computadorizadas de análise de dados em sua prática habitual, em parte devido ao insucesso dos sistemas pioneiros desenvolvidos.

Há muitos fatores que retardaram a aceitação da aplicação de Inteligência Artificial em soluções de Medicina, incluindo a pequena margem de erro na tomada de decisão médica e da disponibilidade de especialistas em diversos ambientes. Estes obstáculos podem mudar no curto prazo, tanto quanto as novas soluções de informática melhorarem, as pressões por desempenho aumentem e se diminua o número de especialistas. Na medida em que as ferramentas computadorizadas consigam atuar como uma extensão ao ofício do clínico, as mesmas tornar-se-ão cada vez mais aceitáveis.

1.3.1 Dispositivos Móveis na Área Médica

Os dispositivos móveis estão moldando o ambiente de tratamento de saúde e impactando a forma como os médicos trabalham. Assim como os consumidores estão cada vez mais fazendo uso de dispositivos móveis para gerenciar *e-mail*, para o consumo de conteúdo e para o uso de aplicativos para simplificar as suas vidas, os médicos também estão incorporando *tablets* e aplicativos móveis em suas práticas. Sendo assim, lenta mas seguramente, os médicos estão se tornando mais experientes com essas tecnologias e com as mídias sociais, integrando o uso delas em suas práticas.

De acordo com o estudo *Taking the Pulse* realizado pela *Manhattan Research*, o uso de *tablets* por médicos quase dobrou no ano passado. O estudo anual se concentra em como os médicos usam a *Web* e outras formas de tecnologia no local de trabalho, fornecendo parâmetros estatísticos que são indicativos de tendências maiores. E, com mais médicos a incorporar a tecnologia para auxiliar no atendimento ao paciente e a integração de EMRs e EHRs tornam-se padrão, o uso de dispositivos móveis provavelmente vai continuar a crescer.

O uso de dispositivos móveis e das mídias sociais significa conveniência. Eles permitem que os médicos reduzam o tempo gasto no manuseio de papéis e levam à integração da tecnologia para o fluxo de trabalho, o que é bom para os profissionais e bom para os pacientes. Os *tablets* também permitem que os médicos rapidamente pesquisem sobre os

sintomas, mantenham-se atualizados sobre as notícias médicas, acessem as bases de dados de referência de medicamentos, busquem informações sobre os testes clínicos entre outras atividades.

Ou seja, a integração de mídias sociais e tecnologia no dia a dia muitas vezes possibilita aos médicos oferecer um melhor atendimento, mais personalizado.

Os aplicativos móveis também estão aparecendo bastante no mercado de consumo de cuidados de saúde, conferindo aos pacientes a possibilidade de tomarem cuidado da sua própria saúde. Existem desde aplicativos relacionados aos registros pessoais que permitem que os pacientes organizem seus próprios dados e acessem seus registros de saúde e resultados de laboratório, até aplicativos que facilitam o processamento de pagamentos e de elegibilidade de cobertura de seguro, e dispositivos que ajudam a monitorar o tratamento de doenças crônicas, e que aumentam seu bem-estar [10].

O estado da arte do mercado de dispositivos PDAs e móveis, atualizado em fevereiro desse ano, mostra que, alguns anos atrás, o pensamento principal era o de adquirir um *Palm* ou *PocketPC*. Com o surgimento de sistemas operacionais embarcados nos dispositivos, esse quadro mudou radicalmente.

Os PDAs da Palm, que por longo período dominaram o mercado, desapareceram. Quase todos os dispositivos agora são telefones móveis com *Internet* e outros inúmeros recursos. Os *iPhones*, fabricados pela *Apple* têm vendido muito bem desde a sua introdução, em junho de 2007 e são claramente o padrão pelo qual todos os outros *smartphones* são medidos. O *iPhone*, no momento, é a plataforma que possui o maior número de aplicativos médicos instaláveis no mercado. Os aparelhos providos de *Android*, a mais nova opção de sistema operacional desenvolvido pelo *Google*, começou a ganhar velocidade no final de 2009, início de 2010. O número de aplicações médicas disponíveis na plataforma *Android* está crescendo, apesar destes ainda estarem significativamente atrás dos dispositivos da *Apple*. Por sua vez, o *Windows Mobile* vêm ainda mais atrás em espaço no mercado de mobilidade. O *Blackberry*, que foi por longo tempo a plataforma líder nos *smartphones*, tem perdido uma fatia de mercado significativa para o *iPhone* e para o *Android*. Outro ponto importante é que as tecnologias de *Internet* banda larga e *Wi-Fi* são comuns na maioria dos dispositivos móveis. As operadoras de celular normalmente exigem a aquisição de algum tipo de plano de dados junto com o seu serviço de telefonia, que permitem o acesso com velocidades cada vez maiores fornecendo aos usuários o acesso a mais recursos da *Web* [11].

Basicamente, o *Android* é um sistema operacional móvel rodando em um kernel Linux.

Ele foi inicialmente desenvolvido pela *Android Inc.*, uma empresa mais tarde comprada pelo *Google*, e ultimamente pela *Open Handset Alliance*. Essa plataforma permite que os desenvolvedores escrevam código na linguagem *Java*, controlando o dispositivo via bibliotecas desenvolvidas pelo *Google*. O *Google* lançou a maior parte do código do *Android* sob a licença *Apache*, ou seja, este se trata de um software livre e com licença de código aberto. O sistema operacional *Android* possui muitas vantagens, como ser de plataforma aberta, o que possibilita que qualquer fabricante possa se juntar à *Open Handset Alliance*. Isso faz com que o número de desenvolvedores *Android* cresça significativamente. Sendo assim, qualquer programador pode desenvolver suas aplicações e colocá-lo à disposição do mercado gratuitamente através da loja virtual de aplicativos *Google Play*. A ampla utilização por diferentes fabricantes de *hardware* e as vantagens da computação em nuvem, visto que o *Google* tem uma grande quantidade de produtos e serviços que podem ser integrados ao *Android* também seduz os desenvolvedores. Resumindo, isso tudo faz com que o *Android* se torne um sistema cada vez mais popular e com tendências claras de crescimento.

Dadas essas informações, neste trabalho procura-se desenvolver uma solução que seja compatível com a plataforma *Android*, justamente por esta possuir um número menor de opções de aplicativos médicos aos seus usuários. Outro fator importante é a questão de que a comunidade de desenvolvedores para *Android* costuma ser bastante ativa na *Internet*, no sentido de facilitar a troca de conhecimentos em fóruns, blogs entre outros tipos de redes de interação. Ainda, diferentemente do *iOS*, sistema operacional da *Apple* e que é embarcado apenas nos *iPhones* dessa mesma empresa, o sistema *Android* está presente em uma série de dispositivos fabricados por diferentes companhias. Companhias estas, como a *HTC* e a *Samsung*, que costumam adicionar funcionalidades ao sistema. Outro ponto favorável ao desenvolvimento para a plataforma Android é o seu crescimento no mercado de *tablets*. A partir de 2011, *tablets Android* têm começado a ameaçar o domínio do *iPad*, *tablet* desenvolvido pela *Apple*. Como exemplos pode-se citar o *Samsung Galaxy Tab*, o *Motorola Xoom* ou ainda o *KindleFire*, produzido pela *Amazon* [12].

2 Fundamentação Teórica

2.1 Sistemas de Gestão de Dados Médicos e Hospitalares

2.1.1 Informação de Saúde

O Seguro de Portabilidade de Saúde e Definição de Lei de Responsabilidade (sigla em inglês, HIPAA), que é a legislação federal norte-americana e que inclui disposições para proteger as informações de saúde dos pacientes de divulgação não autorizada, define informações de saúde como qualquer informação, seja oral ou gravada em qualquer forma ou meio, que:

- é criada ou recebida por um prestador de cuidados de saúde, plano de saúde, autoridades de saúde pública, o empregador, seguradora de vida, escola ou universidade, ou câmara de compensação de cuidados de saúde
- refere-se à saúde passada, presente ou futura, física ou mental de um indivíduo, a prestação de cuidados de saúde a um indivíduo, ou o pagamento passado, presente ou futuro para a prestação de cuidados de saúde a um indivíduo.

A HIPAA refere-se a este tipo de informação, como informações de saúde protegidas. Para satisfazer tal definição, a informação deve em primeiro lugar ser identificável, ou seja, ela tem de ter um ponto de vista do paciente individual e a identidade do paciente deve ser conhecida. A HIPAA é certamente uma importante peça de legislação, e isso tem um impacto direto em como as organizações de cuidados de saúde devem criar e manter informações de saúde. No entanto, nem todas as informações que devem ser geridas numa organização de saúde estão protegidas. Grande parte da informação utilizada por profissionais de saúde e executivos não é identificável. É importante informar que as regulamentações sobre informações de pacientes também são regulamentadas em outros

países e blocos econômicos, como na União Européia, onde existem diretivas do parlamento europeu que protegem o processamento e movimentação de dados pessoais nos processos de cuidados de saúde, e no Canadá, onde existe o *Personal Information Protection and Electronic Documents Act* conhecido como PIPEDA, que estabelece regras para o uso, a divulgação e o armazenamento de dados pessoais [13].

2.1.2 Dados de Cuidados de Saúde

Os dados e informações de cuidados de saúde são divididas em duas categorias, a interna e a externa:

- Dados e informações internas
  - Encontro com o paciente
    - Específico do paciente
    - Agregado
    - Comparativo
  - Operações Gerais
- Dados e informações externas
  - Comparativo
  - Conhecimento de especialistas

Dentro da ampla categoria de dados e informações criadas internamente pela organização de saúde, concentra-se o estudo em informações clínicas e administrativas diretamente relacionadas com as atividades que cercam o encontro com o paciente, tanto encontro do indivíduo quanto do encontro coletivo. Por isso, as informações relacionadas ao encontro com o paciente foram divididas nas subcategorias agregada, específica e comparativa. Ou seja, o foco está na informação do atendimento clínico e administrativo individual e agregada de saúde que está associado a um encontro com o paciente. A tabela 1 lista os vários tipos de dados e informações que se enquadram nas subcategorias encontro com o paciente, específico e agregado. Informações normalmente encontradas em um registro médico do paciente são mostradas em itálico. (Os dados comparativos e a subcategoria informação são encontrados em ambas as categorias, interna e externa).

O segundo componente principal de informação interna de cuidados de saúde são as operações gerais. Todavia, os dados e as informações necessárias para operações gerais da

Tipo	Clínico	Administrativo
Específico do paciente	Folha de Identificação	Folha de Identificação
	Lista de problemas	Consentimentos
	Registro de medicação	Autorizações
	Histórico	Pré-autorização
	Exame físico	Agendamentos
	Notas de progresso	Admissão ou registro
	Consultas	Elegibilidade do seguro
	Ordens médicas	Pagamentos
	Resultados de raio-x	Códigos de diagnósticos
	Resultados laboratoriais	Códigos de procedimentos
	Registro de imunização	
	Relatório de operação	
	Relatório patológico	
	Resumo de alta	
	Código de diagnóstico	
	Código de procedimento	
Agregado	Índice de doenças	Relatório de custos
	Registros especializados	Análise de pedido de revisão
	Dados de resultados de exames	Análise de equipe
	Relatórios estatísticos	Análise de referências
	Análise de tendências	Relatórios estatísticos

Tabela 1: Exemplos dos tipos de informações e dados sobre pacientes

organização de cuidados de saúde não são o foco deste texto. Os executivos de cuidados de saúde que, no entanto, precisam se preocupar não apenas com a informação diretamente relacionada com o encontro com o paciente, mas também com informações sobre as operações gerais da organização. As organizações de saúde são, afinal, as empresas que devem ter faturamento superior aos custos para se manterem viáveis. As atividades administrativas padrão de qualquer organização viável também acontecem em ambientes de cuidados de saúde. Os executivos de saúde interagem com a informação e com os sistemas de informação em áreas como a contabilidade geral, o planejamento financeiro, a administração de pessoal e o planejamento das instalações de uma forma regular, se não diariamente. Nossa decisão de concentrar-se na informação que é exclusiva para os cuidados de saúde e não uma parte das operações de negócios em geral não tem a intenção de diminuir a importância das operações gerais, mas é um reconhecimento de que uma riqueza de recursos para informação geral de negócios e sistemas de informação já existe. Além de usar dados internamente gerados a partir do encontro com o paciente e os dados gerais de operações e informações, as organizações de cuidados de saúde utilizam informações geradas externamente. Os dados comparativos combinam dados internos e externos para auxiliar as organizações na avaliação do seu desempenho. A outra categoria

importante de informação externa utilizada em organizações de saúde é o conhecimento de especialistas, que geralmente é cobrado ou criado por peritos que não fazem parte da organização. Os prestadores de cuidados de saúde e executivos usam esse tipo de informação na tomada de decisão, tanto clínica como administrativa. Um exemplo clássico de conhecimento de informações clínicas é a informação contida em uma revista ou artigo profissional de saúde. Outros exemplos são as bases de dados regionais ou nacionais e *sites* informativos relacionados a questões de saúde ou de gestão.

A maioria das informações clínicas e específicas do paciente utilizadas em organizações de cuidados de saúde pode ser encontrada em ou tem origem de prontuários dos pacientes. A próxima seção irá apresentar alguns componentes básicos do registro médico do paciente. Ele também irá examinar uma internação e um encontro com o paciente ambulatorial, mostrando como o registro médico do paciente geralmente é criado. Todos os tipos de organizações de cuidados de saúde de internação, de ambulatório, de cuidados de longa duração, e assim por diante, tem prontuário. Esses registros podem ser em formato eletrônico ou em papel, mas o propósito e conteúdo básico são semelhantes independentemente do tipo de registro ou organização [14].

2.1.3 Objetivo dos Registros de Pacientes

As organizações de saúde mantêm registros médicos para vários fins. Esses propósitos permanecem constantes tanto se o registro é parte de um sistema eletrônico ou parte de um sistema baseado em papel e são os seguintes, extraídos de [14]:

- Atendimento ao paciente: os registros de pacientes fornecem a base documentada para o planejamento da assistência e do tratamento do paciente. Este propósito é considerado o motivo número um para se manter registros dos pacientes. Executivos de saúde precisam manter esse objetivo principalmente quando se examina os sistemas de informação em cuidados de saúde. Muitas vezes outros fins, particularmente de faturamento e de reembolso, podem parecer ter precedência sobre a assistência ao paciente.

- Comunicação: os registros de pacientes são um importante meio pelo qual os médicos, os enfermeiros, e outros podem se comunicar uns com os outros sobre as necessidades do paciente. Os membros da equipe de cuidados de saúde em geral, interagem com os pacientes em diferentes momentos durante o dia, a semana ou o mês. Além disso, o registro de paciente pode ser o único meio de comunicação entre

os diversos fornecedores.

- Documentação legal: os registros de pacientes, visto que descrevem e documentam os cuidados e o tratamento, também podem se tornar registros legais. Em caso de uma ação judicial ou outra ação legal envolvendo o atendimento ao paciente, o registro torna-se a principal evidência para o que realmente aconteceu durante o episódio de atendimento. Um velho ditado, mas absolutamente verdadeiro sobre a importância jurídica de registros de pacientes diz: "Se não foi documentado, não foi feito."

- Faturamento e reembolso: os registros de pacientes fornecem a documentação que os pacientes e contribuintes utilizam para verificar os serviços faturados. As companhias de seguros e outros pagadores terceiros insistem numa documentação clara para apoiar todas as reivindicações apresentadas. O governo federal realiza a supervisão e processos de revisão nos quais são utilizados registros de pacientes para confirmar a precisão das reclamações. A apresentação de um pedido de um serviço que não está claramente documentado no prontuário do paciente poderia ser interpretada como fraude.

- Pesquisa e gestão da qualidade: os registros de pacientes são utilizados em muitas instalações para fins de pesquisa e de monitoramento da qualidade dos cuidados prestados. Os registros de pacientes podem servir como documentos de origem a partir dos quais informações sobre determinadas doenças ou procedimentos podem ser tomadas, por exemplo. Embora a pesquisa seja mais prevalente em grandes centros médicos acadêmicos, estudos são realizados em outros tipos de organizações de saúde também.

A importância de manter registros dos pacientes completos e precisos não pode ser subestimada. Eles não servem apenas como base para o planejamento de assistência ao paciente, mas também como o registro legal, documentando o cuidado que foi fornecido aos pacientes por parte da organização. Os prontuários de pacientes fornecem grande parte dos dados de origem de informações de saúde que são gerados dentro e entre as organizações de cuidados de saúde. Os dados capturados em um registro médico do paciente pode se tornar um registro permanente de diagnósticos de paciente, de tratamentos e de resposta aos tratamentos.

2.1.4 Conteúdo dos Registros de Pacientes

Conforme encontrado em [14], a Associação de Gerenciamento de Informação em Saúde dos Estados Unidos, lista os seguintes componentes como sendo comuns aos registros de paciente, independentemente do tipo de instalação ou sistema de registro médico (eletrônico ou em papel). O conteúdo do histórico médico é determinado, em grande medida, por exigências externas, normas e regulamentações. Em suma, o registro do paciente é um repositório para uma variedade de dados clínicos e de informação que é produzida por vários indivíduos envolvidos no cuidado do paciente, sendo:

- Folha de identificação: as informações encontradas na folha de identificação (às vezes chamada de uma folha de rosto ou de admissão ou de registro de alta) se originam no momento da inscrição ou admissão. A ficha de identificação geralmente é o primeiro relatório ou tela que um usuário vai encontrar ao acessar um registro do paciente. Ele enumera pelo menos o nome do paciente, o endereço, o número de telefone, a operadora de seguros, e o número da apólice, bem como os diagnósticos do paciente e a disposição na alta. Estes diagnósticos são registrados pelos médicos e codificados pelo pessoal administrativo. A ficha de identificação é usada tanto como um documento clínico quanto como um documento administrativo. Ele oferece uma visualização rápida dos diagnósticos que necessitaram de cuidados durante o encontro. Os códigos e outras informações demográficas são utilizados para efeitos de reembolso e planejamento.

- Lista de problemas: os registros de pacientes freqüentemente contêm uma lista de problemas abrangente, que relacionam doenças significativas e operações que o paciente tenha experimentado. Esta lista é geralmente mantida ao longo do tempo. Ele não é específico para um único episódio de cuidados e pode ser mantido pelo assistente ou pelo médico de cuidados primários ou coletivamente por todos os prestadores de cuidados de saúde envolvidos no cuidado do paciente.

- Registro de medicação: às vezes chamada de registro de administração de medicamentos, este registro lista os medicamentos prescritos e posteriormente administrados ao paciente. Muitas vezes também lista todas as alergias à medicação que o paciente pode ter. Os profissionais de enfermagem são geralmente responsáveis pela documentação e pela manutenção de informações de medicação. No ambiente hospitalar, os enfermeiros são responsáveis pela administração de medicamentos de acordo com as ordens escritas ou verbais dos médicos.

- Histórico e exame físico: o componente histórico deste relatório descreve quaisquer doenças graves e cirurgias que o paciente teve, qualquer histórico familiar significativo de doenças, os hábitos de saúde do paciente e os medicamentos atuais. A informação para o histórico é fornecida pelo paciente (ou alguém em nome dele ou dela) e é documentada pelo médico assistente, no início ou imediatamente antes de um episódio de encontro ou de tratamento. O exame físico neste relatório afirma o que o médico encontrou quando ele ou ela realizou um exame no paciente. O histórico e o exame físico juntos documentam a avaliação inicial do paciente e proporcionam a base para o diagnóstico e o tratamento subsequente. Eles também fornecem um quadro no qual os médicos e outros prestadores de cuidados podem documentar descobertas significativas. Embora a obtenção do histórico inicial e do exame físico sejam uma atividade realizada uma só vez durante um episódio de cuidados de saúde, a reavaliação contínua e a documentação da reavaliação durante o curso de tratamento do paciente é crítica. Resultados de reavaliações são geralmente registrados em notas de progresso.

- Notas de progresso: as notas de progresso são feitas pelos médicos, enfermeiros, terapeutas, assistentes sociais e outros funcionários do ambiente clínico. As notas de progresso devem refletir a resposta do paciente ao tratamento, juntamente com as observações do provedor e planos para a continuação do tratamento. Há muitos formatos de notas de progresso. Em algumas organizações todos os prestadores de cuidados usam um mesmo formato, em outros cada tipo de provedor usa um formato personalizado.

- Consulta: uma nota de consulta ou opiniões registros do relatório sobre o estado do paciente é feita por um prestador de cuidados de saúde que não o médico ou o prestador de cuidados primários. Os relatórios de consulta podem vir de médicos e outros, de dentro ou fora de uma organização particular de saúde, mas as cópias são mantidas como parte do registro do paciente.

- Ordens do médico: as ordens do médico são as direções, as instruções ou as prescrições de um médico dadas a outros membros da equipe de saúde sobre os medicamentos do paciente, os testes, as dietas, os tratamentos e assim por diante. No atual sistema de saúde dos Estados Unidos, os procedimentos e os tratamentos devem ser solicitados pelo médico licenciado apropriado.

- Relatórios de imagem e raio-X: o radiologista é o responsável por interpretar as imagens produzidas através de raios-X, ultra-sonografias, mamografias, exames e afins,

para documentar suas interpretações ou conclusões no prontuário médico do paciente. Estes resultados devem ser documentados em tempo hábil para que estejam disponíveis para o médico apropriado a fim de facilitar o tratamento adequado. Os filmes ou imagens reais são geralmente mantidos nos departamentos de radiologia ou de imagem como cópias impressas ou em um sistema de computador especializado. Essas imagens não são normalmente consideradas parte do registro médico do paciente, mas, como outros relatórios, elas são armazenadas de acordo com as leis estaduais e diretrizes de prática clínica e são documentos importantes do cuidado do paciente.

- Relatórios de laboratório: os relatórios de laboratório contêm os resultados de testes realizados em fluidos do corpo, nas células e nos tecidos. Por exemplo, um laboratório médico pode realizar uma cultura de garganta, de exame de urina, de colesterol ou hemograma completo. Há centenas de testes de laboratório específicos que podem ser executados por organizações de saúde ou laboratórios especializados. Os profissionais de laboratório são responsáveis por documentar os resultados laboratoriais. Os médicos são responsáveis por documentar quaisquer descobertas e planos de tratamento com base nos resultados de laboratório.

- Formulários de autorização e consentimento: as cópias de autorizações para a internação, o tratamento, a cirurgia e a liberação de informação são um importante componente do registro médico e relacionado a seu uso como documento legal. O praticante que realmente oferece o tratamento tem de obter o consentimento para o mesmo. Os pacientes devem assinar documentos de consentimento antes do tratamento ocorrer. As formas de liberação de autorização de informação também devem ser assinados pelos pacientes antes de quaisquer informações específicas do paciente de cuidados de saúde sejam liberados para as partes não diretamente envolvidas no cuidado do paciente.

- Relatório operacional: os relatórios operacionais descrevem qualquer cirurgia realizada e listam os nomes dos cirurgiões e dos assistentes. O cirurgião é responsável pelo relatório operacional.

- Relatório de patologia: os relatórios de patologia descrevem os tecidos removidos durante todo o procedimento cirúrgico e o diagnóstico baseado no exame desse tecido. O patologista é responsável pelo relatório de patologia.

- Sumário de alta: cada prontuário médico hospitalar contém um resumo de alta. O

sumário de alta resume a internação hospitalar, incluindo o motivo da internação, os resultados significativos a partir de testes, os procedimentos realizados, as terapias fornecidas, as respostas aos tratamentos, a condição no momento da alta, e as instruções para os medicamentos, a atividade, a dieta e os cuidados de acompanhamento. O médico assistente é responsável por documentar o resumo de alta no final da estadia do paciente no hospital.

Um exemplo de registro médico eletrônico pode ser visto na figura 2.

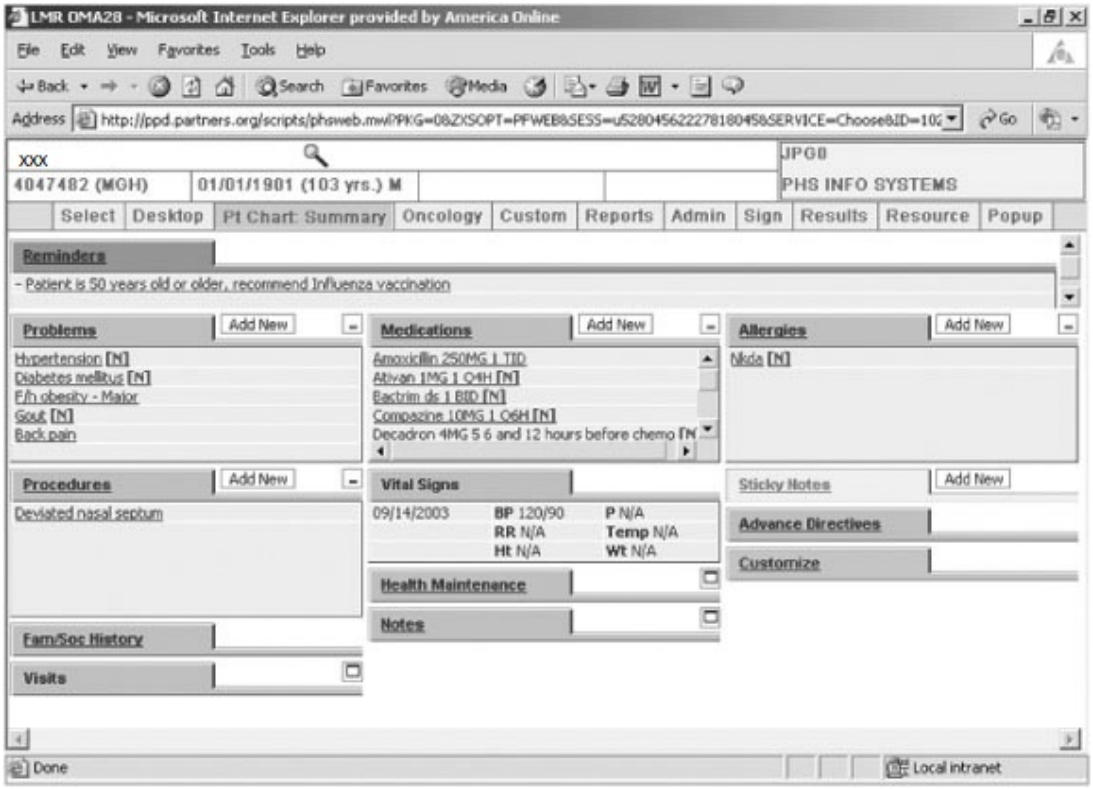

Figura 2: Registro médico eletrônico [14]

2.1.5 Informação versus Dado

Sobre a diferença entre dados e informações, existe uma resposta simples de que a informação são os dados processados. Portanto, podemos dizer que a informação de saúde são os dados de saúde processados. Os dados de saúde são fatos, geralmente armazenadas como caracteres, palavras, símbolos, medições ou estatísticas. Uma coisa aparente sobre os dados de saúde é que eles geralmente não são muito úteis para a tomada de decisão. Os dados de saúde podem descrever um evento particular, mas sozinhos e não processados não são particularmente úteis. Tome por exemplo, este número: 79%. Por si só, isso

pouco significa, porém se processado, descobre-se que este representa a ocupação média dos leitos para um hospital para o mês de janeiro. Sendo assim, esse dado assume um maior significado. Mas ainda assim, isso já pode ser considerado uma informação. Isso depende. Se tudo que um executivo de cuidados de saúde quer ou precisa saber é a taxa de ocupação dos leitos em janeiro, esse percentual poderia ser considerado uma informação. No entanto, para o executivo do hospital que está interessado em conhecer a evolução da taxa de ocupação dos leitos ao longo do tempo ou a taxa da facilidade de ocupação dos leitos comparada com a de outras instalações semelhantes, esta ainda não é uma informação. O conhecimento é visto por alguns como o mais alto nível em uma hierarquia com os dados de fundo e as informações no meio. Isso é representado pela figura 3 O conhecimento é definido por Johns, em 1997, como "uma combinação de regras, relações, idéias e experiência."Outra maneira de pensar o conhecimento é este sendo uma informação aplicada a regras, experiências e relacionamentos, com o resultado que ele pode ser usado para a tomada de decisão. Um artigo de jornal que descreve o uso de taxas de ocupação de leitos na tomada de decisão ou a experiência de uma unidade de saúde, com melhoria das suas taxas de ocupação poderia ser um exemplo de conhecimento.

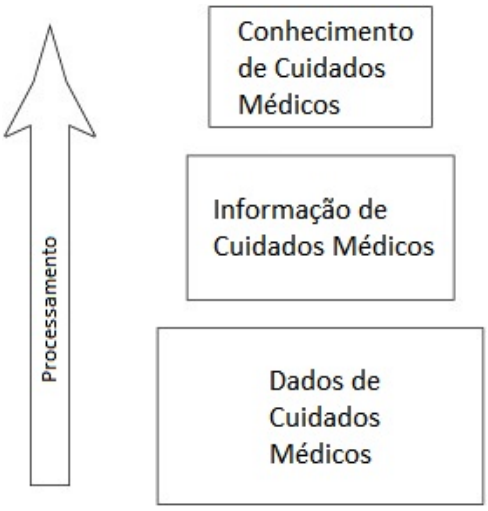

Figura 3: Hierarquia dos dados até a informação. [14]

Portanto, a informação é um bem extremamente valioso em todos os níveis de uma organização de saúde. Os executivos de saúde, os clínicos, e outros dependem de informações para cumprir seus atingimentos nos seus postos de trabalho. Um ponto interessante a se pensar é que os mesmos dados podem fornecer informações diferentes para diferentes usuários. Ou seja, o mais importante não é definir onde termina um dado e começa uma informação, mas sim promover a compreensão da relação entre os dados de saúde e as informações de cuidados de saúde, visto que não se pode criar informações sem dados [14].

2.2 Sistemas de Suporte à Decisão

Os sistemas de suporte à decisão mais recentes possuem um escopo mais objetivo e, provavelmente, chegarão ao uso generalizado assim como outros componentes de *softwares* médicos. É importante salientar que os esses DSSs possuem quatro funções principais.

- A primeira é a função administrativa, na qual os DSSs suportam codificação de dados clínicos e de documentação, bem como a gestão das consultas dos pacientes e de autorização para procedimentos médicos. Um médico de pronto atendimento pode usar uma ferramenta de suporte automatizado de decisão, por exemplo, para determinar se um paciente está numa situação na qual precisará ser encaminhado a um médico e de qual a especialidade.

- A segunda categoria trata sobre o papel dos DSSs na ajuda e controle dos aspectos complexos e variados da assistência médica, tais como o agendamento automático de visitas de acompanhamento, o acompanhamento de pedidos até a sua conclusão e o relatório de saída, gerando automaticamente lembretes em relação ao cuidado preventivo ou controle de adesão a protocolos de pesquisa.

- Uma terceira categoria e que é a mais comumente reconhecida no meio médico é o uso dos DSSs no controle e prevenção de custos pela administração da farmácia, do laboratório, além do controle para evitar a expedição de ordens para testes desnecessários aos pacientes.

- Finalmente, o quarto papel dos DSSs se dá na promoção de boas práticas, de orientações sobre condições específicas (para pacientes asmáticos, por exemplo) e como ferramenta de gestão baseada nas características da população (por exemplo, os negros com hipertensão).

Independentemente do tipo, os DSSs devem começar a partir de uma base de conhecimento, usar algum tipo de motor e produzir ou sugerir recomendações ou intervenções. As bases de conhecimento podem consistir de observações automatizadas sobre os sinais vitais de um paciente ou observações introduzidas manualmente a partir de uma análise de um clínico. Alternativamente, um sistema pode ser preparado a partir do uso de conhecimento de origem acadêmica derivados normalmente de livros ou revistas médicas. Qualquer uma destas bases de conhecimento pode ser melhorada com o conhecimento experiencial proveniente do exercício rotineiro da Medicina. O motor, referido anteriormente, é o *software* de base e a metodologia de análise dos dados. Esse motor pode ser

encarado como uma caixa preta, do ponto de vista do médico, embora seja importante compreender algumas das abordagens comuns, principalmente aquelas que irão tornar-se cada vez mais utilizadas no futuro. Finalmente, os tipos de intervenções que um DSS pode fazer são amplamente classificadas em sistemas passivos, semi-ativos e ativos. A tomada de decisão médica exige o uso de diferentes tipos de conhecimento. Um médico pode, por exemplo, usar informações anatômicas, patológicas, epidemiológicas, taxonômicas, farmacológicas e terapêuticas para chegar a um diagnóstico ou decisão o sobre o tratamento. Um DSS deve, portanto, ser capaz de se utilizar destes mesmos conjuntos de dados. O conhecimento empírico pode ser adquirido por um DSS em uma de duas abordagens gerais. A primeira é a chamada abordagem *top-down*, na qual um especialista ensina para o sistema como ele pensa. Os programas INTERNIST-1 e MYCIN, já referidos, foram desenhados com este modelo. A outra alternativa, a *bottom-up*, é a abordagem cada vez mais prevalente e que tem se tornado possível graças às novas ferramentas de *software* e bases de dados de grandes dimensões atualmente existentes. Neste caso, o conhecimento é obtido automaticamente por um DSS através da análise de um conjunto de dados. Por exemplo, um DSS pode aprender que as frequências cardíacas superiores a certo número estão associadas com a depressão, em um paciente em particular ou numa população de doentes, ou que a hospitalização numa unidade de cuidados intensivos determinada está associada a uma alta taxa de infecção hospitalar. Sistemas *top-down* utilizam-se de regras, normalmente derivadas de especialistas. Já os sistemas *bottom-up* usam ferramentas como redes neurais ou de aprendizagem de máquina nas quais o software inteligente pode encontrar informações novas ou inesperadas por analisar grandes conjuntos de dados para as associações. Os sistemas *top-down* normalmente requerem manutenção contínua e supervisão, enquanto que os *bottom-up* podem ser de auto-aprendizagem.

2.2.1 Lógica de Suporte de Decisão

Os chamados motores de decisão de um sistema de apoio usam combinações diferentes de raciocínio, de explicações e de aprendizagem. Sendo assim, os modelos matemáticos podem ser utilizados para descrever a interação entre a dose e efeito, tal como a utilização de modelos farmacocinéticos relacionados com a administração de certas drogas. Por exemplo, um modelo farmacocinético pode ser usado por um DSS para recomendar a dose apropriada e o intervalo de dosagem para a administração de gentamicina em um paciente com um peso conhecido com uma depuração de creatinina específica. Os métodos estatísticos são tipicamente indutivos, e com base na relação de antecedentes conhecidos (isto é, sinais ou sintomas) com um resultado (isto é, um diagnóstico). Ao

sistema são passados os valores dos antecedentes para um determinado paciente (isto é, rigidez do pescoço = VERDADEIRO, fotofobia = VERDADEIRO) e este produz um resultado (isto é, a probabilidade de meningite) para esse paciente. Esta abordagem pode ser aplicada para o prognóstico, o diagnóstico, ou a abordagem terapêutica. As Redes bayesianas ou redes de crença se enquadram nessa categoria, e contam com a lógica Bayesiana no seu funcionamento. Quanto aos processos de pensamento de especialistas, estes são geralmente representados por uma série de regras destinadas a expressar a maneira que o perito analisa um problema, e pode, portanto, explicar o seu pensamento. Esta capacidade explicativa é atraente porque os médicos estão compreensivelmente cautelosos com as soluções quando eles não entendem a sua derivação. A grande desvantagem para os sistemas especialistas é o grau em que eles precisam ser mantidos com novos conhecimentos que emergem cada vez mais rápido. Além disso, tornou-se claro que os verdadeiros especialistas chegam a conclusões que utilizam um processo associativo ou intuitivo, que não é facilmente traduzido em conjuntos de regras eficazes.

2.2.2 Modos de Suporte de Decisão

Um DSS pode atuar em um de três modos gerais:

- Primeiramente, os sistemas passivos, aos quais são fornecidos dados de entrada com um pedido de uma resposta (diagnóstico). O MYCIN e INTERNIST-1 foram sistemas que funcionaram dessa maneira. Os DSSs passivos podem agir como sistemas de consultoria ou como sistemas críticos em relação à análise de um clínico.

- Os sistemas semi-ativos baseiam-se no conhecimento adquirido para produzir lembretes com recomendações de ações a serem realizadas. Um sistema semi-ativo pode lembrar que uma vacina é necessária, listar as contra indicações referentes à prescrição de determinados medicamentos ou enumerar boas práticas de cuidados de saúde. Os sistemas semi-ativos também podem ser usados para monitorar variáveis fisiológicas, tais como a freqüência cardíaca e a pressão arterial, alertando os médicos sobre o atingimento de limites perigosos para a saúde.

- Finalmente, um DSS ativo é aquele que pode intervir automaticamente. Um DSS ativo pode encomendar terapias ou investigações de forma autônoma. Sistemas ativos podem também ser usados para controlar a titulação de um medicamento (tal como um agente anti-hipertensivo), baseando-se em um circuito de controle de realimentação, ou ainda, administrar o desmame da ventilação mecânica. Alguns

DSSs mais complexos, já foram integrados a desfibriladores implantados em pacientes para controlar, ativamente, o tempo e a dose de corrente elétrica a ser aplicada nos casos de parada cardíaca [2].

2.3 Descoberta de Conhecimento em Bases de Dados

A área de Descoberta de Conhecimento em Bases de Dados, do inglês *Knowledge Discovery in Databases*, ou *KDD*, é relacionada com o desenvolvimento de métodos e técnicas para conferir sentido aos dados armazenados em bases de dados. Basicamente, o processo de KDD realiza um mapeamento de dados de baixo nível, que são tipicamente muito volumosos para se entender e digerir rapidamente, em outras formas mais compactas (por exemplo, um relatório curto), mais abstratas (por exemplo, uma aproximação descritiva) ou mais útil (por exemplo, um modelo preditivo para estimar o valor de casos futuros). No cerne do processo fica o componente responsável pela aplicação de determinados métodos de Mineração de Dados para a descoberta de padrões e posterior extração destes. O método tradicional de transformação de dados em conhecimento baseia-se na análise e interpretação manuais. Por exemplo, nas áreas da saúde e indústria, é comum que os especialistas periodicamente analisem as atuais tendências e mudanças em dados de cuidados de saúde, por exemplo, em uma base trimestral. Os especialistas, então, fornecem um relatório detalhando a análise para a organização de saúde e este relatório torna-se a base para as futuras tomadas de decisão e planejamento da gestão médica. Para estas aplicações, e muitas outras, esta forma de sondagem manual de um conjunto de dados é lenta, cara e altamente subjetiva. Adicionalmente, os volumes de dados crescem dramaticamente, o que torna este tipo de análise de dados manual completamente inviável em muitos domínios. Cabe salientar que as bases de dados estão aumentando suas dimensões de duas maneiras, pelo número de registros ou objetos na base de dados e pelo número de campos ou atributos que compões um objeto. Historicamente, a atividade de encontrar padrões úteis em dados tem sido chamada por uma variedade de nomes, incluindo extração de dados de conhecimento, Mineração de Dados, descoberta de informações, arqueologia de dados e processamento de dados. O termo Mineração de Dados é o que mais ganhou popularidade e por isso é interessante frisar que o KDD refere-se ao processo global de descoberta de conhecimento útil a partir de dados, e a Mineração de Dados refere-se a uma determinada etapa neste processo. A Mineração de Dados é, por sua vez, a aplicação de algoritmos específicos para extrair padrões de dados.

Existem no processo de KDD etapas adicionais tais como a preparação dos dados, a

seleção, a limpeza, a incorporação do conhecimento prévio e uma interpretação correta dos resultados da mineração. Todos esses passos são essenciais para assegurar que conhecimento útil seja derivado a partir dos dados. As etapas do KDD são mais detalhadas a seguir e na figura 4 pode-se visualizar as mesmas num nível mais amplo:

- 1ª etapa: compreensão do domínio da aplicação e do conhecimento anterior relevante, identificando o objetivo do processo de KDD do ponto de vista do cliente.
- 2ª etapa: criação de um conjunto de dados alvo, selecionando um conjunto de dados, ou concentrando-se em um subconjunto de variáveis ou amostras de dados, na qual a descoberta de padrões será realizada.
- 3ª etapa: a limpeza de dados e pré-processamento, que incluem operações básicas como a remoção de ruído, a coleta da informação necessária, decidindo sobre estratégias para lidar com campos faltantes de dados entre outras.
- 4ª etapa: a redução de dados e projeção, a fim de encontrar características úteis para representar os dados dependendo do objetivo da tarefa. Com a redução da dimensionalidade ou transformação, o número efetivo de variáveis sob consideração pode ser reduzida.
- 5ª etapa: combinação dos objetivos do processo de KDD para a extração de dados. Por exemplo, a classificação, a regressão, o agrupamento, que serão mais adiante aprofundados na seção sobre Mineração de Dados.
- 6ª etapa: análise exploratória para a escolha do algoritmo de Mineração de Dados a ser utilizado para a busca de padrões de dados.
- 7ª etapa: Mineração de Dados, ou seja, a busca de padrões de interesse através de regras de classificação, de árvores de regressão, agrupamento entre outras técnicas.
- 8ª etapa: interpretação dos padrões minerados, envolvendo a visualização dos padrões extraídos ou dos modelos gerados.
- 9ª etapa: Ações sobre o conhecimento descoberto através do uso desse conhecimento diretamente, ou incorporando o mesmo em outro sistema, ou simplesmente documentando-o para a informação das partes interessadas [15].

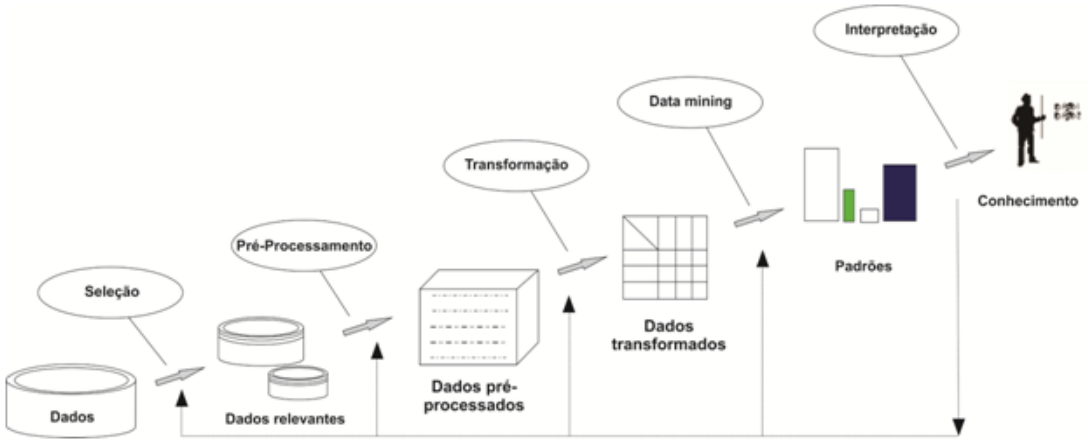

Figura 4: Etapas do processo de KDD [15]

2.4 Mineração de Dados

Com a velocidade na qual percebemos a inserção de dispositivos e serviços informatizados no nosso dia a dia, estamos ficando sobrecarregados com uma quantidade incalculável de dados. Os onipresentes computadores, celulares e *tablets* de uso pessoal com capacidades significativas de armazenamento, tornaram muito fácil aos usuários guardar coisas que anteriormente jogaríamos na lixeira para deleção. Memórias de *Gigabytes* e até mesmo *Terabytes* estão se tornando componentes baratos e com velocidade de resposta rápida, o que nos leva a, sempre que possível, adiar a decisão sobre o que fazer com o material que, no momento, não vemos mais utilidade. Existem ainda outros dispositivos eletrônicos que passando imperceptíveis aos nossos olhos estão todo o tempo gravando nossas decisões, nossas escolhas no supermercado, nossos hábitos financeiros, o nosso ir e vir. Adicionalmente, a rede mundial de computadores nos cerca com uma gigantesca quantidade de informações, e enquanto isso, cada escolha dos usuários é gravada. E expandindo tal situação, além desses dados pessoais, existe toda uma gama de dados sobre as atividades de comércio e da indústria que também estão sendo armazenadas. Sendo assim, percebemos que está se criando um abismo entre a velocidade de geração de dados e a nossa velocidade de entendimento sobre esses mesmos dados. À medida que o volume de dados aumenta, inexoravelmente, a proporção de pessoas que os compreendem diminui de forma alarmante. Ou seja, existem escondidos nesses dados, informações que são potencialmente úteis e que raramente são explicitadas para que se obtenham vantagens sobre. Na Mineração de Dados, os dados são armazenados eletronicamente e a sua posterior pesquisa é automatizada, ou, pelo menos, executada com o auxílio de um programa de computador. O conceito de Mineração de Dados não é novo. Os economis-

tas, os estatísticos, os meteorologistas e os Engenheiros de Comunicação há muito tempo trabalham com a idéia de que padrões nos dados podem ser procurados automaticamente, identificados, validados e, finalmente, utilizados para predições. A novidade é o aumento vertiginoso de oportunidades para se encontrar padrões em dados. O crescimento desenfreado das bases de dados nos últimos anos além dos bancos de dados sobre as atividades cotidianas como as escolhas dos clientes, colocam a Mineração de Dados numa posição de destaque entre as tecnologias a serem empregadas na implantação de novos negócios. Isso porque se inteligentemente analisados, essas bases de dados tornam-se recursos valiosos. Ou seja, essa análise pode levar a novas descobertas e, em ambientes comerciais, a vantagens competitivas. Sendo assim, pode-se definir a Mineração de Dados como sendo a solução de problemas através da análise de dados já presentes em bases de dados. Ou ainda, pode ser definida como o processo de descoberta de padrões de dados, sendo esse processo automático ou semi-automático. Os padrões descobertos devem ser significativos na medida em que eles direcionam os profissionais a obter alguma vantagem, geralmente econômica. Padrões úteis nos permitem fazer previsões não triviais sobre os novos dados. Há dois extremos para a expressão de um padrão: como uma caixa preta, cuja estrutura interna e incompreensível e como uma caixa transparente, cuja construção revela a estrutura do padrão. Ambos têm a capacidade de realizar boas previsões. A diferença é se esses padrões minerados serão representados em termos de uma estrutura a qual poderá ser examinada e usada para tomadas de decisão futuras. Tais padrões são denominados estruturais, visto que os mesmos tornam a estrutura de decisão explícita. Em outras palavras, estes ajudam a explicar algo sobre os dados. A técnica de Mineração de Dados pode ser aplicada em diversas áreas, destacando decisões que envolvem o julgamento, a triagem de imagens, a previsão de carga, as decisões de *marketing*, a predição de diagnósticos e de vendas. A entrada de dados para a execução da Mineração de Dados tem a forma de conceitos, instâncias e atributos. Aquilo que se pretende aprender, a partir da aplicação da mineração, é o que se chama de conceito. Esse conceito nada mais é do que o resultado que se quer encontrar através do processo de aprendizagem, de maneira que esse resultado seja compreensível e aplicado para casos reais. A informação que é passada para a aprendizagem tem a forma de um conjunto de instâncias, sendo que cada instância é um exemplo independente e individual do conceito a ser aprendido. Cada exemplo é caracterizado por valores de atributos que medem diferentes aspectos da instância. Existem muitos tipos diferentes de atributos, embora os métodos mais usados de Mineração de Dados lidem apenas com os tipos numéricos e nominais, ou ainda categóricos. Há muitas maneiras diferentes para representar os padrões que podem ser descobertos pela aprendi-

zagem de máquina e cada um determina o tipo de técnica que pode ser usada para inferir a estrutura de dados de saída.

2.4.1 Tabelas de Decisão

O modo mais rudimentar e direto de representar a saída do processo de Mineração de Dados é fazê-la na forma de uma tabela de decisão, tal como é formatada a entrada de dados do processo.

2.4.2 Árvores de Decisão

A abordagem "dividir para conquistar" de aprendizado a partir de um conjunto de casos independentes leva naturalmente a um estilo de representação chamado de árvore de decisão. Os nós em uma árvore de decisão envolvem o teste de um determinado atributo. Normalmente, o teste num nó compara um atributo com o valor de uma constante. No entanto, algumas árvores comparam dois atributos entre si ou usam alguma função de um ou mais atributos. Já os nodos folhas são responsáveis por conferir uma classificação que se aplica a todos os casos que atingem esta folha, ou um conjunto de classificações, ou a distribuição de probabilidade em todas as classificações possíveis. Para classificar um exemplo desconhecido, este percorre a árvore de acordo com os valores dos atributos testados em nós sucessivos, e, quando uma folha é atingida, a instância é classificada de acordo com a classe atribuída à folha.

2.4.3 Regras de Classificação

O antecedente, ou pré-condição, de uma regra é uma série de testes como os testes em nós em árvores de decisão, e o conseqüente, ou conclusão, dá a classe ou as classes que se aplicam aos casos cobertos por essa regra, ou ainda gera uma distribuição de probabilidade sobre as classes. Geralmente, as pré-condições são logicamente combinadas por operações *AND*, e todos os testes devem passar com êxito se a regra é correta. É fácil ler um conjunto de regras diretamente de uma árvore de decisão. Uma regra é gerada para cada folha. O antecedente da regra inclui uma condição para cada nó no caminho da raiz para aquela folha, e o conseqüente da regra é a classe designada pela folha. No entanto, em geral, as regras que são lidas diretamente de uma árvore de decisão são muito mais complexas do que o necessário, e as regras derivadas de árvores geralmente são podadas para remover testes redundantes.

2.4.4 Regras de Associação

As regras de associação diferem-se das regras de classificação no aspecto de que as regras de associação podem prever qualquer atributo, e não apenas a classe, e isso lhes dá a liberdade de predizer combinações ente os atributos também. Além disso, as regras de associação não se destinam para serem usadas como um conjunto, como as regras de classificação são. Diferentes regras de associação expressam diferentes regularidades que sustentam o conjunto de dados, e elas geralmente preveem coisas diferentes. Devido ao fato de que muitas regras de associação diferentes podem ser derivadas a partir uma pequena base de dados, o interesse é maior sobre as regras que se aplicam a um número razoavelmente grande de instâncias e que têm uma precisão relativamente elevada nos casos nos quais elas se aplicam.

2.4.5 Agrupamento

Quando o agrupamento é utilizado, ao invés de um classificador, a saída tem a forma de um diagrama que mostra como as instâncias podem ser agrupadas em grupos, *clusters*. Em outras palavras, essa operação de agrupamento envolve a associação de um número de grupo para cada caso, o que pode ser representado pela imposição das instâncias em duas dimensões e pelo particionamento do espaço para mostrar cada grupo. O agrupamento é geralmente seguido por uma fase na qual uma árvore de decisão ou um conjunto de regras é inferido, o qual aloca cada instância a um grupo ao qual deve pertencer. Em seguida, a operação de agrupamento é apenas um passo no caminho para se obter uma descrição estrutural [16].

2.5 Algoritmos de Classificação

Os algoritmos que foram avaliados neste trabalho pertencem a dois tipos principais de modelos de classificação, os baseados em árvores de decisão e os baseados em classificadores Bayesianos.

Uma árvore de decisão é uma estrutura de árvore tipo fluxograma ou modelo de decisões, no qual cada nó interno denota um teste de um atributo, cada ramo representa um resultado do teste que leva a um nó folha, representando as classes ou distribuições de classe. O nó mais alto em uma árvore é o nó raiz. As árvores de decisão são construídas em uma forma *top-down* recursiva, utilizando-se da abordagem “dividir para conquistar”.

Começando com um conjunto de treinamento de tuplas e seus rótulos de classe associados, o conjunto de treinamento é recursivamente particionado em subconjuntos menores conforme a árvore está sendo construída. No entanto, nem todos os ramos são vistos em uma árvore de decisão. A técnica chamada de poda da árvore tenta identificar e remover galhos que podem refletir ruídos ou incorreções, com o objetivo de melhorar a precisão da classificação[17].

2.5.1 Árvore de Modelo Logístico

Uma árvore de modelo logístico consiste basicamente de uma estrutura padrão de árvore de decisão com funções de regressão logística nas folhas, bem como um modelo de uma árvore de regressão é uma árvore com funções de regressão nas folhas. Assim como em árvores de decisões comuns, um teste num dos atributos é associado com cada nó interior. Para um atributo nominal enumerado com k valores, o nó tem k nós filhos, e as instâncias são ordenadas decrescentemente em cada um dos ramos k, dependendo do seu valor do atributo. Para atributos numéricos, o nó tem dois nós filhos e o teste consiste na comparação do valor do atributo com um valor limiar: uma instância é classificada para o ramo esquerdo se o seu valor de atributo que é menor do que o do limiar ou classificado para o ramo direito caso contrário [18].

2.5.2 Árvores Funcionais

Dado um conjunto de exemplos e um construtor de atributo, o algoritmo geral usado para construir uma árvore funcional é apresentado na figura 5.

Este algoritmo é semelhante a muitos outros, exceto na fase construtiva (etapas 2 e 3). Aqui, uma função é construída e mapeada para novos atributos. Há alguns aspectos deste algoritmo que devem ser explicitados. No passo 2, um modelo é construído usando a função de construtor. Isso é feito utilizando apenas os exemplos que se enquadram neste nó. Depois, no passo 3, o modelo é mapeado para novos atributos. A função de construtor deve ser um classificador ou um regressor, dependendo do tipo do problema. No primeiro, o número de novos atributos é igual ao número de classes, no último a função construtor é mapeada para um novo atributo. No passo 3, cada novo atributo é calculado como o valor previsto pela função construída para cada exemplo. Na configuração de classificação, cada novo atributo-valor é a probabilidade de que o exemplo pertence a uma determinada classe do modelo construído. O mérito de cada novo atributo é avaliado utilizando a função de

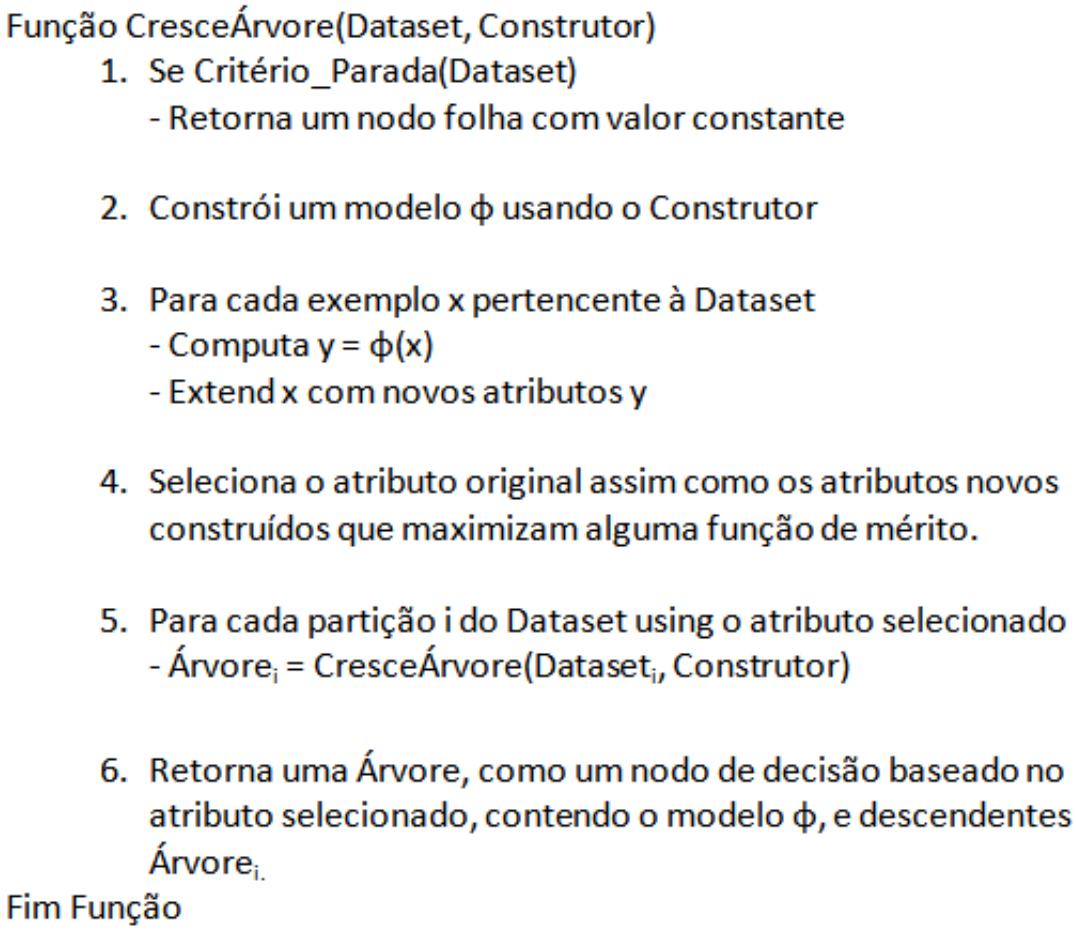

Figura 5: Construção de uma Árvore Funcional [19]

mérito da árvore univariada, e em concorrência com os atributos originais (passo 4). O modelo construído pelo algoritmo tem dois tipos de nós de decisão: aqueles com base em um teste de um dos atributos originais, e aqueles que se baseiam nos valores da função de construtor. Ao utilizar modelos lineares generalizados (GLM), como o construtor de atributo, cada novo atributo é uma combinação linear dos atributos originais. Os nós de decisão com base em atributos construídos definem uma superfície multivariada de decisão.

Uma vez que uma árvore foi construída, ele é podada. O algoritmo geral para podar a árvore é apresentado na Figura 6. A árvore é percorrida no sentido *bottom-up*. Para cada nó não-folha duas quantidades são estimadas: o erro estático eo erro de *backup*. O erro estático é uma estimativa do erro, como se fosse o nó fosse uma folha. Já o erro de *backup* é a soma ponderada da estimativa dos erros de todas as subárvores do nó atual. A estimativa do erro de cada ramo é ponderada pela probabilidade de que um exemplo segue o ramo. Se o erro de *backup* é maior ou igual do que o erro estático, então o nó é substituído por

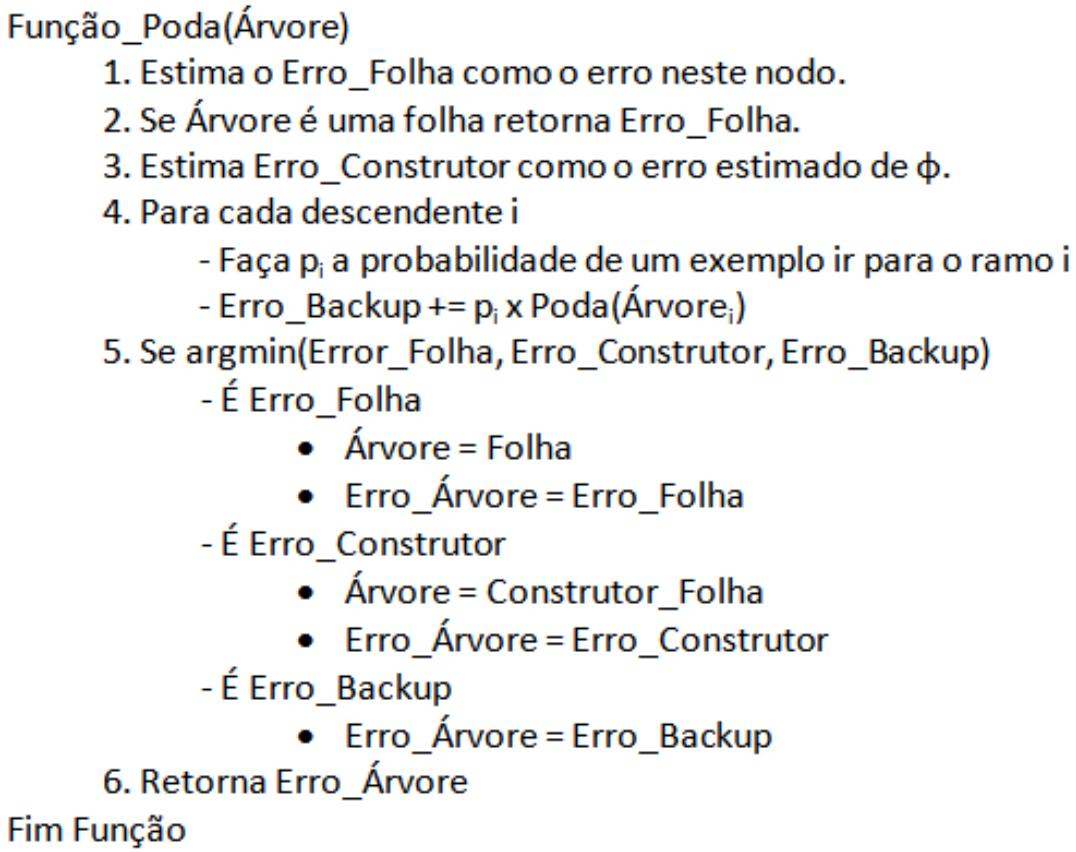

Figura 6: Poda de uma Árvore Funcional [19]

uma folha que contém a classe majoritária do nó. O aspecto fundamental do algoritmo de poda é a estimativa de erro na etapa 1. Em cada nó, é necessário calcular a probabilidade de erro dado o erro na amostra de exemplos que caem neste nó. A probabilidade de erro não pode ser determinada exatamente. Para um dado nível de confiança podemos obter um intervalo de confiança $[L_cf; U_cf]$ que, com probabilidade $1 - cf$ contém o erro verdadeiro. O limite superior do intervalo de confiança U_cf é usado como uma estimativa pessimista para o erro verdadeiro. Existe uma abordagem, chamada de *FT-Leaves*, na qual os modelos funcionais não são utilizados na divisão de teste, mas podem ser usados em folhas. No algoritmo de árvore isto é feito através da restrição da seleção do atributo de teste (passo 4 no algoritmo de crescimento) para os atributos originais. No entanto, ainda é construída, em cada nó, a função de construtor. O modelo construído pelo função de construtor é usado posteriormente na fase de poda. Desta forma, todos os nós de decisão são baseados nos atributos originais. Um nodo folha contém um modelo de construtor se e somente se no algoritmo de poda o erro estimado do modelo construtor é menor do que o erro de *backup* e do que o erro estático [19].

2.5.3 Classificadores Bayesianos

Os classificadores Bayesianos são classificadores estatísticos com base no teorema de Bayes, que preveem a probabilidade de uma tupla pertencer a uma determinada classe. Da mesma forma que as árvores de decisão e os classificadores baseados em Redes Neurais, quando aplicados em grandes bases de dados, os classificadores Bayesianos (como o *Bayesian Naïves* e redes Bayesianas) mostram alta precisão e velocidade. O Teorema de Bayes dá a probabilidade *a posteriori* de um evento H condicionada por X, P(H—X). Isso requer a probabilidade *a priori* de H, P(H), a probabilidade posterior de X condicionadas em H, P(X—H) e a probabilidade anterior de X, P(X). Isto pode ser visualizado na equação 2.1.

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \tag{2.1}$$

Os classificadores Naive Bayesianos, (*naive*, do inglês, ingênuo) assumem independência condicional de classe, o que significa que o efeito do valor de um atributo em uma determinada classe é independente dos valores dos outros atributos [17].

2.5.4 Classificação Naive Bayesiana

Para um conjunto de dados no qual cada tupla é um vetor de dimensão n, $X = (x1, x2, ..., xn)$, representando, respectivamente, n atributos, $A1, A2, ..., An$, tendo C como o vetor de classe com dimensão m, $C = C1, C2, ..., Cm$, o classificador Naive Bayesiano funciona dessa forma:

- (1) Para cada tupla, X, o classificador irá predizer que X pertence à classe que possui a maior probabilidade, condicionada em X, se e somente se:

$$P(C_i|X) > P(C_j|X) \text{ for } 1 \leq j \leq m, j \neq i \tag{2.2}$$

  Sendo assim, $P(Ci|X)$ é maximizada pelo Teorema de Bayes,

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \tag{2.3}$$

- (2) Somente $P(X|Ci)P(Ci)$ precisa ser maximizada porque $P(X)$ é constante para todas as classes, e se as probabilidades anteriores da classe são desconhecidas, assume-se que as classes são equivalentes, portanto, $P(X|Ci)$ será maximizada.

- (3)É feita uma suposição de independência condicional de classe, reduzindo a computação em $P(X|Ci)$. Portanto, os valores dos atributos são presumivelmente condicionalmente independentes uns dos outros, dado o rótulo da classe da tupla.

- (4) Para a previsão do rótulo da classe, $P(X|Ci)P(Ci)$ é avaliada para cada classe Ci, para que o rótulo da classe da tupla X seja previsto como a classe Ci para a qual $P(X|Ci)P(Ci)$ é máximo [17].

$$P(X|C_i)P(C_i) > P(X|C_j)P(C_j) \text{ for } 1 \leq j \leq m, j \neq i \tag{2.4}$$

2.5.5 Rede Bayesiana

O classificador Naive Bayesiano produz uma estimativa da probabilidade, em vez de classificações rígidas. Para cada valor de classe, ele estima a probabilidade de uma determinada tupla pertencer a essa classe. Além disso, para uma determinada classe de uma tupla, assume-se que os atributos são condicionalmente independentes uns dos outros, o que simplifica a computação. Desenvolvido por Pearl (1995), as redes bayesianas, também conhecidas como Redes de Bayes, são uma alternativa estatística pertencente à família de modelos probabilísticos gráficos que representam um conjunto de variáveis aleatórias nos nodos e suas dependências condicionais nas arestas entre os nós, combinando os princípios da teoria gráfica, da teoria da probabilidade, das Ciências da Computação e da estatística. As Redes de Bayes especificam conjuntos de distribuições de probabilidades condicionais que permitem independências condicionais de classe a serem definidas entre grupos de variáveis. Assim como o algoritmo Naive Bayesiano, as Redes de Bayes também usam métodos estatísticos Bayesianos, oferecendo uma abordagem eficiente e de princípios para evitar a super especialização(*overfitting*) de dados [17].

2.6 Critérios de Avaliação de Algoritmos de Classificação

A maior parte da análise da avaliação começa a partir de uma matriz de confusão, que exibe a quantidade de classificações corretas e incorretas de cada classe, podendo ser vista na figura 7. Os verdadeiros positivos (em inglês, *true positives*, TP) e verdadeiros negativos (do inglês, *true negative*, TN) são as classificações corretas. Um falso positivo (do inglês, *false positive*, FP) ocorre quando o resultado está previsto incorretamente como positivo, quando na verdade é negativo. Um falso negativo (do inglês *false negative*, FN)

ocorre quando o resultado é incorretamente previsto como negativo quando realmente é positivo. A taxa de verdadeiros positivos é igual ao TP dividido pelo número total de positivos, o que é TP + FN, e a taxa de falsos positivos é igual ao FP dividido pelo número total de negativos, FP + TN. A taxa de sucesso geral é o número de classificações corretas dividido pelo número total de classificações:

$$\text{Sucesso Geral} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.5}$$

Finalmente, a taxa de erro é um menos isso [16].

Figura 7: Exemplo de matriz de confusão [20]

2.6.1 Precisão

A precisão de um classificador é o percentual de casos corretamente classificados em um conjunto de teste, medindo quão bem o classificador reconhece casos das diferentes classes [17]. Ou seja, é representada pela mesma fórmula 2.5.

2.6.2 Área sob a Curva ROC

A curva ROC foi desenvolvida no contexto de detecção de sinais eletrônicos e problemas com radares, durante a Segunda Guerra Mundial, com o objetivo de quantificar a habilidade dos operadores dos radares (chamados de *receiver operators*) em distinguir um sinal de um ruído. Esta habilidade era, pois, chamada de *receiver operating characteristic*, ou ROC. Nos anos 70, essa metodologia foi amplamente disseminada dentro da

pesquisa biomédica, com o objetivo de auxiliar a classificação de indivíduos em doentes e não doentes. Para se entender melhor essa curva é necessário compreender dois conceitos, o da sensibilidade e o da especificidade. A sensibilidade é definida como a probabilidade do teste sob investigação fornecer um resultado positivo, dado que indivíduo é realmente portador da enfermidade. Já a especificidade é definida como a probabilidade do teste fornecer um resultado negativo, dado que o indivíduo está livre da enfermidade. A expressão matemática da sensibilidade pode ser vista na equação 2.6 e a da especificidade está na equação 2.7. É importante salientar que essas duas medidas não são calculadas sobre os mesmos indivíduos, ou seja, no cálculo da sensibilidade utilizam-se apenas os indivíduos doentes e no caso da especificidade utilizam-se os não doentes. Portanto, essa duas medidas são independentes entre si. Além disso, ambas as medidas não são afetadas pela prevalência da doença sobre a população [21]. Como o resultado de sistemas de classificação em classes geralmente são contínuos, ou seja, produzem um valor situado dentro de um determinado intervalo contínuo, como [0;1], é necessário definir um ponto de corte, ou um limiar de decisão, para se classificar e contabilizar o número de predições positivas e negativas (como diagnósticos verdadeiros e falsos no caso de ocorrência de uma patologia). Como este limiar pode ser selecionado arbitrariamente, a melhor prática para se comparar o desempenho de diversos sistemas é estudar o efeito de seleção de diversos limiares sobre a saída dos dados. Para cada ponto de corte são calculados valores de sensibilidade e especificidade, que podem então serem dispostos em um gráfico denominado curva ROC, que apresenta no eixo das ordenadas os valores de sensibilidade e nas abscissas o complemento da especificidade, ou seja, o valor (1-especificidade). Um classificador perfeito corresponderia a uma linha horizontal no topo do gráfico, porém esta dificilmente será alcançada. Na prática, curvas consideradas boas estarão entre a linha diagonal e a linha perfeita, onde quanto maior a distância da linha diagonal, melhor o sistema. A linha diagonal indica uma classificação aleatória, ou seja, um sistema que aleatoriamente seleciona saídas como positivas ou negativas, como jogar uma moeda para cima e esperar cara ou coroa. Uma medida padrão para a comparacao de sistemas é a área sob a curva (AUC), que pode ser obtida por métodos de integração numérica, como por exemplo, o método dos trapézios. Teoricamente, quanto maior a AUC, melhor o sistema [20]. Por exemplo, um classificador ideal tem uma AUC de 1 enquanto um classificador mais pobre tem uma área de 0,5 [17]. Um exemplo ilustrativo mostrando as curvas ROC, perfeita, aleatória e boa está na figura 8.

$$\text{Sensibilidade} = \frac{TP}{TP + FN} \tag{2.6}$$

$$\text{Especificidade} = \frac{TN}{TN + FP} \tag{2.7}$$

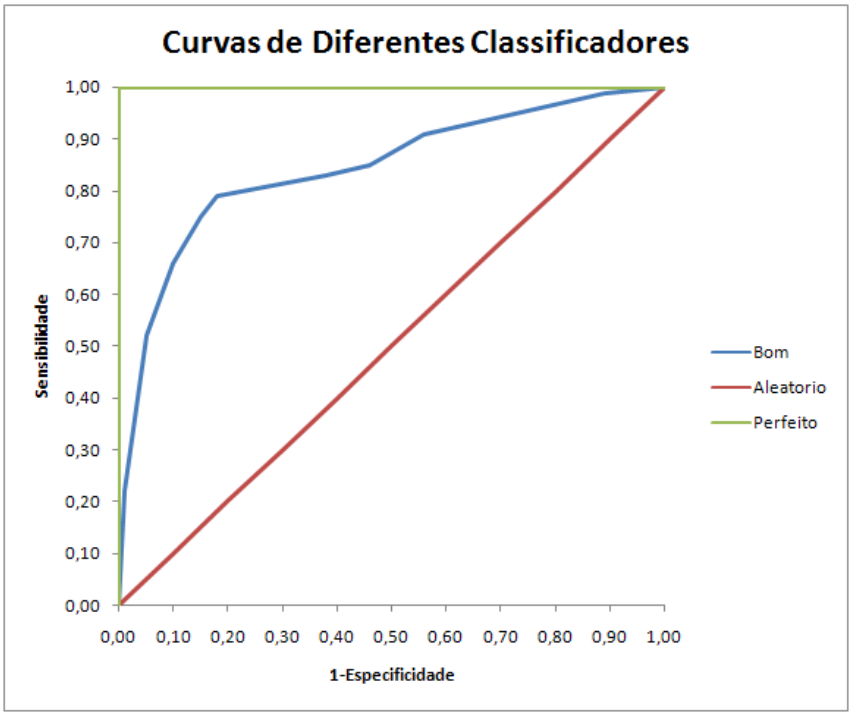

Figura 8: Exemplo de curvas ROC [20]

2.6.3 Estatística *Kappa*

A porcentagem total de casos corretamente classificados reflete uma avaliação simples de um classificador, a mesma avaliação pela área sob a curva de funcionamento do receptor (ROC). Porque um classificador contando com seleção aleatória de casos, com freqüência classifica alguns casos corretamente, o índice *Kappa* é utilizado para controlar essas instâncias que podem ter sido corretamente classificadas apenas por acaso. Além disso, pode ser avaliada a precisão de cada classificador por sua medida-F (do inglês, *F-measure*), que representa a média harmônica entre a precisão e o *recall* [22]. A estatística *Kappa* é utilizada para avaliar a precisão de qualquer caso de medida,e é usualmente utilizada para distinguir entre a confiabilidade dos dados coletados e a sua validade. A pontuação *Kappa* média de um algoritmo para que possa se tornar confiável gira em torno de 0,6-0,7 [23]. Pode ser calculada através da taxa de concordância observada (tco) e da taxa de concordância esperada(tce) se as respostas das duas ocasiões fossem estatisticamente independentes. Como exemplo, utiliza-se a matriz de confusão da figura 9. Nesse caso, a taxa de concordância observada é igual à somas dos valores nos quais as duas

ocasiões classificaram negativos como negativos e positivos como positivos, ou seja, é a soma dos valores da diagonal principal (24 + 17) dividida pelo total de classificações (50). Isso resulta em 0,82. Já para taxa de concordância esperada primeiro se calcula a probabilidade dos casos negativos em relação ao total de casos tanto para a ocasião 1 como para a ocasião 2, ou seja, será igual a 27/50 multiplicado por 30/50. Faz-se o mesmo agora para os casos positivos, e tem-se 23/50 multiplicado por 20/50. Depois, somam-se essas duas quantias e tem-se 0,51 como resultado. Finalmente, o valor de kappa será igual à 0,63 de acordo com a equação 2.8:

$$\text{kappa} = \frac{tco - tce}{1 - tce} \tag{2.8}$$

		Ocasião 1		
		Negativo	Positivo	Total
Ocasião 2	Negativo	24	3	27
	Positivo	6	17	23
	Total	30	20	50

Figura 9: Exemplo de Matriz de Confusão [24]

2.6.4 Medida F

A medida F é usada porque, apesar da precisão e do *recall* serem métricas válidas, uma pode ser otimizada em detrimento da outra. A medida F somente produz um resultado elevado quando tanto a precisão quanto o *recall* estão balanceados, tornando portanto esta medida bastante significativa[25].

A precisão é a proporção de resultados relevantes que estavam corretos:

$$\text{Precisão } (\%) = \frac{TP}{FP + TP} \tag{2.9}$$

Já o *Recall*, é a proporção de resultados relevantes que foram identificados corretamente:

$$\textit{Recall } (\%) = \frac{TP}{FN + TP} \tag{2.10}$$

Por fim, a medida F é derivada dos valores de precisão e *recall*:

$$\text{Medida F } (\%) = \frac{2 * Recall * Precision}{Recall + Precision} \tag{2.11}$$

2.7 Medidas de Interesse para Regras de Associação

Os algoritmos de descoberta de regras fazem uso de medidas de interesse, a fim de diminuir o número de regras geradas na saída de seus algoritmos. As medidas de interesse universalmente mais utilizadas são o apoio (ou suporte) e a confiança. O suporte é uma medida que avalia a frequência com que os termos de uma regra aparecem nos dados. Em outras palavras, o número de transações nas quais os itens presentes na regra aparecem ao mesmo tempo nos dados. Já a confiança é uma medida que se refere a um valor de correspondência entre os itens que compõem uma regra. Então, esta expressa a porcentagem de transações nas quais, tendo o antecedente, o conseqüente também existe [26].

2.8 Regras de Associação com o Algoritmo *Apriori*

Uma das mais populares abordagens de Mineração de Dados é encontrar conjuntos de itens freqüentes de uma transação em um conjunto de dados e extrair regras de associação. Encontrar os conjuntos de itens freqüentes (conjuntos de itens com freqüência maior ou igual que um suporte mínimo indicado pelo usuário) não é trivial devido à sua explosão combinacional. Uma vez que os conjuntos de itens freqüentes são obtidos, é simples gerar regras de associação com confiança maior que ou igual ao valor mínimo de suporte indicado pelo usuário. O algoritmo *Apriori* é um algoritmo que permite encontrar conjuntos de itens freqüentes usando geração de candidatos. Caracteriza-se como um algoritmo de busca de nível completo, utilizando anti-monotonicidade dos conjuntos de itens, "se um conjunto de itens não é freqüente, qualquer um dos seus conjuntos nunca é freqüente". Por convenção, o *Apriori* assume que os itens dentro de uma transação ou conjunto de itens são classificados em ordem lexicográfica. Diga-se que o conjunto de itens freqüentes de tamanho K seja F_k e seus candidatos sejam C_k. O *Apriori* primeiro verifica o banco de dados e procura por conjuntos de itens freqüentes de tamanho 1, acumulando a contagem para cada item e recolhendo aqueles que satisfazem o requisito de suporte mínimo. Em seguida, ele itera as três etapas seguintes e extrai todos os conjuntos de itens frequentes.

1. Gera $C_k + 1$, candidatos de conjuntos de itens freqüentes de tamanho $k + 1$, do

conjunto de itens frequentes de tamanho k.

2. Verifica o banco de dados e calcula o suporte de cada candidato de conjunto de itens freqüentes.

3. Adiciona esses conjuntos de itens que satisfazem o requisito mínimo de suporte à $F_k + 1$.

Finalmente, muitos dos algoritmos de pesquisa por padrões como árvore de decisão, regras de classificação e técnicas de agrupamento que são frequentemente utilizados na Mineração de Dados foram desenvolvidas na comunidade de pesquisa em aprendizado de máquina. A mineração de padrões frequentes e de regras de associação é uma das raras exceções a essa tradição. A introdução desta técnica aumentou a pesquisa em Mineração de Dados e seu impacto é enorme. O algoritmo é bastante simples e fácil de implementar. Experimentos com o algoritmo Apriori são a primeira ação que os mineradores de dados devem tentar fazer [27].

3 Proposta

A presente dissertação é desenvolvida com a proposta de, fundamentalmente, demonstrar a possibilidade de aplicação de conceitos da área da Tecnologia da Informação na Medicina, a fim de agregar a esta última, melhorias nos seus processos de diagnóstico e de extração de conhecimento médico. Em contrapartida, os estudos necessários para tornar possível essa aproximação entre conceitos da Computação com a prática médica, levam os profissionais da Informática a obter uma maior compreensão sobre os temas correlatos, ou seja, os que se referem à Mineração de Dados. Por isto, propõe-se fazer utilização de três bases de dados relacionadas ao tratamento medicinal, e que contenham dados extraídos de exames ou consultas médicas aplicados sobre um determinado número de pacientes, abordando-as de duas maneiras distintas:

- A primeira abordagem, objetivando a predição dos diagnósticos das doenças alvo dessas bases de dados.
- A segunda abordagem, objetivando a extração de regras de relacionamento entre os elementos levantados durante os exames ou consultas e que estão implícitos aos médicos e profissionais relacionados.

Tais propostas de abordagem são computacionalmente possíveis graças à aplicação de técnicas de aprendizado de máquina sobre essas bases de dados, aplicação esta que chamamos de Mineração de Dados, tema já elucidado no capítulo de Fundamentos Teóricos. A predição de diagnósticos aqui proposta é realizada empregando-se algoritmos de classificação de bases de dados, que são responsáveis por, a partir dos exemplos de dados e diagnósticos atribuídos a um conjunto de pacientes, construir um modelo que representa como esses dados se relacionam. Ou seja, um algoritmo de classificação faz com que uma máquina possa aprender como os diagnósticos existentes foram gerados e, quando futuramente essa máquina receber novos dados sobre novos pacientes, a mesma fará uso do modelo aprendido para efetuar novos diagnósticos. Como dentro da área de aprendizado de máquina, de onde provém o desenvolvimento e estudo desses algoritmos de

classificação de dados, existem diversas propostas de classificadores distintos, propõe-se a realização de experimentos comparativos do desempenho de alguns desses classificadores sobre as três bases de dados a serem trabalhadas. Analogamente, a área de aprendizado de máquina também oferece algoritmos de associação de dados, que serão aplicados nesse trabalho para a realização da segunda abordagem, a de extração de conhecimento. Para a definição do algoritmo mais adequado a ser aplicado, propõe-se um estudo da bibliografia relacionada, que demonstra as vantagens da utilização de determinado algoritmo em detrimento de outros. Sendo assim, essas duas propostas de estudo são os referentes ao objetivo 1 já citado no capítulo de Objetivos deste documento.

Para a realização do segundo objetivo, que prevê a preparação das bases de dados para a futura utilização por parte dos profissionais da área médica, projeta-se o uso das técnicas de pré-processamento de bases de dados, referidas nos processos de seleção e de transformação constituintes do processo de extração de conhecimento de bancos de dados, o KDD já referenciado neste volume. A vantagem dessas etapas de pré-processamento consiste na remoção de campos de dados não interessantes ao processo de diagnóstico e que, se considerados, degradam significativamente o processo de aprendizado de máquina.

Dessa forma, as bases de dados pré-processadas no objetivo 2, são o esteio sobre o qual os estudos propostos no objetivo 1 realizar-se-ão. Cabe salientar que os algoritmos de Mineração de Dados obtidos do objetivo 2 são, de fato, os elementos responsáveis pela etapa de busca de padrões de dados do processo de KDD, e por isso, sendo considerado o cerne desse processo.

Assim sendo, dando continuidade à sequencia de etapas do processo de KDD, faz-se necessário que os produtos da etapa de Mineração de Dados sejam interpretados pelos profissionais da área médica, para que estes possam desfrutar dos benefícios apresentados na seção de justificativas deste documento nas suas atividades cotidianas. Para tal, propõe-se a implementação de uma ferramenta baseada nos dispositivos móveis em voga atualmente (*smartphones* e *tablets*), que permite aos seus usuários duas *interfaces* de utilização:

- Entrada de dados provenientes de exames ou consultas médicas de pacientes para a geração dos diagnósticos sobre estes.
- *Interface* para a consulta às regras de relacionamento (conhecimento) acerca das doenças representadas pelas bases de dados alvo desse projeto de pesquisa.

Propõe-se, portanto, que a primeira *interface* agregue precisão e agilidade aos processos de diagnóstico médico, significando uma segunda opinião da qual o médico poderá lançar mão em poucos segundos quando o mesmo julgar necessário. Ao passo que a segunda *interface* propõe a possibilidade de um aprimoramento na prospecção de novos pontos de pesquisa da área médica, visto que dentre as regras de relacionamento a respeito das doenças alvo, podem existir informações observáveis nos pacientes, mas que não se sabe exatamente o porquê das suas causas. Além disso, dada a crescente velocidade com que evoluem os conhecimentos na área médica, essa *interface* de consulta serve como um caminho a ser seguido pelo profissional para a atualização do seu entendimento sobre as doenças em questão. Nas figuras 10 e 11 são mostrados um diagrama de blocos relacionando os tópicos de Mineração de Dados com o funcionamento em alto nível da solução proposta e uma imagem do ambiente de utilização dessa solução.

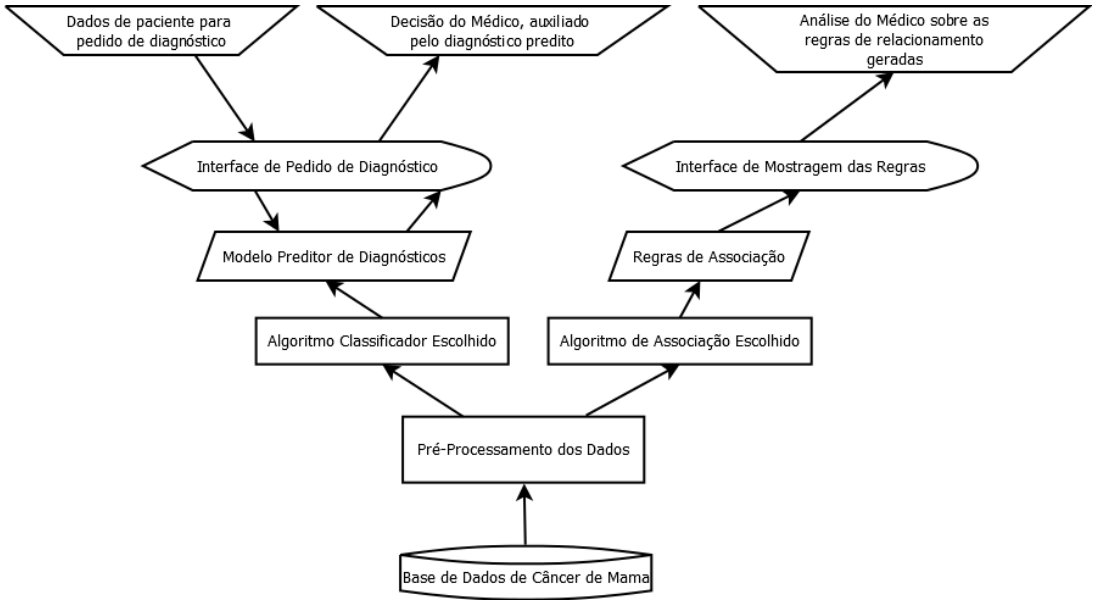

Figura 10: Diagrama de blocos da proposta da solução de diagnóstico e de geração de regras de relacionamento para uma das bases de dados.

Figura 11: Modelo de ambiente no qual a proposta de solução produzida nessa pesquisa será aplicada.

Na figura 10, na parte esquerda é demonstrada a solução responsável pelo auxílio de diagnóstico, partindo da base de dados, passando pelo pré-processamento desta, seguindo da aplicação do algoritmo de classificação escolhido e chegando à interface Android que leva o resultado até o médico especialista. Analogamente, na parte direita da figura, o algoritmo de associação é aplicado e as regras de relacionamento entre atributos são mostradas para o médico. Na figura 11 é possível visualizar a arquitetura de funcionamento da solução como um todo, enfatizando a existência de diversos dispositivos móveis fazendo requisições de processamento para um servidor central que acessa as bases de dados médicas e realiza a aplicação dos algoritmos de Mineração de Dados.

4 Materiais e Métodos

4.1 Ferramentas

4.1.1 WEKA

De acordo com [16], a experiência mostra que nenhum esquema de aprendizagem de máquina sozinho é adequado para todos os problemas de Mineração de Dados. O aprendiz universal é uma situação ideal, visto que os conjuntos de dados reais variam, e para obter modelos precisos o viés de cada algoritmo de aprendizagem deve coincidir com a estrutura do domínio dos dados. Então, a Mineração de Dados é uma ciência fundamentalmente experimental. Para auxiliar na construção desses experimentos, fez-se uso da ferramenta *Weka*, que por sua vez é uma coleção do estado-da-arte de algoritmos de aprendizagem de máquina e de ferramentas de pré-processamento de dados. Ele é projetado para que se possam experimentar rapidamente os métodos existentes de mineração sobre novos conjuntos de dados de forma flexível. A ferramenta fornece suporte extensivo para todo o processo de Mineração de Dados experimental, incluindo a preparação da entrada de dados, avaliando os sistemas de aprendizagem estatisticamente e visualizando os dados de entrada e os resultados da aprendizagem. Este diversificado e abrangente conjunto de ferramentas é acessado através de uma *interface* comum para que os usuários possam comparar diferentes métodos e identificar aqueles que são mais adequados para o problema em questão. O *Weka* foi desenvolvido na Universidade de Waikato, na Nova Zelândia, e o seu nome significa *Waikato Environment for Knowledge Analysis* [16]. Essa ferramenta é implementada em *Java* e é distribuída sob os termos da *General Public License*. O *Weka* disponibiliza as implementações de algoritmos de aprendizado de máquina a fim de que estas possam ser aplicadas sobre as bases de dados em estudo. Além disso, também provê ferramentas para o pré-processamento e transformação dessas bases, com por exemplo, algoritmos de discretização de atributos. Dentre os métodos de Mineração de Dados oferecidos estão os de regressão, classificação, associação, agrupamento, entre outros. O formato tradicional de arquivo que o *Weka* interpreta é o *ARFF*, mas também aceita

formatos mais comumente utilizados como o *CSV*, do inglês, *Comma-Separated Values*. A tela inicial do *Weka*, vista na figura 67, mostra as quatro aplicações que compõem a ferramenta, que são o modo *Explorer*, o *Experimenter*, o *Knowledge Flow* e o *SimpleCLI*. O modo *Explorer* é utilizado para aplicar os métodos de Mineração de Dados e de pré-processamento sobre uma base de dados específica. Ao utilizar o *Explorer*, e selecionar determinado conjunto de dados, o *Weka* mostra ao usuário quantos e quais são os atributos que definem cada instância da base e o seus tipos (nominais, numéricos, etc). Além disso, para cada atributo são mostradas informações como o valor mínimo e máximo do atributo, sua média e o seu desvio padrão. Também é informado se existem instâncias com atributos ausentes, quanto valores de atributos são distintos ou únicos. Essas informações são importantes pois, a partir da análise destas, o pesquisador começa a traçar qual será a estratégia de pré-processamento da base de dados alvo. Na figura 68, são mostradas as informações citadas, que ficam na aba de pré-processamento do modo *Explorer* do *Weka*. Nessa figura é importante destacar que a escolha dos filtros de pré-processamento comentados é realizada a partir do botão *Choose* (no canto superior esquerdo da figura), dentro da seção *Filter*. Das demais abas do modo *Explorer*, para as tarefas de classificação e de associação de bases de dados, utilizam-se as abas *Classify* e *Associate*. Na figura 69 visualiza-se a *Interface* disponibilizada pelo *Weka* para a aplicação de algoritmos de classificação sobre a base de dados escolhida previamente na aba *Preprocess*. Para tanto, na seção *Classifier* (no canto superior esquerdo da tela), é feita a escolha do algoritmo a ser empregado. Clicando duas vezes sobre o nome do algoritmo escolhido, é possível alterar os parâmetros de configuração existentes a fim de refinar a classificação dos dados. Logo abaixo, é possível escolher qual o tipo de teste que será aplicado para determinar a avaliação do desempenho do algoritmo. Pode-se escolher entre usar o próprio corpo de dados de treino como dados de teste, indicar o uso de um arquivo com instâncias par teste em separado, usar o *cross-validation* ou ainda determinar um percentual de divisão da base de dados carregado na aba *Preprocess* entre dados para treino e dados para teste. Após essas escolhas, ao usuário clicar no botão *Start*, aparece no painel à direita, a saída correspondente à execução da tarefa de classificação da base de dados configurada. Nela é possível verificar os resultados que algumas métricas de desempenho geraram, como por exemplo a matriz de confusão. Já na figura 70, é possível visualizar a aba *Associate*, que permite ao usuário aplicar algoritmos de associação sobre a base de dados carregada na aba *Preprocess* do *Weka*. Aqui, na seção *Associator* (no canto superior esquerdo da tela), é feita a escolha do algoritmo de associação a ser aplicado. Nesse caso, para alterar as configurações do algoritmo também basta clicar duas vezes sobre o nome do algoritmo

escolhido, assim como na aba *Classify*. Realizadas as configurações, o usuário clica no botão *Start* e, no painel à direita da tela são mostradas as regras de associação extraídas a partir da base de dados sob estudo. Retornando aos estudos sobre algoritmos de classificação, quando se faz necessário realizar uma comparação entre diferentes algoritmos, o *Weka* oferece um ambiente gráfico apropriado que pode ser acionado na tela inicial pelo botão *Experimenter*. Ao entrar no *Experimenter*, o usuário visualiza a aba *Setup*, na qual é possível configurar onde será salvo o experimento, (arquivo .exp), qual será o tipo de experimento (divisão da base de dados para treino e teste ou *cross-validation*), qual ou quais as bases de dados a serem experimentadas e quais os algoritmo de classificação (ou regressão) que serão comparados. Após essa configuração, o usuário deve ir para a aba *Run*, na qual é disparado o experimento, ou seja, os algoritmos selecionados serão aplicados sobre as bases de dados alvo, armazenando os resultados de desempenho percebidos. A aba *Run* mostra um relatório da execução do experimento, indicando a presença ou não de erros. O conteúdo dessa aba pode ser visto na figura 72. Por fim, na aba *Analyse*, clica-se no botão *Experiment* (no canto superior direito da tela) para que sejam carregados os resultados de desempenho do experimento que foi concluído na aba *Run*. Então, existem algumas configurações a serem realizadas nos botões dispostos do lado esquerdo da tela para que se indique qual será o campo de comparação dos algoritmos e qual será o teste de significância a ser utilizado. É sobre esse teste que o *Weka* irá determinar qual ou quais algoritmos tiveram desempenho significativamente superior ou inferior em relação ao algoritmo base da comparação (que é selecionado através do botão *Test base*). No painel do lado direito da tela são mostrados os resultados comparativos para análise do pesquisador. Sobre as demais opções da tela inicial do *Weka*, o *Knowledge Flow* é responsável por permitir que os filtros de pré-processamento e os algoritmos de classificação e associação sejam aplicados à base de dados através de uma *interface* gráfica estilo *drag and drop*, na qual os blocos responsáveis pela filtragem e pela Mineração de Dados são conectados por setas indicando o fluxo do experimento. Essa representação do *Knowledge Flow* pode ser visto na figura 74. Quanto ao *SimpleCLI*, este se trata de um terminal para a execução dos experimentos já citados, só que apenas permitindo ao usuário entrar com linhas de comandos, sem *interface* gráfica.

Após a demonstração dos diferentes tipos de serviços providos pela ferramenta *Weka* através da sua *interface* gráfica, é importante ressaltar que os algoritmos dos filtros, dos classificadores, associadores, entre outros métodos de Mineração de Dados e afins, são disponibilizados via uma API, do inglês *Application Programming Interface*. Sendo assim, é possível fazer uso dessa API para implementar programas de computador capazes

de executar as diferentes etapas do processo de descoberta de conhecimento de bases de dados. Essa API é escrita em linguagem *Java*.

Algumas telas do *software Weka* são mostradas no Apêndice desse trabalho, a fim de ilustrá-lo melhor.

4.1.2 Sistema Operacional Android

Tecnicamente, o *Android* é uma pilha de *software* para dispositivos móveis que inclui aplicações de um sistema operacional. O *Android SDK* fornece as ferramentas e as APIs necessárias para começar o desenvolvimento na plataforma, através do uso da linguagem de programação *Java*.

- Características do *Android*
  - *Framework* de Aplicação: Permitindo a reutilização e a substituição de componentes.
  - Suporte Java: o código implementado em *Java* pode ser compilado para ser executado na máquina virtual *Dalvik*, a qual é uma máquina virtual especialmente projetada para o uso em dispositivos móveis.
  - Armazenamento: o *software* de banco de dados *SQLite* é usado para o armazenamento de dados.
  - Suporte adicional de *hardware*: possibilitando o uso de câmeras, telas sensíveis ao toque, GPS, acelerômetros, magnetômetros e aceleração de gráficos 3D.
  - Ambiente de desenvolvimento: incluindo um emulador de dispositivo móvel, ferramentas de depuração e um *plug-in* para o *Eclipse IDE*.
  - Conectividade: o *Android* suporta tecnologias como GSM, EDGE, CDMA, UMTS, Bluetooth e Wi-Fi.
  - Loja virtual de aplicativos: A *Google Play* é um catálogo de aplicativos que podem ser baixados e instalados diretamente no dispositivo *Android*, sem o uso de um computador para tal.

- Arquitetura *Android*

  O diagrama a seguir mostra os componentes do sistema operacional *Android*:

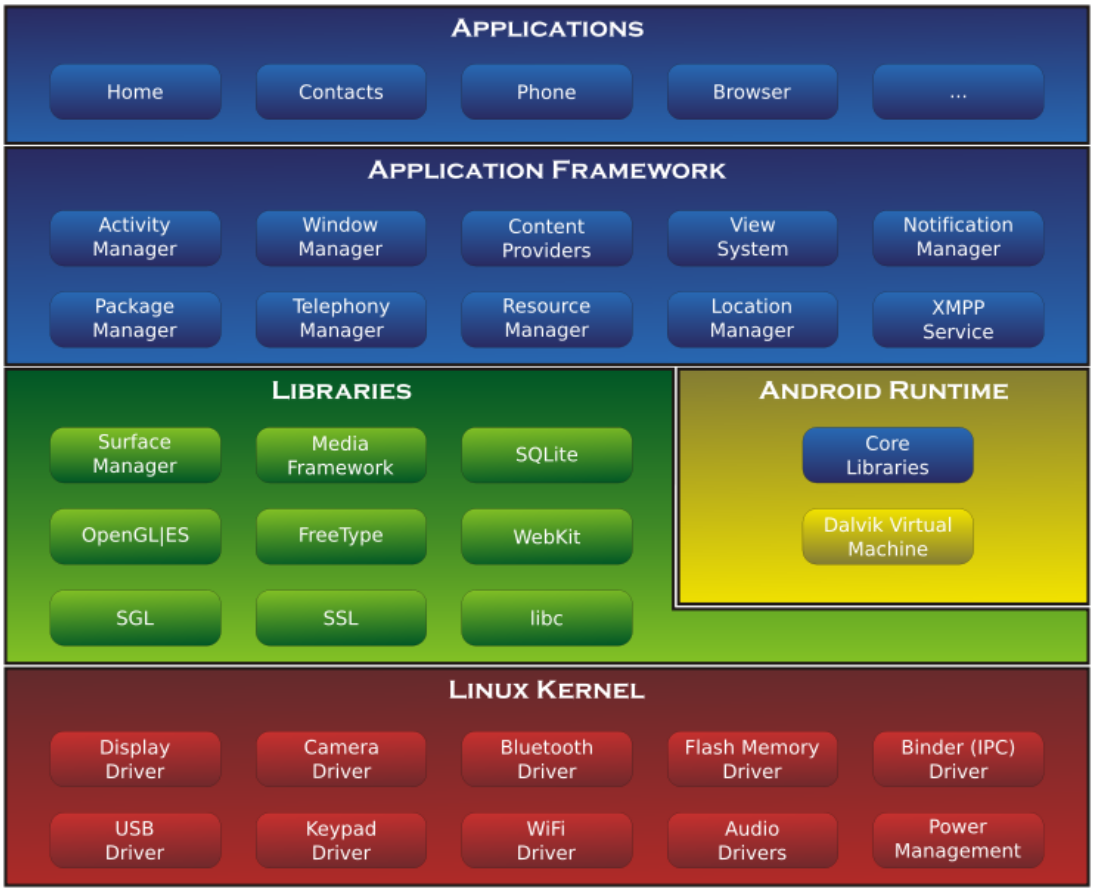

Figura 12: Arquitetura *Android* [12]

– Aplicações: o *Android* é lançado com um conjunto de aplicativos essenciais, incluindo um cliente de *e-mail*, calendário, mapas, navegador de *Internet*, contatos, e outros. Todas as aplicações são escritas usando *Java*.

– *Framework* de aplicação: os desenvolvedores têm acesso total à mesma estrutura de APIs usada pelos aplicativos principais. A arquitetura de aplicação é projetada para simplificar a reutilização de componentes. Qualquer aplicação pode publicar suas capacidades e qualquer outra aplicação pode então fazer uso dessas capacidades. Por exemplo, um desenvolvedor pode usar seu próprio aplicativo de contatos, em vez dos contatos principais de outra aplicação.

– Bibliotecas: o *Android* inclui um conjunto de bibliotecas na linguagem de programação C/C++ usadas por vários componentes do sistema *Android*. Algumas das principais bibliotecas são a de System C, as bibliotecas de Mídia, LibWebCore, SGL, bibliotecas 3D e SQLite.

– *Android Runtime*: o *Android* inclui um conjunto de bibliotecas que fornece a maioria das funcionalidades disponíveis nas principais bibliotecas do *Java* e

cada aplicação *Android* é executada em seu próprio processo, com sua própria instância da máquina virtual *Dalvik*, que conta com o kernel do Linux para funcionalidades como gerenciamento de memória e paralelismo.

– *Kernel Linux*: o *Android* conta com um *Linux* versão 2.6 para os serviços essenciais do sistema. O *kernel* também atua como uma camada de abstração entre o *hardware* e do resto da pilha de *software* [12].

4.2 Bases de Dados

As bases de dados utilizadas nesse trabalho e que serão apresentadas a seguir, foram selecionadas do repositório de aprendizado de máquina da Universidade de Irvine, na Califórnia. Dentre as bases relacionadas com a área médica existentes, foram escolhidas as que tinham uma relação entre o número de atributos e instâncias que não comprometesse o processamento de Mineração de Dados. Por exemplo, foram evitadas a utilização de bases com poucos atributos, o que não gera muitas regras de associação, e também as que contém poucas instâncias, visto que tais bases possuem poucos exemplos de pacientes para a aprendizagem do modelo de classificação.

4.2.1 Tumor de Mama

Esse banco de dados de tumor de mama foi obtido a partir da Universidade de Wisconsin, em Madison, Winsconsin, EUA, do Dr. William H. Wolberg e foi doado em 15 de julho de 1992 para o repositório de aprendizado de máquina da Universidade da California Irvine [28]. Cada instância tem duas classes possíveis: a de tumor benigno ou de tumor maligno. O número total de casos é de 699 e o número de atributos é 10, mais o atributo de classe. Os atributos são explicados na tabela 2:

4.2.2 Dermatologia

Os proprietários originais deste conjunto de dados, que pode ser acessado em [29], são Nilsel Ilter, MD, Ph.D., da Universidade de Gazi, e Altay H. Guvenir, PhD., da Bilkent University, Departamento de Ciência da Computação e Engenharia da Informação, sendo todas essas instituições situadas em Ancara, Turquia. O conjunto de dados foi doado em janeiro de 1998. Essa base de dados contém 34 atributos, sendo 33 dos atributos lineares e um deles nominal. O diagnóstico diferencial de doenças eritêmato-escamosa é um ver-

Atributo	Domínio
Código da amostra do paciente	número de identificação
Clump Thickness	1 - 10
Uniformity of Cell Size	1 - 10
Uniformity of Cell Shape	1 - 10
Marginal Adhesion	1 - 10
Single Epithelial Cell Size	1 - 10
Bare Nuclei	1 - 10
Bland Chromatin	1 - 10
Normal Nucleoli	1 - 10
Mitoses	1 - 10
Class	(2 for benign, 4 for malignant)

Tabela 2: Atributos da Base de Dados de Tumor de Mama

dadeiro problema em dermatologia. Todos eles compartilham as características clínicas da eritema e da descamação, com diferenças muito pequenas. As doenças neste grupo são a psoríase, a dermatite seborreica, o líquen plano, a pitiríase rósea, a dermatite crônica e a pitiríase rubra pilar. Normalmente, uma biópsia é necessária para o diagnóstico, mas infelizmente essas doenças compartilham muitas características histopatológicas. Outra dificuldade para o diagnóstico diferencial é que uma doença pode apresentar as características de outra doença na fase inicial e pode ter as características específicas nas fases seguintes. Os pacientes foram avaliados clinicamente primeiro com 12 características. Em seguida, amostras de pele foram tomadas para a avaliação de 22 histopatologias. Os valores das características histopatológicas são determinados por uma análise das amostras sob um microscópio. No conjunto de dados construído para este domínio, o recurso de histórico familiar tem o valor 1, se qualquer uma destas doenças tem sido observada na família, e zero caso contrário. O recurso de idade representa simplesmente a idade do paciente. Todas as outras características (clínico e histopatológicas) foram avaliadas num intervalo de 0 a 3. Aqui, zero indica que o recurso não estava presente, 3 indica a maior quantidade possível, e 1 e 2 indicam os valores intermediários. O número total de ocorrências são 366 e o número de atributos são 34. Os atributos clínicos assumem valores 0, 1, 2, 3, salvo indicação em contrário. A distribuição de classe é mostrada na tabela 3:

4.2.3 Coluna Vertebral

Os doadores desse conjunto de dados são Guilherme de Alencar Barreto e Ajalmar Rêgo da Rocha Neto, do Departamento de Engenharia Teleinformática, da Universidade Federal do Ceará, Fortaleza, Ceará, Brasil, e Henrique Antonio Fonseca da Mota Filho, do

Código da Classe	Classe	Número de Instâncias
1	psoriasis	112
2	seboreic dermatitis	61
3	lichen planus	72
4	pityriasis rosea	49
5	cronic dermatitis	52
6	pityriasis rubra pilaris	20

Tabela 3: Atributos da Base de Dados sobre Dermatologia

Hospital Monte Klinikum, em Fortaleza, Ceará, Brasil. Esse conjunto de dados biomédicos foi construído pelo Dr. Henrique da Mota, durante um período de residência médica no Grupo de Pesquisa Aplicada em Ortopedia (GARO) do Centro Médico-Cirúrgica de readaptação des Massues, Lyon, França. Os dados foram organizados em duas tarefas de classificação diferentes, mas relacionadas. A primeira tarefa consiste em classificar os pacientes como pertencentes a uma de três categorias: normal (100 pacientes), hérnia de disco (60 pacientes) ou Espondilolistese (150 pacientes). Para a segunda tarefa, a hérnia de disco e espondilolistese foram fundidas em uma única categoria rotulada como anormal. Assim, a segunda tarefa consiste em classificar os pacientes como pertencentes a uma de duas categorias: Normal (100 pacientes) ou anormal (210 pacientes). Nesse trabalho, foi utilizada a versão do conjunto de dados com 3 categorias. Cada paciente é representado no conjunto de dados de seis atributos biomecânicos derivadas da forma e da orientação da coluna vertebral lombar e da pelve (por esta ordem): incidência pélvica, a inclinação da pelve, lordose lombar, inclinação do sacro, raio pélvico e grau de espondilolistese. A convenção a seguir é usada para os rótulos de classe: DH (hérnia de disco), espondilolistese (SL), Normal (NO) e anormal (AB). Essa base de dados pode ser acessada em [30].

4.3 Metodologia da Dissertação

A partir da contextualização do trabalho e das justificativas que suportam a definição dos objetivos anteriormente explanados, pretende-se nessa seção apresentar as metodologias que serão empregadas para atingir as metas traçadas.

4.3.1 Metodologia para o Atingimento do Objetivo 1

Para realizar o estudo sobre a Mineração de Dados, primeiramente é necessário recorrer a uma pesquisa bibliográfica a fim de definir, dentro das diferentes técnicas de mineração, quais as que são mais apropriadas para extração de conhecimentos e predição

de diagnósticos a partir de bases de dados. Esse estudo é auxiliado pela ferramenta Weka, um *software* de código aberto e que permite a execução de algoritmos de Mineração de Dados e a posterior avaliação dos resultados obtidos. O Weka será melhor detalhado dentro do capítulo de Materiais e Métodos. As bases de dados médicas que são analisadas nesse projeto, são oriundas do repositório de aprendizado de máquina da Universidade da Califórnia Irvine. Portanto, nesse repositório, foram selecionadas quais as bases que pertenciam à área médica, visto que o mesmo possui bases relacionadas aos mais variados temas. Depois, dentre as bases médicas, foram feitas algumas avaliações quanto à quantidade de características que definem cada entrada da base, o número total de entradas da base, a quantidade de classes que definem a classificação das entradas e se necessitam ou não de operações de filtragem como etapa de pré-processamento. Ou seja, bases com um número muito grande de características (por exemplo, 200) e dividida em um total de entradas em torno de 400, podem apresentar resultados pobres quando minerados, visto que existem poucos exemplos do problema para extrair informações significativas de um número tão elevado de características. Entre as bases de dados que possuem um número de entradas similar, prefere-se utilizar as que tenham maior número de características, pois dessa forma a extração de regras de relacionamento entre tais características será mais rica, e também porque geralmente as bases de dados médicas costumam possuir números elevados dessas características. Por fim, pretende-se também usar nesse trabalho as bases que contém entradas classificadas com um número diferente de classes. Por isso, as bases a serem utilizadas são as seguintes:

- Tumor de Mama: Essa base de dados tem as suas entradas classificadas em dois tipos diferentes de classes, que são o tumor benigno e o tumor maligno.
- Dermatologia: Essa base de dados possui um total de seis classes diferentes para determinar qual a doença dermatológica relacionada para cada entrada.
- Coluna Vertebral: A base de dados de coluna vertebral possui duas classes que definem condições anormais da coluna e uma classe que define a condição normal da mesma.

Maiores detalhes sobre essas bases de dados encontram-se no capítulo de Materiais e Métodos.

4.3.2 Metodologia para o Atingimento do Objetivo 2

Conforme será visto nos próximos capítulos desse volume, durante o processo de descoberta de conhecimento em bases de dados, existe uma etapa de pré-processamento dos dados, e que é executada anteriormente à etapa de Mineração de Dados. Por isso, essas três bases de dados, são convertidas para um formato reconhecido pela ferramenta de mineração Weka, que neste caso é o csv, do inglês, *comma-separated values*. Depois, são executados os filtros de discretização, de transformação de dados numéricos para nominais, remoção de entradas de dados com valores ausentes ou desnecessários para a mineração. A atuação desses filtros é melhor detalhada na seção de Materiais e Métodos.

4.3.3 Metodologia para o Atingimento do Objetivo 3

Depois do estudo e dos experimentos de Mineração de Dados sobre as três bases de dados escolhidas, pretende-se implementar uma solução que seja capaz de executar dois tipos de tarefas:

- Predição de Diagnósticos: a partir do uso dos algoritmos de classificação de bases de dados, a solução será capaz de, a partir do recebimento de uma nova entrada de dados sobre o paciente, fazer uma predição do seu diagnóstico.
- Extração de Regras de Conhecimento: a partir do uso de algoritmos de associação, a solução será capaz de extrair regras que mostrem os relacionamentos entre as características que compõem as entradas de dados dos pacientes. Essas regras serão repassadas aos profissionais da área médica para que, de alguma forma, possam auxiliá-los em seu trabalho, visto que entre estas regras algumas delas podem trazer à tona informações relevantes que estavam implícitas nas bases de dados.

Para tanto, a implementação dessa solução é dividida em dois aplicativos:

- Aplicativo Android de Interface para a Mineração de Dados: este será o aplicativo a ser executado em dispositivos móveis baseados no sistema operacional *Android*. Será a partir dessa interface que o médico poderá entrar com os dados do paciente e pedir uma predição de diagnóstico ou ainda, pedir a visualização das regras de conhecimento extraídas das bases de dados.
- Aplicação *Java* de Mineração de Dados: Essa aplicação *Java* será responsável por executar os algoritmos de classificação e de associação sobre as bases de dados

selecionadas, conforme as informações recebidas pela interface de usuário *Android*. Por fim, os resultados da execução desses algoritmos são retornados à interface para a utilização e a informação do usuário. A codificação dessa aplicação fará uso da API disponibilizada pela ferramenta *Weka*, que contém as funções necessárias para a execução das tarefas de Mineração de Dados sobre as bases de dados selecionadas.

4.4 Transformação das Bases de Dados

Para esse trabalho foi necessário transformar os conjuntos de dados para um formato de arquivo aceito pelo *Weka*, assim tornando possível o pré-processamento e a mineração dos dados. Assim, os dados foram organizados manualmente em arquivos csv.

4.5 Filtragem para as tarefas de Associação e Classificação de Bases de Dados

Antes de executar a etapa de Mineração de Dados dentro do processo de KDD, existe uma fase de pré-processamento, que, neste caso, será responsável por remover instâncias ausentes e os atributos relacionados com números de identificação dos pacientes (necessário somente na base de dados de tumor de mama) que não são úteis para a tarefa de mineração. Além disso, para a tarefa de classificação especificamente, é necessário efetuar uma conversão do tipo dos atributos de numéricos para nominais, para que a base de dados seja compatível com um número maior de algoritmos de classificação presentes na plataforma *Weka*. Já para a tarefa de associação, aplica-se uma discretização sobre os atributos. Cabe salientar que a discretização dos atributos contínuos dividiu o mesmo em 10 intervalos, conforme o padrão sugerido pelo *Weka*. No entanto, o número de divisões da discretização foi alterado em alguns casos específicos. Para a base de dados de dermatologia, os dados de entrada já eram valores discretos entre 0 e 3, portanto, não foi necessário aplicar uma discretização sobre esses atributos. A exceção ocorreu no atributo idade e, por isso, sobre este atributo, aplicou-se uma discretização que dividiu o valor do atributo em 10 intervalos. Para a base de dados da coluna vertebral, observou-se uma degradação dos resultados dos algoritmos classificadores quando estes eram aplicados sobre a base após ser discretizada. Sendo assim, como a discretização é necessária para a extração de regras usando os algoritmos de associação, a base de dados da coluna vertebral ficará sem a aplicação da discretização. Ou seja, quando for necessária a aplicação de métodos de associação sobre essa base, a discretização da mesma será feita em tempo

de execução pelo módulo de processamento de mineração de dados. Todas essas ações de pré-processamento foram realizadas utilizando os recursos disponíveis da ferramenta *Weka*.

4.6 Bases de Dados de Treino e de Teste

Após a operação de filtragem das bases de dados, cada uma destas foi dividida em dois arquivos de dados distintos: um a ser utilizado para o treino dos modelos de classificação e o outro para servir de corpo de dados para o teste do modelo, ou seja, que é usado para a avaliação do desempenho dos modelos. A divisão das bases de dados é realizada após a filtragem porque se a discretização fosse feita separadamente no conjunto de dados de treino e de teste, os intervalos discretizados muito provavelmente seriam distintos, o que invalidaria o processo de avaliação do desempenho do sistema. A divisão das bases de dados baseia-se na proporção 66/33, ou seja, 66% da base de dados é destinada para o treino e 33% é destinada para o teste. Antes de serem realizadas as divisões, as bases de dados originais foram ordenadas randomicamente para evitar uma desproporção da quantidade de dados de uma mesma classe aparecer somente no conjunto de dados de treino e não no de teste. Isso ocorreria na base de dados da coluna vertebral, visto que a mesma tinha suas instâncias organizadas por ordem de classe. Por fim, essas operações de divisão foram realizadas com o apoio das funções providas pelo *Weka* para tal. Sobre a aplicação dos algoritmos de associação, responsáveis por extrair as regras de conhecimento da base de dados, não é necessário fazer divisão nas bases de dados, visto que o sistema se utiliza de todas as entradas de dados no seu processamento.

4.7 Métodos de Comparação entre Classificadores

Pela adoção da ferramenta *Weka*, foram escolhidos a partir dos algoritmos de classificação disponíveis, os que são capazes de lidar com os tipos de dados utilizados nos conjuntos de dados selecionados. Para escolher o melhor método de classificação para cada conjunto de dados individualmente (tumor de mama, doenças dermatológicas e da coluna vertebral), foram selecionados 12 algoritmos de classificação baseados em árvores de decisão e 3 com base em modelos Bayesianos. Em seguida, foi escolhido o melhor método entre os 12 de árvores de decisão e o melhor entre os 3 Bayesianos. Em seguida, esses dois algoritmos foram comparados a fim de finalmente decidir qual era o melhor para o conjunto de dados em experiência. A lista de algoritmos de árvores de decisão

utilizados é a seguinte:

1. *Best-Firts Decision Trees*
2. *Decision Stump*
3. *Functional Trees*
4. *J48*
5. *J48 graft*
6. *Logistic Model Tree*
7. *Naive-Bayes Trees*
8. *Random Forest*
9. *Random Trees*
10. *SimpleCart*
11. *Logic-Boost Alternating Decision Tree*
12. *REPTree (Fast Decision Tree)*

A lista dos classificadores Bayesianos utilizados é a seguinte:

1. *BayesNet*
2. *NaiveBayes*
3. *NaiveBayesUpdateable*

O critério utilizado em todas as comparações é o conjunto de medidas composto pelo percentual de instâncias corretamente classificadas, a estatística Kappa, a área abaixo da curva ROC e a medida F. A abordagem de teste utilizada baseia-se na divisão das bases de dados em dois grupos, numa proporção de 66/33. Em outras palavras, o treinamento do modelo dos algoritmos foi feito usando 66.6% da base de dados enquanto os restantes 33.3% foram usados para o teste do modelo treinado.

Ao final, escolhido o melhor algoritmo de classificação para cada base de dados, serão testadas diferentes configurações de parâmetros desses algoritmos, a fim de refinar o desempenho dos mesmos e diminuir o número de instâncias classificadas erroneamente.

4.8 Extração de Regras de Associação

Para essa tarefa foi escolhido o algoritmo *Apriori*, pelas razões mencionadas anteriormente neste trabalho. A geração de regras com o *Apriori* é baseada na métrica de confiança. Então, o objetivo é extrair 100 regras com uma confiança mínima de 0,9 pontos.

4.9 Arquitetura da Solução de Predição de Diagnósticos e de Extração de Conhecimento de Bases de Dados Médicas

4.9.1 Processador de Mineração de Dados

O Processador de Mineração de Dados é o módulo responsável por acessar as bases de dados de tumor de mama, de dermatologia e sobre a coluna vertebral, e aplicar sobre elas os métodos de Mineração de Dados escolhidos conforme os experimentos realizados no Capítulo 4. Portanto, são empregados para as atividades de classificação o método de árvores de decisão FT e o classificador Bayesiano *BayesNet*. Já para a execução das tarefas de associação é utilizado o algoritmo *Apriori*. Cabe salientar, portanto, que este módulo de processamento não é responsável por decidir que algoritmos de mineração utilizar, isso porque, tomando por base os resultados dos experimentos desenvolvidos no *Weka*, a diferença de desempenho entre os diferentes modelos testados muitas vezes não é significativa. Ou seja, tentando retreinar os modelos de classificadores de tempos em tempos a fim de decidir qual o melhor a utilizar aumentaria a complexidade computacional da aplicação sem surtir grandes efeitos de melhoria de desempenho da mesma. Portanto, a ideia desse módulo de processamento de mineração é que o mesmo seja uma aplicação *Java* hospedada em um servidor que contém as bases de dados alvo. No caso desse projeto, o servidor é representado por um computador pessoal, e as bases de dados são salvas num diretório previamente definido. Basicamente, esse módulo (servidor) fica em *loop*, esperando as requisições do aplicativo Android de interface do usuário (cliente). A comunicação entre o cliente e o servidor se dá através de um socket, aberto na porta 4444 do computador servidor. O servidor escuta o cliente até que este envie uma *String* com o conteúdo *End of Data*, que denota o final do envio de dados do cliente para o servidor. Esse conjunto de dados enviado pelo cliente é gerado de acordo com as operações executadas pelo usuário. Esse conjunto de dados segue um protocolo simples, que possui um cabeçalho usado para a distinção das operações, e uma seção de dados contendo as

informações do paciente e que são utilizadas somente no caso da predição de diagnósticos. Sendo assim, o pacote de dados se organiza da seguinte forma:

- Modo
  - Médico: este é o modo no qual são executadas as requisições vindas da interface Android. É o processador de Mineração de Dados propriamente dito.
  - Engenheiro: este é utilizado apenas pelo desenvolvedor do projeto para operações de depuração.
- Operação
  - Diagnóstico: representa as operações de diagnóstico de tumor de mama, de doenças dermatológicas e de problemas da coluna vertebral.
  - Extração de Regras: representa a operação de extração de conhecimento baseado nas relações entre os atributos de cada uma das bases de dados supracitadas.
- Doença
  - Tumor de Mama: indica que as operações de diagnóstico ou extração de regras serão executadas sobre a base de dados de tumor de mama.
  - Dermatologia: indica que as operações de diagnóstico ou extração de regras serão executadas sobre a base de dados de problemas dermatológicos.
  - Coluna Vertebral: indica que as operações de diagnóstico ou extração de regras serão executadas sobre a base de dados de problemas da coluna vertebral.
- Dados do Paciente: seção de dados de tamanho variável, conforme a quantidade de atributos de cada base de dados, e que serão somente geradas e processadas nos casos das operações de diagnóstico.

Quando uma operação de diagnóstico é requisitada, o módulo de processamento de mineração executa a função *applyClassificationMethod*, passando como argumentos o nome da base de dados alvo e os dados do paciente. O retorno dessa função é o resultado numérico do índice da classe predita pelo classificador definido para a base de dados. Esse retorno será então enviado via *socket* para a interface do usuário para que esta, por sua vez, mostre o resultado do diagnóstico para o usuário de forma amigável, interpretável pelo médico. Devido à aplicação prévia de filtros de discretização sobre as bases

de dados, algumas funções auxiliares foram desenvolvidas a fim de tratar as conversões necessárias entre os índices utilizados pelo usuário na interface e os índices utilizados na base de dados. Por exemplo, a entrada referente à idade do paciente, no caso da predição de diagnóstico de doenças dermatológicas, deve ser convertida para um valor entre 0 e 9 que corresponde aos 10 intervalos de idade utilizados para discretizar esse atributo.

A aplicação dos métodos de classificação *BayesNet* e *FT* são realizados através da utilização da API da ferramenta *Weka*. Para tanto, implementou-se uma classe chamada *DataMiningMethod*, que contém os métodos utilizados para a aplicação de um método de classificação de Mineração de Dados, e que são os seguintes:

- buildClassifier: método responsável pela construção do classificador utilizando-se das instâncias da base de dados para treino e as correspondentes opções de configuração disponíveis do classificador.
- evaluateCrossValidation: método responsável pela avaliação do desempenho do algoritmo utilizando o método de *cross-validation*.
- evaluateTestSet: realiza a avaliação de desempenho do classificador utilizando uma parte da base de dados para teste.
- classifyInstance: executa a classificação uma instância única. É o método utilizado para predizer o diagnóstico de um paciente.

Sendo assim, as classes *BayesNet* e *TreesFT*, que são referentes à aplicação dos métodos de classificação escolhidos para uso nesse projeto, são especializações da classe *DataMiningMethod*.

Por sua vez, a classe *DataMiningMethod* é uma especialização da classe *InstancesAndOptions*, que possui os métodos referentes à atribuição do conjunto de instâncias a ser utilizado pelo modelo do classificador e as opções de configuração deste.

Também é uma especialização da classe *InstancesAndOptions*, a classe *AssociatorDataMining*, que contém os métodos necessários para a criação de um associador que são os apresentados a seguir:

- buildAssociator: é o responsável pela construção do modelo do associador.
- getAssociatorRules: função que retorna as regras de associação extraídas da base de dados.

Sendo assim, a classe *AssociatorApriori* é uma especialização da classe *AssociatorDataMining*.

Quando essa operação de extração de conhecimento de uma base de dados for solicitada, o módulo de processamento executará a função *getAssociatorRules* do algoritmo *Apriori*. O retorno dessa função será o conjunto de regras geradas para a base de dados indicada e serão encaminhadas para o aplicativo da interface do usuário, via *socket*, para posterior visualização dos usuários, dos médicos ou dos pesquisadores. Na figura 13 é mostrado o diagrama de classe em UML que demonstra com mais clareza essas interações entre as classes, seus atributos e seus métodos:

4.9.2 Interface *Android* de Usuário

Esse módulo da solução tem por objetivo prover a interface de entrada e saída de dados para o usuário que se utiliza de dispositivos móveis *Android*. Por isso, essa implementação divide-se em uma série de classes do tipo *Activity*, que são as classes utilizadas no *Android* para a implementação de cada uma das telas de uma aplicação. Portanto, a classe *UserInterface* implementa a tela inicial do aplicativo, que contém dois botões, um para a entrada no modo Médico e outro para a entrada no modo Engenheiro do aplicativo. Para uso final, essa opção do modo Engenheiro deverá ser deprecada. Entrando no modo Médico, é apresentada ao usuário a tela implementada pela classe *MedicalUserInterface*, que contém três pares de botões, cada par representando cada uma das três doenças alvo desse projeto. Posteriormente, se a escolha do usuário for para a predição de diagnóstico do paciente, é mostrada ao mesmo a tela com vários campos de edição de texto que permitem ao mesmo inserir os dados sobre o paciente para o posterior pedido de diagnóstico. As classes responsáveis por isso, são as chamadas *ColumnLauncher*, *BreastCancerLauncher* e *DermatologyLauncher*. A partir do toque do usuário no botão de pedido de geração de diagnóstico que fica na porção inferior direita da tela, essas classes se comunicam à classe *SocketComm*, que é responsável por efetuar a transmissão, via *socket*, dos dados do paciente para o módulo de processamento de dados anteriormente exposto no presente capítulo. Essa classe se utiliza da classe *InitTask*, que extende a *AsyncTask*, e que permite que se façam operações em *background* em uma aplicação *Android*, alternativamente à criação de *Threads* em Java. Essa separação da execução das tarefas de comunicação via *socket* do processo principal do aplicativo é uma requisição do sistema *Android* que o faz para evitar que possíveis atrasos decorrentes das operações de rede possam ocasionar perda de desempenho no tratamento das tarefas de manipulação das telas da aplicação,

o que é extremamente incômodo ao usuário.

Com o recebimento da resposta do módulo de processamento de Mineração de Dados sobre o diagnóstico do paciente, a classe *ProtocolAndroid* mostra ao usuário o diagnóstico de forma interpretável, ou seja, associa o índice numérico da classificação do paciente à doença que o mesmo está representando.

Para o caso do usuário escolher pelo botão responsável pela extração de conhecimento da base de dados, a classe *MedicalUserInterface* comunica-se diretamente com a classe *SocketComm* para pedir as regras de associação ao módulo de processamento de dados. Por fim, também é através da classe *ProtocolAndroid* que as regras recebidas são mostradas ao usuário.

Na figura 14 mostrado o diagrama de classe em UML que demonstra com mais clareza essas interações entre as classes, seus atributos e seus métodos:

4.9.3 Telas do Aplicativo

Na presente subseção são mostradas as telas do aplicativo na sequência em que são mostradas ao usuário:

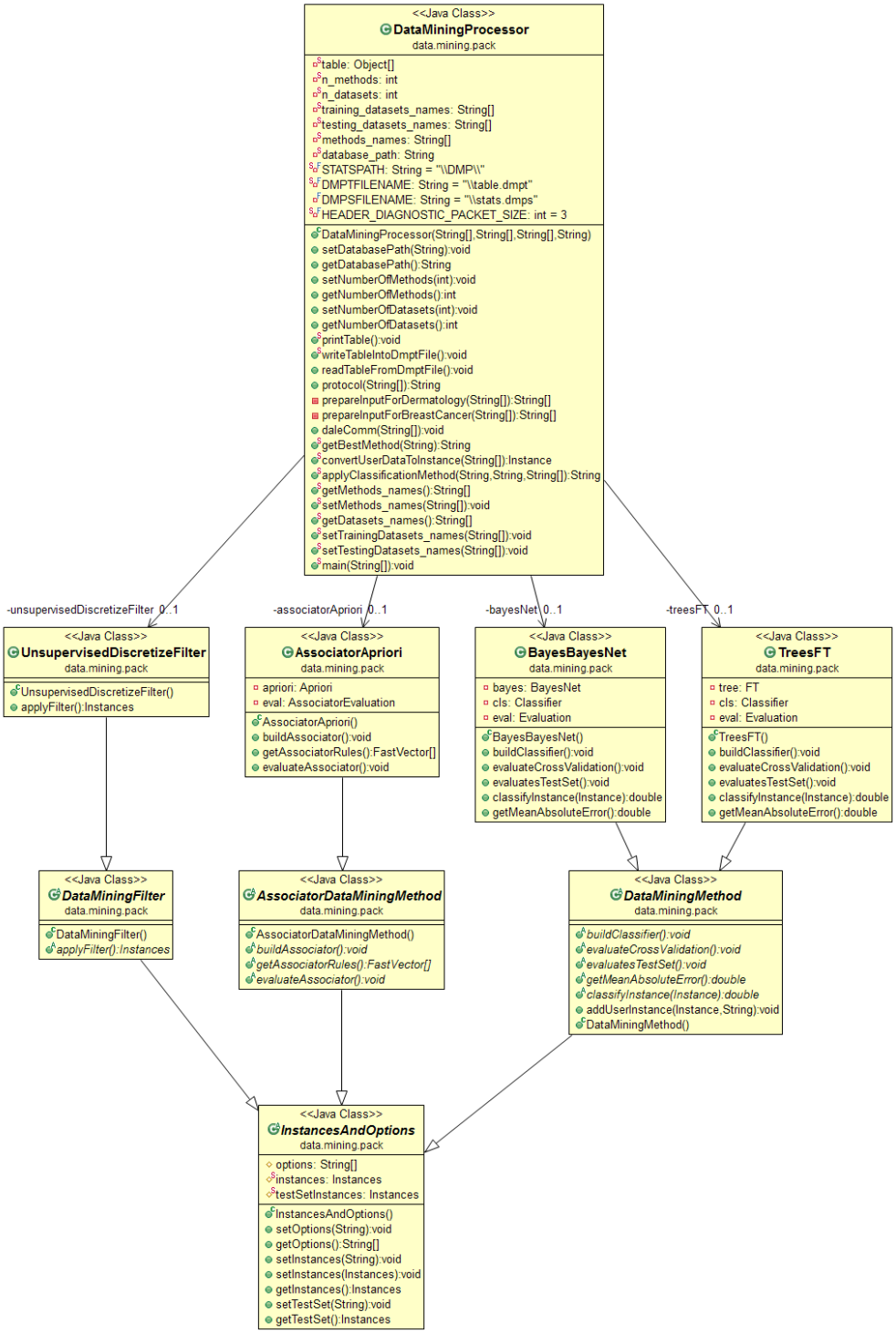

Figura 13: Diagrama de classes UML do módulo de processamento de Mineração de Dados

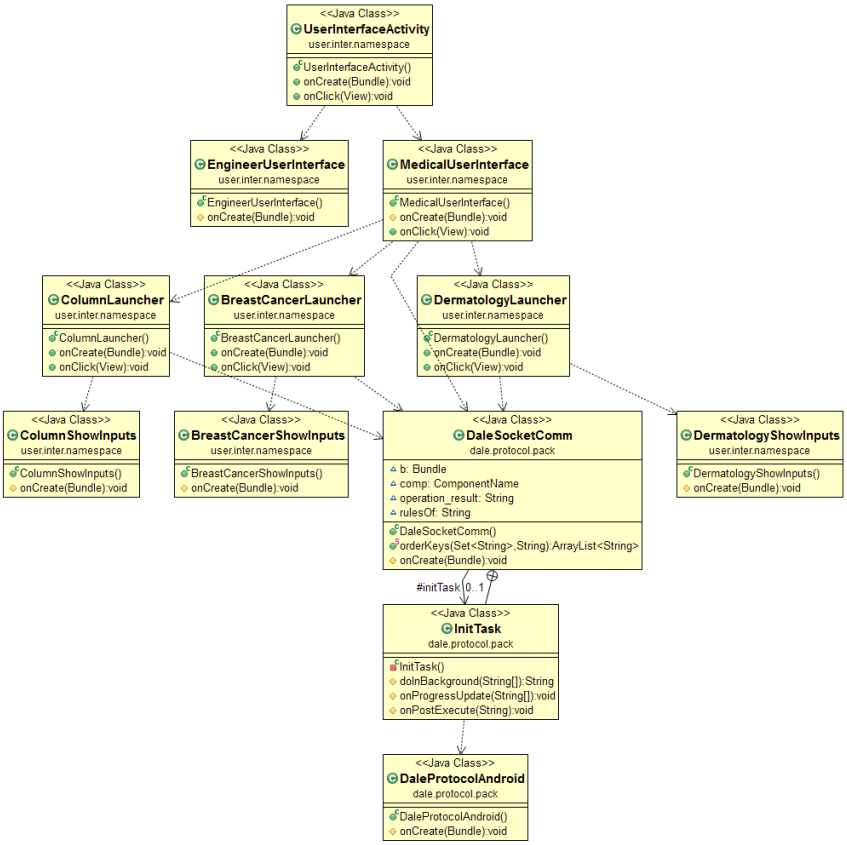

Figura 14: Diagrama de classes UML do módulo de interface de usuário *Android*

Figura 15: Tela inicial do aplicativo

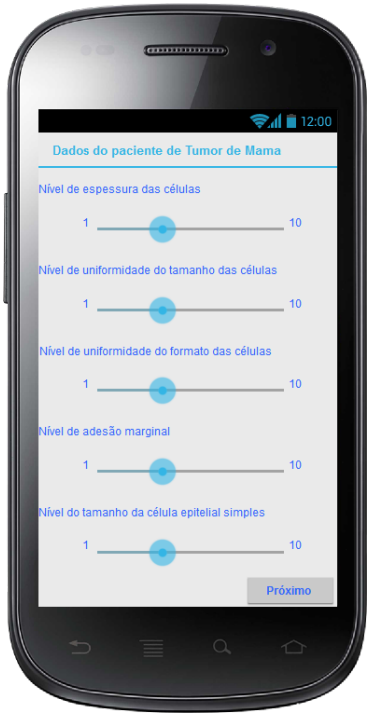

Figura 16: Tela 1 de entrada de dados do paciente com tumor de mama

Figura 17: Tela 2 de entrada de dados do paciente com tumor de mama

Figura 18: Tela de processamento do diagnóstico do paciente com tumor de mama

Figura 19: Tela de resultado de diagnóstico de tumor de mama benigno

Figura 20: Tela de resultado de diagnóstico de tumor de mama maligno

Figura 21: Tela 1 de entrada de dados dermatológicos do paciente

Figura 22: Tela 2 de entrada de dados dermatológicos do paciente

Figura 23: Tela 3 de entrada de dados dermatológicos do paciente

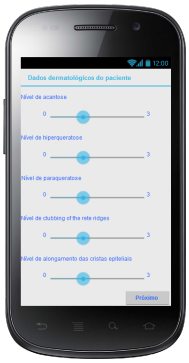

Figura 24: Tela 4 de entrada de dados dermatológicos do paciente

Figura 25: Tela 5 de entrada de dados dermatológicos do paciente

Figura 26: Tela 6 de entrada de dados dermatológicos do paciente

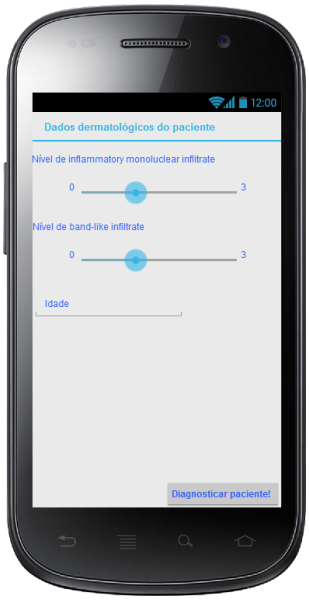

Figura 27: Tela 7 de entrada de dados dermatológicos do paciente

Figura 28: Tela de processamento do diagnóstico dermatológico do paciente

Figura 29: Tela de resultado de diagnóstico de dermatite crônica

Figura 30: Tela de resultado de diagnóstico de dermatite seborréica

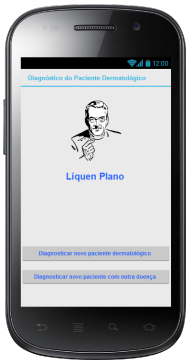

Figura 31: Tela de resultado de diagnóstico de líquen plano

Figura 32: Tela de resultado de diagnóstico de pitiríase rósea

Figura 33: Tela de resultado de diagnóstico de pitiríase rubra pilar

Figura 34: Tela de resultado de diagnóstico de psoríase

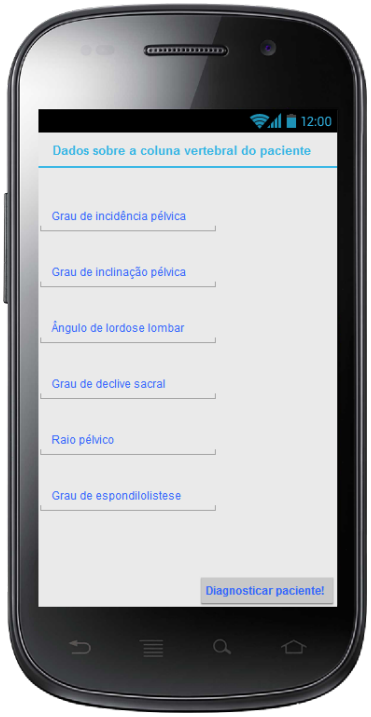

Figura 35: Tela 1 de entrada de dados do paciente de coluna vertebral

Figura 36: Tela de processamento do diagnóstico do paciente de coluna vertebral

Figura 37: Tela de resultado de diagnóstico de hérnia

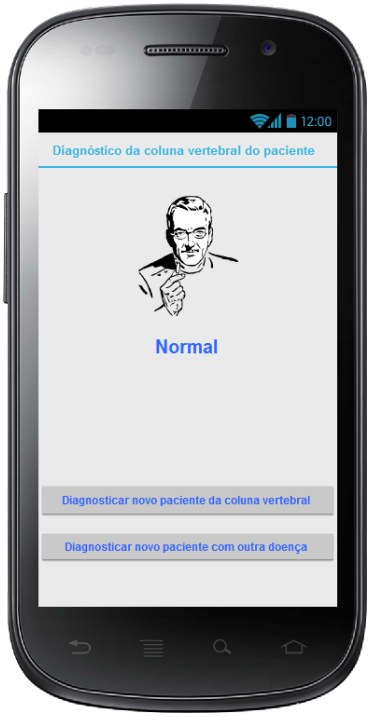

Figura 38: Tela de resultado de diagnóstico normal

Figura 39: Tela de resultado de diagnóstico de espondilolistese

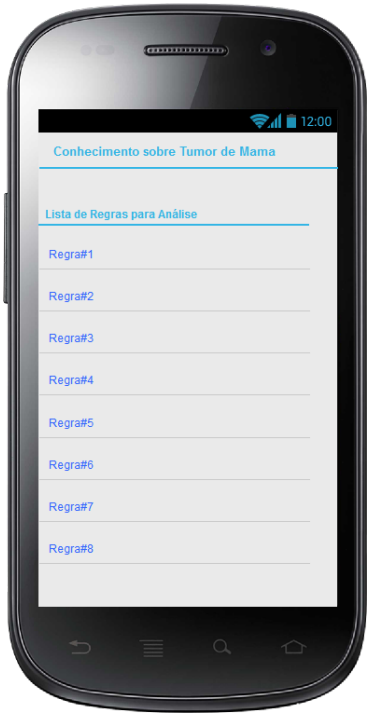

Figura 40: Tela com as regras extraídas da base de dados de tumor de mama

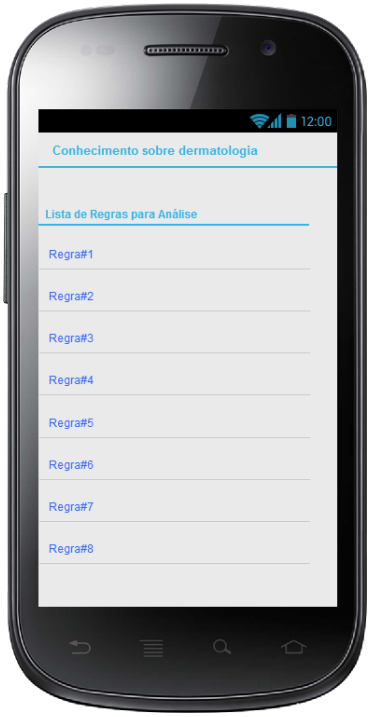

Figura 41: Tela com as regras extraídas da base de dados de dermatologia

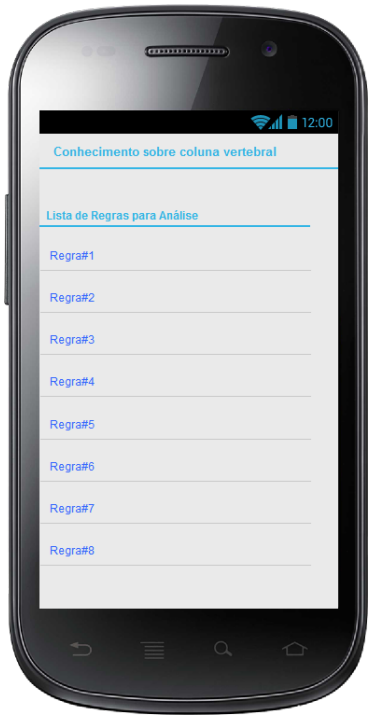

Figura 42: Tela com as regras extraídas da base de dados da coluna vertebral

4.10 Estratégia de Teste da Solução de Predição de Diagnósticos de Bases de Dados Médicas

Dada a arquitetura da implementação da solução de predição de diagnósticos e de extração de conhecimento de bases de dados médicas anteriormente citada, necessita-se de uma estratégia de execução de testes para validar se o funcionamento geral do sistema está de acordo com o projetado inicialmente. Para tanto, cria-se um teste de integração, no qual os módulos do aplicativo *Android* e da aplicação *Java* de processamento de Mineração de Dados são combinados e testados em grupo. O objetivo de se executar testes de integração é verificar os requisitos funcionais, de desempenho e de confiabilidade da modelagem do *software*. Ou seja, utilizando-se esse teste, pode-se encontrar possíveis erros nas *interfaces* entre os componentes do sistema. Para tanto, um projeto de teste foi criado, fazendo uso da classe *ActivityInstrumentationTestCase2* do *Android*. Essa classe provê testes funcionais para uma *Activity*. Portanto, a *Activity* em teste é criada usando a infraestrutura do sistema, permitindo a manipulação da mesma diretamente, através de comandos de instrumentação, chamados pelo método *InstrumentationTestCase.launchActivity()*. Esse teste de integração proposto poderia ser realizado de forma manual ou automatizada. Dadas as quantidades de dados a serem inseridos via *interface* do aplicativo serem grandes, opta-se pela execução automatizada dos testes de integração. Assim sendo, o projeto de teste *Android* criado compõe-se de três rotinas de teste, cada uma com o intuito de testar cada uma das três bases de dados utilizadas pela solução proposta neste documento. Portanto, o algoritmo de teste é semelhante nos três casos, visto que, em linhas gerais, os três testam o sistema a partir da inserção dos dados dos pacientes na tela de informações mostrada ao usuário e, por fim, lêem o resultado do diagnóstico gerado e o comparam com o diagnóstico esperado. Se os resultados, gerado pela solução e o esperado forem iguais, isso indica que o sistema fez uma predição correta, caso contrário a predição foi equivocada. Procurando detalhar essa rotina de teste, seguem os seus principais passos:

- Execução do comando *setActivityInitialTouchMode(false);*: esse comando é responsável por fazer com o que o emulador sobre o qual os testes do aplicativo estão sendo executados não receba cliques externos. Isso serve para evitar que eventuais cliques acidentais na tela da aplicação possam interferir no fluxo de teste.

- *Instrumentation instrumentation = getInstrumentation();*: com esse comando, a rotina de teste poderá em seguida fazer uso dos métodos de instrumentação necessários

para manipular o aplicativo em teste.

- *Instrumentation.ActivityMonitor monitor = instrumentation.addMonitor(UserInterfaceActivity.class.getName(), null, false);*: adiciona um monitor sobre a classe *UserInterfaceActivity*. Isso é feito pela necessidade de monitorar a execução das classes nas quais pretende-se efetuar operações com os seus componentes, tais como botões, caixas de edição de texto, etc.

- ```
  {Intent intent = new Intent(Intent.ACTION_MAIN);
  intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
  intent.setClassName(instrumentation.getTargetContext(),
  UserInterfaceActivity.class.getName());
  instrumentation.startActivitySync(intent);}
  ```

  : esse bloco de código é responsável por iniciar a *Activity UserInterfaceActivity*, que é a que desenha a primeira tela do aplicativo.

- *Activity currentActivity = getInstrumentation().waitForMonitorWithTimeout(monitor, 5);*: esse comando faz com que a rotina de teste aguarde pelo início da *Activity* que está sendo monitorada. Se a *Activity* for iniciada dentro do *timeout* estipulado, esta é guardada na variável *currentActivity*, que por sua vez é testada quanto à sua nulidade no comando *assertNotNull(currentActivity);*.

- *medicalInterfaceButton = currentActivity.findViewById(user.inter.namespace.R.id.meduserinterf);*: como pretende-se efetuar um clique no botão que permite o acesso à tela que contém as operações médicas do aplicativo, executa-se esse comando para associar esse botão na variável *medicalInterfaceButton*.

- *instrumentation.removeMonitor(monitor);*: depois de ter salvo as variáveis dos componentes de interesse da *currentActivity*, remove-se o monitor previamente adicionado.

- *monitor = instrumentation.addMonitor(MedicalUserInterface.class.getName(), null, false);*: novamente, adiciona-se um monitor, agora para controlar a *Activity MedicalUserInterface*, na qual são mostrados ao usuário os botões para o acionamento dos pedidos de diagnóstico.

- *TouchUtils.clickView(this, medicalInterfaceButton);*: nesse momento é que se realiza o clique do botão anteriormente salvo na variável *medicalInterfaceButton*. É importante salientar aqui que sempre precisa-se primeiro adicionar o monitor para
  ```

a próxima tela, antes de efetuar, de fato, o clique no botão que realiza a transição para a próxima tela.

A partir desse momento, prossegue-se com comandos similares com os mostrados anteriormente, para que se siga no fluxo de telas até a que contém os campos de edição de texto nos quais os dados dos pacientes são digitados. Esses dados dos pacientes são armazenados em três arquivos distintos, um para cada base de dados, e localizam-se no cartão de memória do emulador utilizado para os testes. Portanto, um laço é criado para ler cada uma das linhas do arquivo de teste, visto que cada linha contém os dados referentes a um paciente único. Depois que os campos de entrada de dados são preenchidos, a rotina de teste clica no botão responsável pelo pedido de geração de diagnóstico. Nesse momento, os dados do paciente são enviados ao módulo *Java* de processamento através de um *socket http*. Depois, a rotina de teste aguarda o retorno do resultado de diagnóstico gerado e o salva numa variável chamada *mClass*. Por fim, o teste compara o conteúdo dessa variável com o da variável *expectedClass*, que contém o valor de diagnóstico esperado para aquele paciente. Em caso de igualdade do conteúdo dessas variáveis, o teste considera que para esse paciente, a predição de diagnóstico foi realizada com sucesso, caso contrário, houve um equívoco na predição. Em seguida, a rotina de teste segue para a próxima linha do arquivo de testes a fim de proceder o seu fluxo. Os frutos da execução da estratégia de testes aqui apresentada são mostrados no capítulo de Resultados.

5 Resultados

Primeiramente, neste capítulo, são reportados os resultados dos experimentos realizados com os algoritmos de classificação e de associação, com o objetivo de escolher quais os melhores entre eles para serem aplicados sobre as bases de dados de tumor de mama, de doenças dermatológicas e de problemas da coluna vertebral. Posteriormente, esses algoritmos considerados melhores têm seus desempenhos refinados a partir de experimentos variando seus parâmetros de configuração.

Por fim, são mostrados os resultados obtidos dos testes de integração da solução de predição de diagnósticos, solução essa que abrange os módulos da *interface Android* e da aplicação *Java* de processamento de Mineração de Dados.

5.1 Resultados dos Experimentos com os Algoritmos de Classificação

Algoritmo	Precisão	Estatística Kappa	AUC	Medida F
ADTree	94.96	0.89	**0.99**	0.96
BFTree	92.94	0.84	0.92	0.95
Decision Stump	87.86	0.75	0.90	0.90
FT	95.50	0.90	0.98	0.97
Id3	90.25	0.89	0.94	0.96
J48	92.39	0.83	0.96	0.94
J48graft	92.27	0.83	0.96	0.94
LADTRee	94.37	0.87	**0.99**	0.96
LMT	95.04	0.89	**0.99**	0.97
NBTree	**96.81**	**0.93**	**0.99**	**0.98**
Random Forest	96.22	0.92	**0.99**	0.97
Random Tree	92.65	0.84	0.94	0.94
REPTree	93.99	0.87	0.97	0.95
Simple Cart	93.28	0.85	0.93	0.95

Tabela 4: Resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de tumor de mama.

Algoritmo	Precisão	Estatística Kappa	AUC	Medida F
BayesNet	**97.18**	**0.94**	**0.99**	**0.98**
NaiveBayes	97.14	**0.94**	**0.99**	**0.98**
NaiveBayesSimple	97.14	**0.94**	**0.99**	**0.98**
NBUpdateable	97.14	**0.94**	**0.99**	**0.98**

Tabela 5: Resultados de desempenho dos classificadores Bayesianos sobre a base de dados de tumor de mama.

5.1.1 Refinamento dos Parâmetros de Configuração do Algoritmo BayesNet Sobre a Base de Dados de Tumor de Mama

As redes de Bayes podem utilizar distintos algoritmos de busca e de medidas de qualidade. Os parâmetros de configuração disponíveis para o mesmo são os seguintes:

- Estimador: algoritmo selecionado para encontrar as tabelas de probabilidade condicional da Rede de Bayes.
- Algoritmo de Busca: usado para a busca das estruturas da rede.
- Uso da *Alternating Decision Tree*: quando é usada uma *Alternating Decision Tree*(a estrutura de dados para aumentar a velocidade de contagem, não deve ser confundido com o classificador sob o mesmo nome) o tempo de aprendizagem cai, geralmente. No entanto, por apresentarem uso intensivo de memória, problemas podem ocorrer. Mudar esta opção faz com que os algoritmos de aprendizagem fiquem mais lentos.

Através do uso da ferramenta *Weka*, foram realizadas alterações nesses parâmetros, sem que fossem percebidas melhoras no desempenho do algoritmo. Portanto, os parâmetros padrão são mantidos, sendo:

- Estimador: *SimpleEstimator -A 0.5*. O *SimpleEstimator* é usado para estimar as tabelas de probabilidade condicional de uma rede de Bayes uma vez que a estrutura foi aprendida. Tem como opção o chamado *alfa*, que é utilizado para estimar as tabelas de probabilidade e pode ser interpretado como a contagem inicial de cada valor.
- Algoritmo de Busca: *K2 -P 1 -S BAYES*. Este algoritmo de aprendizagem Bayesiano usa o *Hill Climbing* restringido por uma ordem sobre as variáveis.
- Uso da *Alternating Decision Tree*: Falso

Sendo assim, o resultado final do desempenho do algoritmo BayesNet, extraído da ferramenta *Weka* é mostrado no apêndice desse volume.

Algoritmo	Precisão	Estatística Kappa	AUC	medida F
BFTree	95.56	0.94	0.99	0.97
Decision Stump	50.28	0.34	0.90	0.81
FT	**96.77**	**0.96**	**1.00**	**1.00**
Id3	87.93	0.87	0.96	0.96
J48	92.67	0.91	0.97	0.94
J48graft	92.75	0.91	0.97	0.93
LADTree	95.64	0.95	**1.00**	0.99
LMT	96.61	**0.96**	**1.00**	**1.00**
NBTree	93.96	0.92	**1.00**	0.97
Random Forest	93.80	0.92	**1.00**	0.98
Random Tree	87.12	0.84	0.96	0.94
REPTree	84.05	0.80	0.98	0.93
Simple Cart	95.48	0.94	0.99	0.97

Tabela 6: Resultados do desempenho dos algoritmos de árvore de decisão sobre a base de dados de dermatologia.

Algoritmo	Precisão	Estatística Kappa	AUC	Medida F
BayesNet	**98.06**	**0.98**	**1.00**	**1.00**
NaiveBayes	97.74	0.97	**1.00**	**1.00**
NaiveBayesSimple	97.74	0.97	**1.00**	**1.00**
NBUpdateable	97.74	0.97	**1.00**	**1.00**

Tabela 7: Resultados de desempenho dos classificadores Bayesianos sobre a base de dados de dermatologia.

5.1.2 Refinamento dos Parâmetros de Configuração do Algoritmo BayesNet Sobre a Base de Dados de Dermatologia

Para o presente refinamento, são alterados os mesmos parâmetros do caso do tumor de mama, visto que o melhor algoritmo para ambas bases de dados é o mesmo *BayesNet*. Todavia, diferentemente do caso anterior, nos experimentos sobre os dados dermatológicos, a alteração do uso do algoritmo de busca do *K2 -P 1 -S BAYES* para o *TAN -S BAYES*, fez com que o número de instâncias classificadas equivocadamente caísse de 3 para apenas 1. Os demais parâmetros são mantidos em seus valores padrão. No apendice, mostra-se o resultado final do desempenho do algoritmo BayesNet, extraído da ferramenta *Weka*.

Algoritmo	Precisão	Estatística Kappa	AUC	medida F
BFTree	80.84	0.69	0.84	0.61
Decision Stump	77.51	0.63	0.79	0.00
FT	84.44	0.75	0.90	0.67
J48	81.12	0.70	0.82	0.61
J48graft	81.12	0.70	0.82	0.61
LADTree	81.98	0.71	0.89	0.61
LMT	**85.67**	**0.77**	**0.95**	**0.69**
NBTree	79.89	0.68	0.88	0.58
Random Forest	84.06	0.74	0.92	0.67
Random Tree	77.80	0.64	0.70	0.52
REPTree	79.12	0.67	0.84	0.56
Simple Cart	79.61	0.67	0.82	0.51

Tabela 8: Resultados do desempenho dos algoritmo de árvore de decisão sobre a base de dados de problema de coluna vertebral.

5.1.3 Refinamento dos Parâmetros de Configuração do Algoritmo FT Sobre a Base de Dados da Coluna Vertebral

Para o refinamento da classificação da base de dados da coluna vertebral, alteraram-se os seguintes parâmetros do algoritmo de árvore de decisão FT:

Algoritmo	Tempo de Treinamento do Modelo de Aprendizagem Supervisionada em segundos
BFTree	0.22
Decision Stump	0.00
FT	0.12
J48	0.01
J48graft	0.06
LADTree	0.09
LMT	1.32
NBTree	0.38
Random Forest	0.03
Random Tree	0.00
REPTree	0.01
Simple Cart	0.41

Tabela 9: Resultados dos tempos de treinamento do modelo de aprendizagem supervisionada dos algoritmo de árvore de decisão sobre a base de dados de problema de coluna vertebral.

Algoritmo	Precisão	Estatística Kappa	AUC	Medida F
BayesNet	**73.06**	**0.57**	**0.89**	**0.61**
NaiveBayes	72.68	0.56	**0.89**	0.59
NaiveBayesSimple	72.68	0.56	**0.89**	0.59
NBUpdateable	72.68	0.56	**0.89**	0.59

Tabela 10: Resultados do desempenho dos classificadores Bayesianos sobre a base de dados de coluna vertebral.

- *binSplit*: converte todos os atributos nominais para binários antes de construir a árvore. Isto significa que todos os grupos no final serão uma árvore binária.

- *errorOnProbabilities*: minimiza o erro em probabilidades em vez de erros de classificação, quando valida de forma cruzada o número das iterações *LogitBoost*. Quando definido, o número de iterações *LogitBoost* é escolhido o que minimiza o erro quadrático em vez do erro de classificação.

- *minNumInstances*: define o número mínimo de casos em que um nó é considerado para a separação. O valor padrão é 15.

- *modelType*: o tipo do modelo FT. Zero, para FT, 1, para FTLeaves, e 2, para FTInner.

- *numBoostingIterations*: define um número fixo de iterações para o *LogitBoost*. Se maior ou igual a 0, esta define um número fixo de iterações *LogitBoost* que é usado em todos os lugares na árvore. Se menor que 0, o número é validado de forma cruzada.

- *useAIC*: a AIC é usada para determinar quando parar as iterações *LogitBoost*. O padrão é não usar a AIC.

- *weightTrimBeta*: define o valor beta utilizado para corte em peso no *LogitBoost*. Somente as instâncias que transportam (1 - beta) % do peso da iteração anterior são utilizados na iteração seguinte. Definir como 0 para nenhum peso de corte. O valor padrão é 0.

Após os experimentos variando esses parâmetros do FT, percebeu-se que apenas a alteração da configuração *modelType*, para o valor *FTLeaves* foi a que trouxe melhoria no desempenho do algoritmo, que assim passou a classificar erroneamente 13 instâncias, ao invés de 17. Portanto, os demais parâmetros são mantidos, sendo que a configuração final fica:

- *binSplit*: falso.
- *errorOnProbabilities*: falso.
- *minNumInstances*: 15.
- *modelType*: FTLeaves.
- *numBoostingIterations*: 15.
- *useAIC*: falso.
- *weightTrimBeta*: 0.

O resultado final do desempenho do algoritmo FT após refinamento pode ser visto no apêndice desse trabalho.

5.2 Resultados dos Experimentos com Algoritmos de Associação

Como mencionado anteriormente, a geração de regras de associação foi feita considerando como métrica uma confiança mínima de 0.9, e restringindo o número de regras geradas em 100. Essas regras são mostradas, a fim de ilustrar os resultados obtidos, no apêndice desse volume. Para a base de dados de tumor de mama, o algoritmo Apriori obteve 100 regras com confiança acima de 0,9. A 100ª regra foi extraída com confiança de 0,93. Já para a base de dados de dermatologia, também foram extraídas 100 regras, todas com confiança acima dos 0,9 pontos. Por fim, extraiu-se 100 regras de associação para a base de dados de coluna vertebral.

5.3 Resultados dos Testes Automatizados de Integração da Solução

5.3.1 Resultados dos Testes de Integração para a Predição de Diagnóstico sobre a Base de Dados de Tumor de Mama

Considerando-se o arquivo de entradas de dados de pacientes para teste, contendo 233 instâncias, encontraram-se 6 casos de pacientes para os quais a solução errou a predição do diagnóstico, sendo mostrados na tabela 11:

Atributos	Paciente 1	Paciente 2	Paciente 3	Paciente 4	Paciente 5	Paciente 6
Código da amostra do paciente	1231706	242970	1239232	1096352	1017023	1226012
Clump Thickness	8	5	3	6	6	4
Uniformity of Cell Size	4	7	3	3	3	1
Uniformity of Cell Shape	6	7	2	3	3	1
Marginal Adhesion	3	1	6	3	5	3
Single Epithelial Cell Size	3	5	3	3	3	1
Bare Nuclei	1	8	3	2	10	5
Bland Chromatin	4	3	3	6	3	2
Normal Nucleoli	3	4	5	1	5	1
Mitoses	1	1	1	1	3	1
Diagnóstico esperado	Benigno	Benigno	Benigno	Benigno	Benigno	Maligno
Diagnóstico predito	Maligno	Maligno	Maligno	Maligno	Maligno	Benigno

Tabela 11: Pacientes com diagnóstico errado pelo classificador BayesNet

5.3.2 Resultados dos Testes de Integração para a Predição de Diagnóstico sobre a Base de Dados de Dermatologia

Considerando-se o arquivo de entradas de dados de pacientes para teste, contendo 122 instâncias, encontrou-se 1 caso de paciente para o qual a solução errou a predição do diagnóstico, sendo mostrado na tabela 12:

5.3.3 Resultados dos Testes de Integração para a Predição de Diagnóstico sobre a Base de Dados da Coluna Vertebral

Considerando-se o arquivo de entradas de dados de pacientes para teste, contendo 104 instâncias, encontraram-se 13 casos de pacientes para os quais a solução errou a predição do diagnóstico, sendo mostrados nas tabelas 13 e 14:

Atributos	Paciente 1
erythema	3
scaling	3
definite borders	2
itching	2
koebner phenomenon	0
polygonal papules	0
follicular papules	0
oral mucosal involvement	0
knee and elbow involvement	2
scalp involvement	0
family history, (0 or 1)	0
melanin incontinence	0
eosinophils in the infiltrate	0
PNL infiltrate	1
fibrosis of the papillary dermis	0
exocytosis	2
acanthosis	1
hyperkeratosis	0
parakeratosis	2
clubbing of the rete ridges	1
elongation of the rete ridges	1
thinning of the suprapapillary epidermis	1
spongiform pustule	1
munro microabcess	0
focal hypergranulosis	0
disappearance of the granular layer	0
vacuolisation and damage of basal layer	0
spongiosis	0
saw-tooth appearance of retes	0
follicular horn plug	0
perifollicular parakeratosis	0
inflammatory monoluclear inflitrate	2
band-like infiltrate	0
age	46
Diagnóstico esperado	psoriasis
Diagnóstico predito	seboreic dermatitis

Tabela 12: Pacientes com diagnóstico errado pelo classificador BayesNet

Atributos	Paciente 1	Paciente 2	Paciente 3	Paciente 4	Paciente 5	Paciente 6	Paciente 7
incidência pélvica	54.600316	31.276012	56.030218	44.489275	43.349606	45.366754	49.828135
inclinação da pelve	21.488974	3.144669	16.297915	21.786433	7.467469	10.755611	16.736435
lordose lombar	29.360216	32.562996	62.275275	31.474154	28.065483	29.038349	28
inclinação do sacro	33.111342	28.131342	39.732303	22.702842	35.882137	34.611142	33.0917
raio pélvico	118.343321	129.011418	114.023117	113.778494	112.776187	117.270068	121.435559
grau de espondilolistese	-1.471067	3.62302	-2.325684	-0.284129	5.753277	-10.675871	1.913307
Diagnóstico esperado	Normal	Hernia	Hernia	Normal	Hernia	Hernia	Normal
Diagnóstico predito	Hernia	Normal	Normal	Hernia	Normal	Normal	Hernia

Tabela 13: Pacientes com diagnóstico errado pelo classificador FT

Atributos	Paciente 8	Paciente 9	Paciente 10	Paciente 11	Paciente 12	Paciente 13
incidência pélvica	53.936748	76.147212	67.538182	56.103774	64.311867	65.007964
inclinação da pelve	20.721496	21.936186	14.655042	13.106307	26.328369	27.602608
lordose lombar	29.220534	82.961502	58.001429	62.63702	50.958964	50.947519
inclinação do sacro	33.215251	54.211027	52.883139	42.997467	37.983498	37.405357
raio pélvico	114.365845	123.93201	123.63226	116.228503	106.177751	116.581109
grau de espondilolistese	-0.42101	10.431972	25.970206	31.172767	3.118221	7.015978
Diagnóstico esperado	Normal	Spondylolisthesis	Normal	Normal	Normal	Spondylolisthesis
Diagnóstico predito	Hernia	Normal	Spondylolisthesis	Spondylolisthesis	Spondylolisthesis	Hernia

Tabela 14: Pacientes com diagnóstico errado pelo classificador FT

6 Discussões

Primeiramente, dada a intenção de se utilizar alguma metodologia da área de Inteligência Artificial a fim de contribuir positivamente para a área médica, conclui-se que a abordagem de extração de conhecimento sobre bases de dados é a mais apropriada. Isso porque as atividades praticadas nos ambientes clínico e hospitalar geram uma enormidade de dados que são organizados em bancos de dados. Portanto, utilizar-se de métodos da Mineração de Dados, combinados com os filtros de pré e pós processamento de dados, é importante para extrair conhecimentos que estão, de certa forma, escondidos nessas grandes massas de dados.

Quanto aos diversos tipos de métodos de Mineração de Dados existentes, conclui-se que os mais apropriados para a obtenção de conhecimento sobre as bases selecionadas são os que se referem à classificação e à associação. Isto porque o primeiro é responsável por aprender a diagnosticar doenças baseando-se na criação de um modelo de aprendizagem, partindo de uma massa de dados de treino e, em seguida, diagnosticar novas entradas de dados de pacientes. Já o segundo atinge o propósito de extrair relações entre os atributos das bases de dados e em relação com a classe de diagnóstico das doenças analisadas.

Sobre essa tarefa de associação, dados os experimentos realizados com o *Weka*, e que foram relatados na seção de Resultados deste documento, foi possível obter um conjunto de pelo menos 100 regras com confiança acima dos 90% para as três bases de dados em questão. Todavia, para a base de dados de problemas da coluna vertebral, percebeu-se a necessidade de alguns experimentos a mais em relação às demais bases. Isso porque a base de dados da coluna vertebral possui atributos numéricos com valores Reais, ao contrário das demais bases que são constituídas por atributos numéricos, porém discretos. Adicionalmente a isso, a aplicação do algoritmo *Apriori*, que se concluiu ser o mais adequado para utilização através dos estudos feitos sobre a literatura especializada, exige que os atributos sejam discretizados previamente. Portanto, enquanto as bases de dados sobre o tumor de mama e a dermatologia já indicaram, pelas categorias de cada atributo, em quantos intervalos de valor estes atributos deveriam ser discretizados, os atributos Reais

que descrevem as angulações de porções da coluna vertebral precisaram ser discretizados através de experimentos para que se chegasse a um número adequado. Sendo assim, percebeu-se que para o valor padrão do *Weka* de discretização dos atributos em 10 intervalos de valor, o algoritmo *Apriori* apenas conseguiu extrair 18 regras de relacionamento entre os atributos com confiança de 90%. Quando o número de intervalos foi aumentado, o algoritmo passou a encontrar menos relações ainda. Em 5 intervalos, o algoritmo conseguiu obter pelo menos 100 regras de associação, tal como nas demais bases de dados. Por isso, concluiu-se que a discretização de dados, como etapa de pré-processamento em relação à Mineração dos Dados, pode e muito colaborar tanto com o sucesso como com o insucesso de um projeto de extração de conhecimento. É interessante notar esse "cobertor curto" existente na relação entre querer quantizar cada vez mais atributos Reais, a fim de obter mais precisão, mas que em contrapartida, gera muitos intervalos de valor que acabam ocorrendo com baixa frequência na base de dados e levando o algoritmo *Apriori* (ou até outros algoritmos relacionados) a não obterem regras com a confiança desejada de 90% ou mais. Também conclui-se que é interessante a análise de um perito na área de conhecimento sobre esse conjunto de regras que é gerado automaticamente, e por isso, as relações listadas nesse trabalho são, posteriormente, apresentadas aos mesmos para posterior consulta e análise.

As regras mostradas na seção de Resultados da presente pesquisa estão no formato de representação da ferramenta *Weka*, que, conclui-se não ser a forma mais amigável para futuras análises. Por isso, algumas delas serão aqui interpretadas como exemplo, para facilitar o entendimento:

- `UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' 347 ==> Class=2 347 conf:(1):`

  Dessa regra de associação conclui-se que se a uniformidade do tamanho da célula tiver valor 1 (lembrando-se que nessa base de dados os valores dos atributos originais variam entre 1 e 10) e a quantidade de núcleos das células não cercados por citoplasma tiver nível igual a 1, então o diagnóstico do paciente é de tumor benigno (classe de valor igual a 2).

- `UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 327 ==> Mitoses='(-inf-1.9]' Class=2 322 conf:(0.98)`

  Dessa regra conclui-se que se a uniformidade do tamanho da célula for igual a 1, se a quantidade de núcleos não envolvidos por citoplasma tiver nível igual a 1, se o nível de presença de nucléolos for igual a 1, então o nível de mitoses é igual a 1 e o diagnóstico do paciente é de tumor benigno.

- `UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' Class=2 347 ==> NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 322 conf:(0.93)`

> Dessa regra, percebe-se que se a uniformidade do tamanho da célula for de nível 1, se o nível de núcleos não envolvidos por citoplasma for igual a 1 e o paciente tiver diagnóstico de tumor benigno, então o nível de presença de nucléolos nas células é igual a 1 e o nível de mitoses é igual a 1.

Após os exemplos acima citados, e da análise das demais regras apresentadas no capítulo de Resultados, percebe-se que foram extraídas tanto regras que possuem somente atributos no antecedente e diagnóstico no consequente, quanto atributos e diagnóstico no antecedente e atributos no consequente. Justamente essas variações do aparecimento de atributos no consequente e do diagnóstico no antecedente que fazem com que essas regras de associação sejam diferentes das regras de classificação que sempre apresentam atributos no antecedente e somente o diagnóstico no consequente. É importante reiterar que é fundamental que tais regras sejam analisadas por um médico das respectivas áreas da oncologia, da dermatologia e da ortopedia para que possam ser utilizadas ou não nas pesquisas médicas conforme sua relevância prática.

Quanto ao emprego dos algoritmos de classificação de dados empregados na geração de modelos de aprendizagem supervisionada de diagnósticos, percebeu-se que existem diversos tipos, e, por isso, objetivou-se fechar o escopo desta pesquisa sobre dois grupos: o dos que classificam bases de dados utilizando-se da Matemática probabilística baseada nas distribuições de probabilidade de classe (Teoria Bayesiana) e os que se utilizam de uma heurística baseada na construção de árvores de decisão. Concluiu-se que outra abordagem, a do uso de Redes Neurais, não seria tão interessante, visto que necessitam da construção de um modelo de aprendizagem bastante complexo de ser entendido e de construção lenta, o que poderia degradar o tempo de execução de uma ferramenta *online* de execução de pedidos de diagnóstico. Dentre os modelos de árvores de decisão e de lógica Bayesiana, concluiu-se que, se o objetivo for entender como o modelo foi construído para inclusive se basear nele para futuras tomadas de decisão, as árvores de decisão cumprem melhor esse papel, visto que da visualização dos nodos, arestas e folhas da árvore, podem-se gerar regras de classificação da base de dados. É importante salientar que a extração de regras de associação continua tendo sua validade e importância, pois que as regras de classificação relacionam apenas certas combinações de atributos com o valor da classe do problema, ao passo que as de associações também estabelecem relações entre os atributos, sem necessariamente envolver-se com o valor de classe. Isso, por fim, traz mais possibilidades de estudo e pesquisa sobre o problema estudado. E por isso, a conclusão de entregar para análise dos médicos as regras extraídas via métodos de associação em detrimento das resultantes dos métodos de classificação. Voltando a tratar dos algoritmos de classificação,

se os modelos de árvores de decisão são de mais fácil leitura, os modelos de aprendizagem Bayesiana são mais rápidos em sua construção, por basearem-se no cálculo das probabilidades de certo evento ocorrer dada a probabilidade de outro ocorrer. Ou seja, conclui-se que esse se trata de um modelo com algoritmo mais simples de construção, ao passo que as árvores de decisão possuem uma complexidade inerente à construção de estruturas de dados em árvore. Adicionalmente, os modelos Bayesianos como o *Naive Bayes* partem de uma premissa confortável de que não existem dependências entre os atributos, o que ocasionalmente é verdade. Existem também na literatura indicações de que os classificadores Bayesianos são bastante empregados e que apresentam desempenho similar aos de árvores de decisão. Dadas as conclusões sobre esses dois modelos, concluiu-se que a melhor maneira de escolher qual o melhor a ser empregado na ferramenta de geração de diagnósticos ao final de pesquisa seria a partir da experimentação dos diversos algoritmos de árvore de decisão e de modelos Bayesianos sobre as bases de dados. No entanto, existem diversas métricas de possível utilização para comparar o desempenho de classificadores. Da literatura, percebeu-se uma grande utilização apenas da métrica de precisão nas comparações efetuadas. Por isso, nesta pesquisa, procurou-se empregar outras métricas, a fim de saber se existiriam outras que agregassem mais valor à comparação de desempenho. Por isso, nos experimentos realizados, utilizou-se além da precisão, a estatística Kappa, a análise da área sob a curva ROC e a Medida F, por serem métricas que levam em conta a sensibilidade e especificidade dos testes de diagnóstico, e também por basearem seus cálculos a partir dos valores apresentados na matriz de confusão, ou matriz de classificação.

Sendo assim, geraram-se os resultados mostrados no capítulo anterior, utilizando essas quatro métricas, e que levaram às seguintes conclusões:

6.1 Discussão sobre os Experimentos com a Base de Dados de Tumor de Mama

Analisando os resultados para o tumor de mama a partir das tabelas 4 e 5 e dos gráficos 43, 44, 46, 45 , 48, 50, 47 e 49 é percebido que o melhor algoritmo de árvore de decisão é o NBTree, enquanto o melhor classificador Bayesiano é o BayesNet. Finalmente, a comparação entre esses dois algoritmos revela que o BayesNet é o melhor, tendo os mesmos valores para a Medida F e para a AUC, mas com valores mais altos de estatística Kappa e de precisão em relação ao NBTrees.

6.2 Discussão sobre os Experimentos sobre a Base de Dados de Dermatologia

Dos resultados das tabelas 6 e 7 e dos gráficos 51, 52, 54, 53, 56, 58, 55 e 57 conclui-se que o melhor classificador de árvore de decisão para a base de dados de dermatologia é o FT. Já entre os classificadores Bayesianos, o BayesNet é o melhor por uma diferença de 0,1 de precisão. Na comparação entre esses dois algoritmos, o BayesNet leva vantagem em precisão por 0,5 pontos sobre o FT. Para os demais critérios, esses dois algoritmos são equivalentes. Então, o melhor algoritmo de classificação para a base de dados de dermatologia é o BayesNet.

6.3 Discussão sobre os Experimentos com a Base de Dados de Coluna Vertebral

A partir dos resultados das tabelas 8 e 10 e dos gráficos 60, 62, 59, 59, 61, 64, 66, 63 e 65, o melhor classificador de árvore de decisão é o LMT. Entre os classificadores Bayesianos, ambos NaiveBayes e NaiveBayesUpdateable têm o mesmo desempenho para todos os critérios e foram melhores que o BayesNet. Comparando esses dois classificadores Bayesianos com os resultados do LMT, nota-se que o LMT tem os valores maiores de precisão, estatística Kappa e AUC. Então, a princípio, o LMT é o melhor classificador para as atividades de classificação sobre a base de dados de problemas da coluna vertebral. No entanto, analisando-se outro fator importante que é o tempo necessário para a execução do treino do modelo e posterior classificação, percebe-se, pelos resultados da tabela 9, que o modelo de aprendizagem construído no algoritmo LMT é bastante demorado, em comparação com o algoritmo FT, que possui desempenho semelhante. Essa demora não foi significativa, dado o tamanho da base de dados utilizada, porém, se esta base for acrescida com novos dados com o tempo, ou ainda se este algoritmo for aplicado a outra base de dados muito maior, a ferramenta de geração *online* de diagnósticos poderá ter seu desempenho de resposta degradado.

Após essas comparações entre algoritmos, notou-se que somente para a base de dados referente aos problemas da coluna vertebral, o algoritmo *BayesNet* não foi melhor do que alguns modelos de classificadores baseados em árvores de decisão. Desse fato, conclui-se que o motivo dessa diferença está nas diferenças de constituição das bases de dados envolvidas, visto que enquanto a base de dados de coluna vertebral possui dados não discretizados e pertencentes ao conjunto dos números Reais, as outras duas bases possuem

bases mais organizadas, fechadas em atributos categorizados numericamente, o que facilita o algoritmo do modelo de cálculos probabilísticos Bayesianos. Além disso, a árvore de decisão LMT, que se utiliza de funções logísticas em seus nodos internos, parece ser bastante apropriada para lidar com atributos cujos valores são Reais e com uma precisão representada muitas vezes por mais de 4 casas decimais.

Retornando ao assunto das diferentes métricas de desempenho, notou-se que para as métricas de Medida F, de estatística Kappa e da área sob a curva ROC, vários algoritmos tiveram desempenho igual, restando recorrer à métrica de precisão para de fato decidir entre um algoritmo ou outro. Sendo assim, conclui-se que, para as bases de dados utilizadas, seria somente necessária a adoção da métrica de precisão para a escolha dos melhores algoritmos de classificação de dados, embora seja importante utilizar pelo menos uma outra métrica como apoio. Além disso, conclui-se que uma boa métrica de apoio para o problema específico de diagnostico médico é a análise da área sob a curva ROC, pois é bastante empregada no meio médico ao levar em conta a sensibilidade e a especificidade do teste diagnóstico.

Dadas as conclusões sobre os melhores métodos a serem empregados para cada base de dados, cogitou-se a possibilidade de fazer alterações nos parâmetros configuráveis de cada algoritmo a fim de obter um número menor ainda de instâncias sendo equivocadamente classificadas. Observando-se os resultados dos refinamentos mostrados no capítulo de Resultados deste volume, conclui-se que essa abordagem foi importante para tornar o classificador ainda mais preciso, reduzindo significativamente a quantidade de erros de diagnóstico. Todavia, para a base de dados de tumor de mama, alteraram-se as configurações, mas o desempenho do algoritmo não melhorou e sim, em alguns casos, piorou. Portanto, nesse caso, conclui-se que o melhor é aplicar o algoritmo *BayesNet* sobre essa base de dados com os seus parâmetros padrão. Ao passo que o algoritmo *BayesNet* sobre a base de dermatologia, que errava o diagnóstico de três instâncias, passou a errar apenas uma, com a substituição do algoritmo de busca utilizado do K2 para o TAN. Conclui-se que essa melhora foi observada pelo fato de que o algoritmo TAN alivia a restrição existente na construção da estrutura da rede Bayesiana ao permitir a representação de dependências entre os atributos. Ou seja, as redes Bayesianas que utilizam o algoritmo TAN consideram também as dependências entre os outros atributos ao invés de somente contar com o atributo de classe nos cálculos das distribuições de probabilidade. No caso do algoritmo FT sobre a base de dados da coluna vertebral, a modificação do modelo da FT para FTLeaves fez com que o número de instâncias com diagnóstico errado caísse de 17 para 13. Conclui-se que isso ocorreu nesse tipo de árvore funcional porque nela os

modelos funcionais são utilizados não nos testes de divisão de instâncias, mas sim nas folhas da árvore.

Sobre as instâncias classificadas incorretamente, coletadas durante os testes de integração da solução de geração de diagnóstico, e que foram mostradas nas tabelas 11, 12, 13 e 14, algumas conclusões foram obtidas.

Primeiramente, no caso do tumor de mama, existe uma discrepância entre o número de instâncias classificadas incorretamente nos resultados do refinamento e no número levantado pelo teste de integração. Isso ocorreu porque no processamento executado no *Weka*, o filtro de pré-processamento responsável por tratar as instâncias com atributos ausentes não exclui as instâncias, mas sim, atribui um valor para o atributo ausente, que é igual à moda ou à média da base de dados de treino, conforme ess atributo ausente seja nominal ou numérico. Já na implementação dos testes automatizados, o arquivo com as entradas de teste é lido e as instâncias com atributos ausentes são ignoradas. No caso dos diagnósticos da base de dados de tumor de mama, percebeu-se que o classificador errou mais no sentido de confundir os casos benignos por malignos. Apenas um diagnóstico de tumor dito benigno era, na verdade, maligno, ao passo que cinco diagnósticos de tumor maligno foram atribuídos a pacientes com tumor benigno. Dessa situação, observa-se que o erro mais perigoso do ponto de vista do tratamento médico é o do caso único de diagnóstico benigno para o tumor maligno, pois este pode vir a camuflar a real situação do paciente que têm a forma mais agressiva da doença em questão. Esse diagnóstico provavelmente foi equivocado pelo fato dos níveis de uniformidade do tamanho e da forma das células serem iguais a 1, uma característica marcante dos exames dos pacientes com tumor benigno. Essa informação foi extraída da análise do gráfico provido pelo *Weka*, que pode ser visto nas figuras 75 e 76, localizaas no Apêndice desse trabalho. Nele, a porção azul representa a quantidade de pacientes benignos com o determinado valor de atributo e, consequentemente, a porção vermelha representa os casos malignos.

Em contraponto, a situação dos demais equívocos, que diagnosticaram como malignos tumores benignos, levanta um sinal de alerta sobre a possibilidade do paciente ter a forma mais agressiva da doença, o que provavelmente levará este a passar por exames complementares que, mais a frente, poderão fazer com que o médico entende que, na verdade, esses pacientes apresentam a forma mais branda do tumor. Ou seja, pode ser menos pior o erro de diagnóstico que leva a um aprofundamento das investigações sobre a saúde do paciente do que um erro que ofusque uma condição mais grave duma enfermidade.

Sobre o único caso de erro de diagnóstico dos pacientes dermatológicos, conclui-se

que os atributos referentes aos níveis de envolvimento do couro cabeludo (nível zero), da infiltração PNL (nível um) e da exocitose (processo pelo qual uma célula eucariótica viva libera substâncias para o fluido extracelular - nível 2) apresentados pelo paciente com psoríase, são mais comuns em casos de dermatite seborréica e, provavelmente, foram os responsáveis pelo equívoco do modelo de aprendizagem construído.

Sobre os casos de erros de diagnóstico para os pacientes com problemas de coluna vertebral, percebeu-se que todos os casos em que houve erro para o diagnóstico de hérnia, foi gerado um diagnóstico de coluna normal. Já para os casos em que houve erro para o diagnóstico normal, quatro diagnósticos foram de hérnia e três foram de espondilolistese.

Já para os erros envolvendo espondilolistese, um diagnóstico gerado foi de hérnia e outro foi de condição normal da coluna vertebral. Ou seja, o modelo de aprendizado construído para essa base de dados conseguiu aprender melhor a diagnosticar casos de espondilolistese, provavelmente pelo fato de a maior parte dos pacientes dessa base de dados conterem essa anomalia da coluna, o que forneceu mais exemplos para o algoritmo de aprendizagem obter conhecimento sobre essa condição. Além disso, os valores das medidas angulares extraídas dos exames de raio-x da coluna vertebral utilizados como atributos dessa base de dados são mais semelhantes para os casos de coluna normal e com hérnia do que em relação aos casos de espondiloslitese. Para essa última deformidade da coluna, os valores de ângulo de incidência pélvica e do grau de espondilolistese (também chamado de grau de escorregamento de vértebra) são geralmente bem maiores do que nas colunas dos pacientes com hérnia ou sem deformidades.

Cabe salientar que não necessariamente por a base de dados conter uma maioria de casos de espondilolistese que exista, na população de indivíduos em geral, também uma predominância desse tipo de anomalia da coluna vertebral.

7 Conclusões

Sendo assim, dadas as discussões anteriores, percebeu-se que o primeiro objetivo do trabalho do estudo e utilização da Mineração de Dados sobre as bases de dados a fim de extrair conhecimento novo sobre as doenças alvos foi atingida com sucesso.

Sobre o objetivo de preparar as bases de dados para futuras pesquisas sobre as doenças alvo desse trabalho, conclui-se que esse objetivo foi atingido. Agora, a base de dados está pré-processada (os atributos ausentes em instâncias foram substituídos por valores coerentes com a base dados) e dividida em dois corpos de dados, um para treino dos modelos de aprendizagem e outro para o teste do modelo. Além disso, a base de dados completa também é mantida em um arquivo, com seus atributos já discretizados, para o caso de futuras utilizações de algoritmos de extração de regras de associação (assim como o *Apriori*) sobre as doenças alvo desse trabalho.

Quanto ao último objetivo do trabalho, que é a implementação de um aplicativo móvel capaz de auxiliar elaboração de diagnósticos sobre as doenças selecionadas, a ser executada em dispositivos *Android*, percebeu-se a sua viabilidade dada a arquitetura proposta, um modelo cliente-servidor entre o aplicativo *Android* requisitando predições de diagnóstico e um aplicativo *Java* processando as análises sobre as bases de dados. Concluiu-se que foi possível utilizar com sucesso a API provida pelo *Weka* para que fossem empregados no módulo de processamento de Mineração de Dados em *Java* os algoritmos escolhidos como os mais adequados para os objetivos de classificação e de associação das bases de dados a partir dos experimentos realizados de forma *offline* no *Weka*. Também concluiu-se que a solução de auxílio ao diagnóstico consegue realizar as operações pedidas pelo usuário e retornar respostas em tempo hábil (praticamente instantâneo). Todavia, a presente implementação foi toda desenvolvida utilizando um simulador de dispositivo *Android* rodando no mesmo computador em que é executado o aplicativo de processamento de Mineração de Dados. Ou seja, num ambiente real, no qual o aplicativo *Android* estará sendo executado num dispositivo real, comunicando-se com um servidor através de protocolos de comunicação sem fio, o tempo de resposta naturalmente será maior. Somem-se a isso, os

tempos necessários para o acesso do servidor aos repositórios de dados que, muito provavelmente, em ambientes reais, serão bem maiores do que os utilizados neste trabalho de pesquisa. Quanto à *interface* de usuário do aplicativo, procurou-se fazer a mesma da forma mais simples possível, dado que em termos de desenvolvimento de *design*, frequentemente menos é mais. No entanto, somente com a utilização dessa *interface* por usuários reais da área da saúde é que seria possível chegar a uma implementação talvez mais satisfatória. Por isso, o foco maior da solução neste trabalho reside na confiabilidade dos seus resultados, que conforme as análises anteriores, mostrou-se capaz de diagnosticar poucos pacientes de forma equivocada, tornando-a um caso de sucesso entre as aplicações da área de Tecnologia da Informação para a área médica, visto que se propõe a ser um auxiliar ao trabalho do médico e, jamais, se tornar um substituto do mesmo.

APÊNDICE A – Gráficos de desempenho dos algoritmos de classificação

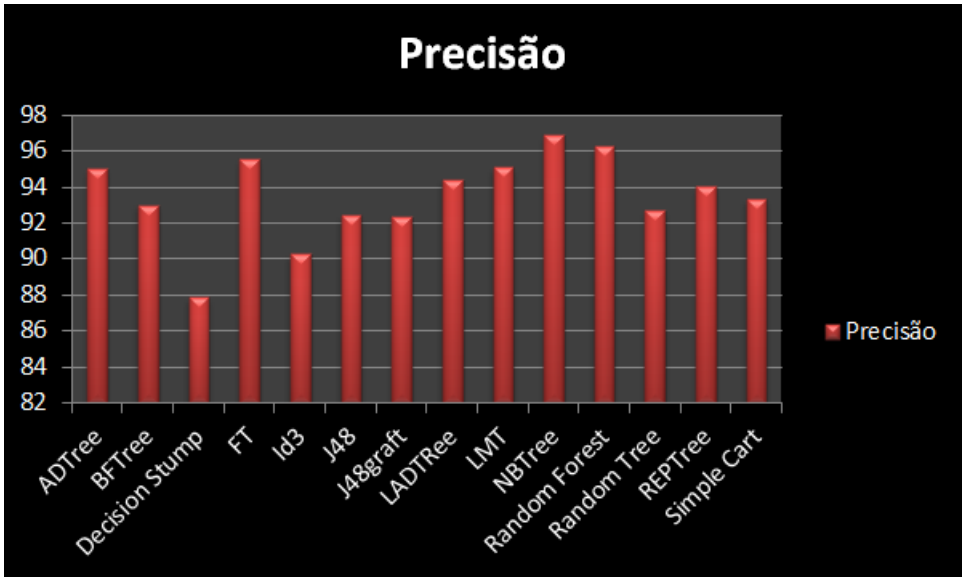

Figura 43: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de câncer de mama em relação ao critério de Precisão.

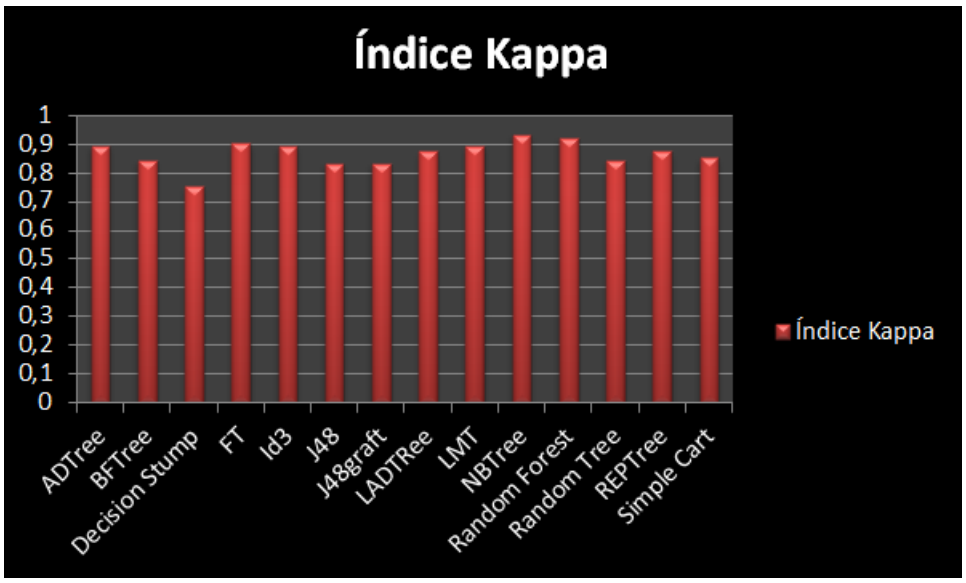

Figura 44: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de câncer de mama em relação ao critério do Índice Kappa.

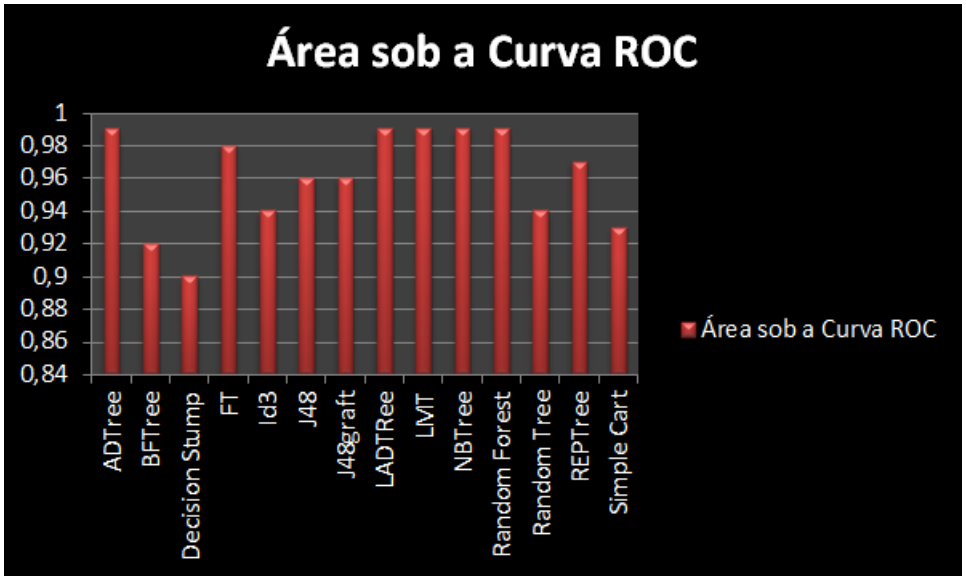

Figura 45: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de câncer de mama em relação ao critério da Área sob a Curva ROC.

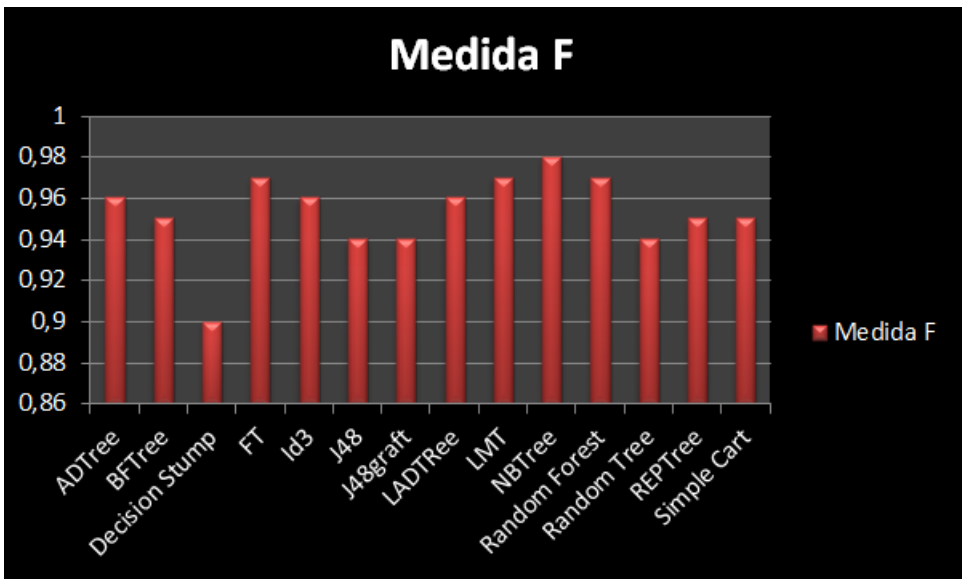

Figura 46: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de câncer de mama em relação ao critério da Medida F.

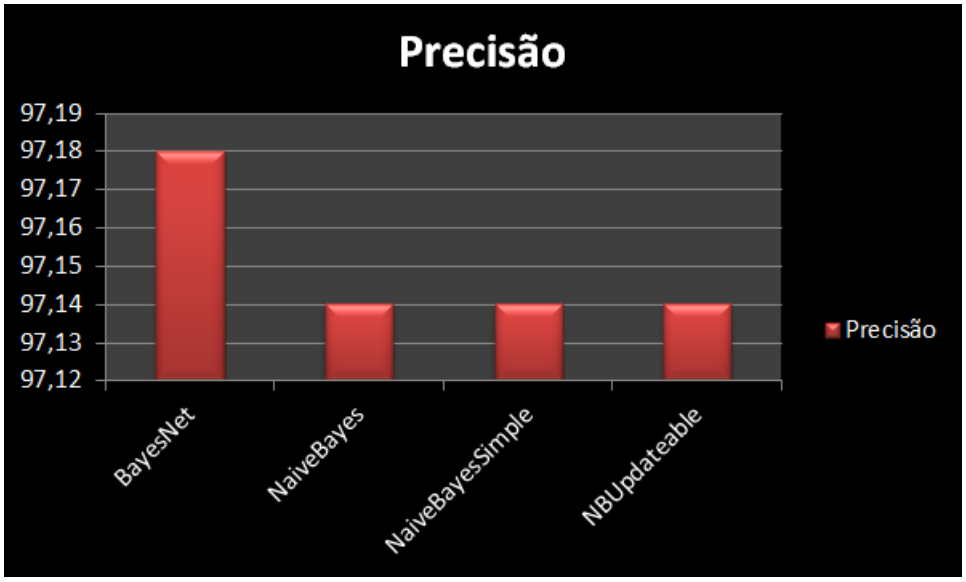

Figura 47: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de câncer de mama em relação ao critério de Precisão.

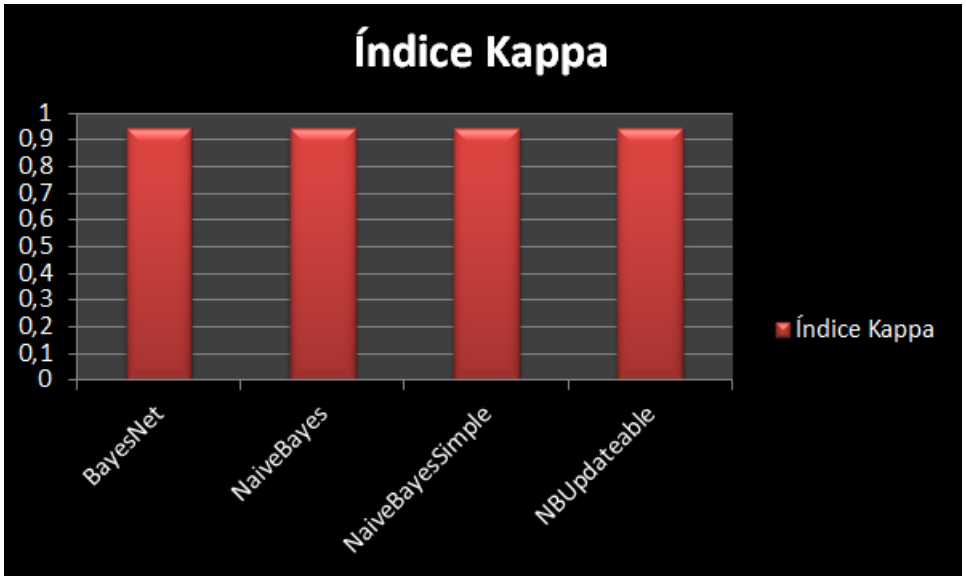

Figura 48: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de câncer de mama em relação ao critério do Índice Kappa.

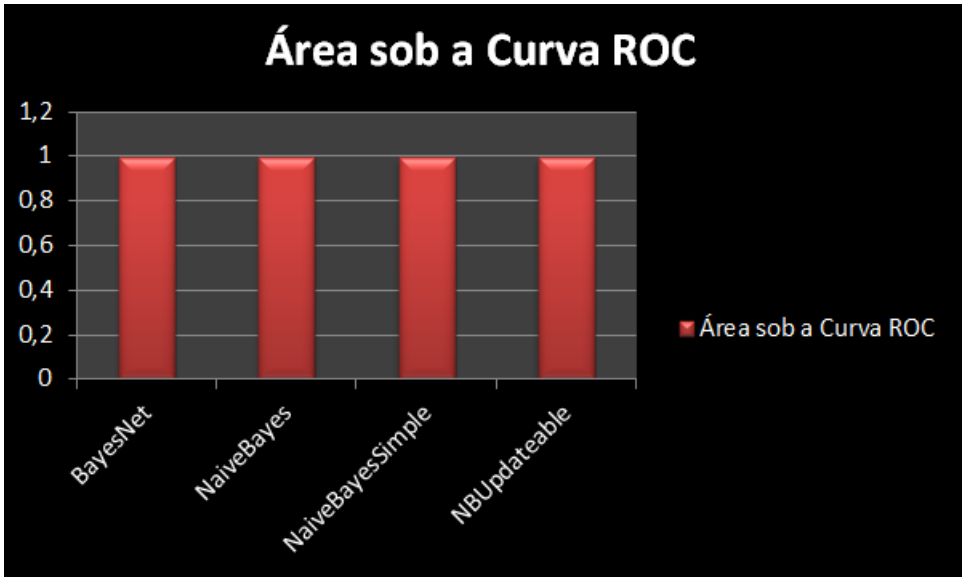

Figura 49: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de câncer de mama em relação ao critério da Área sob a Curva ROC.

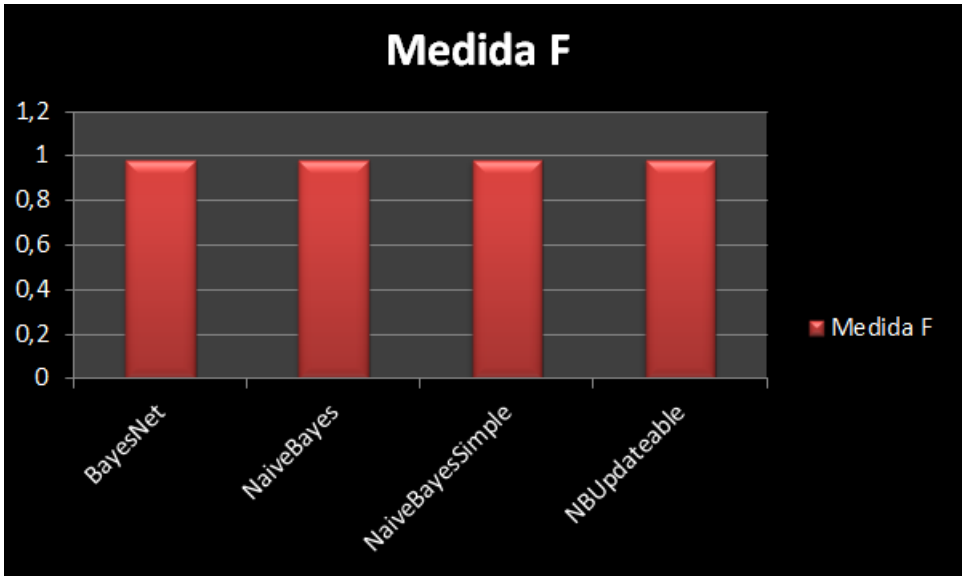

Figura 50: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de câncer de mama em relação ao critério da Medida F.

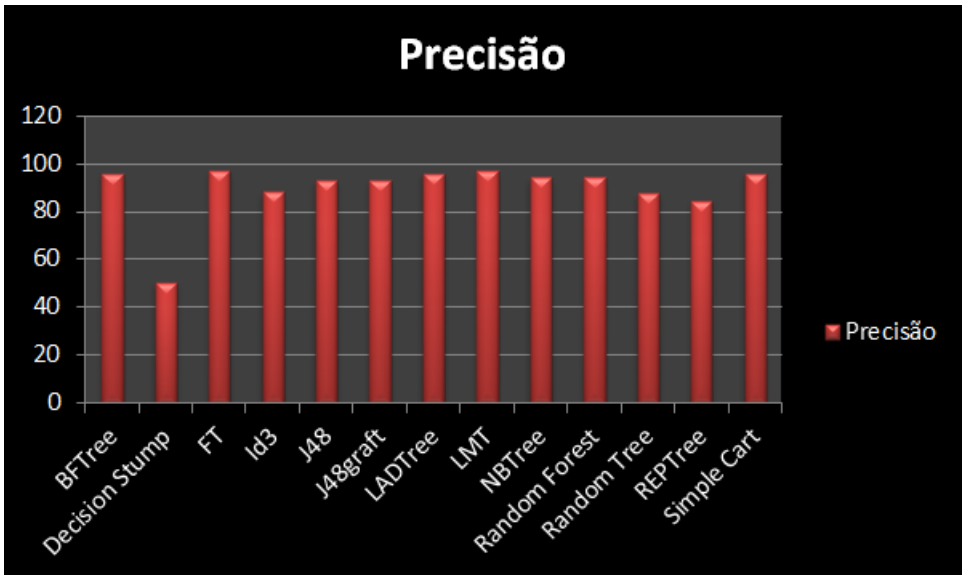

Figura 51: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de dermatologia em relação ao critério de Precisão.

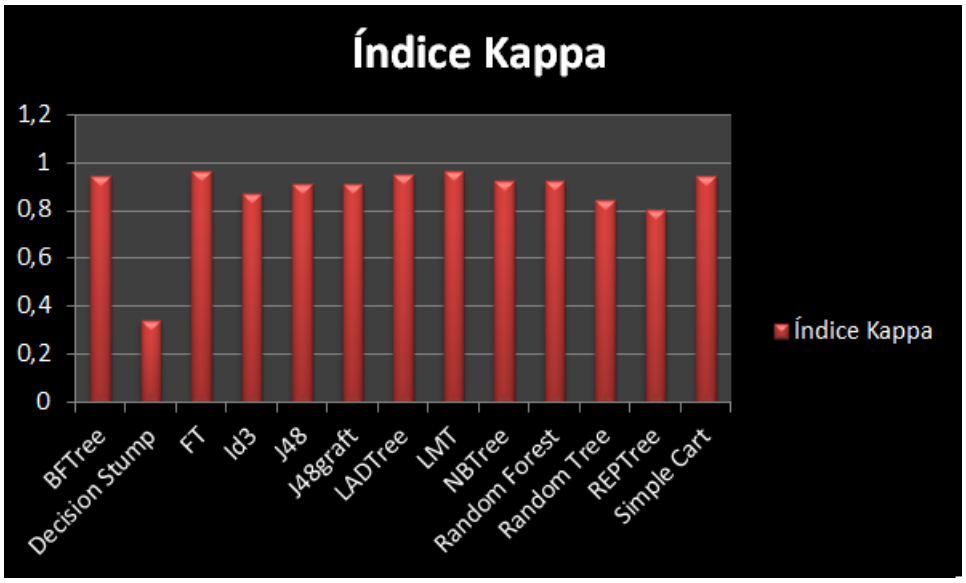

Figura 52: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de dermatologia em relação ao critério do Índice Kappa.

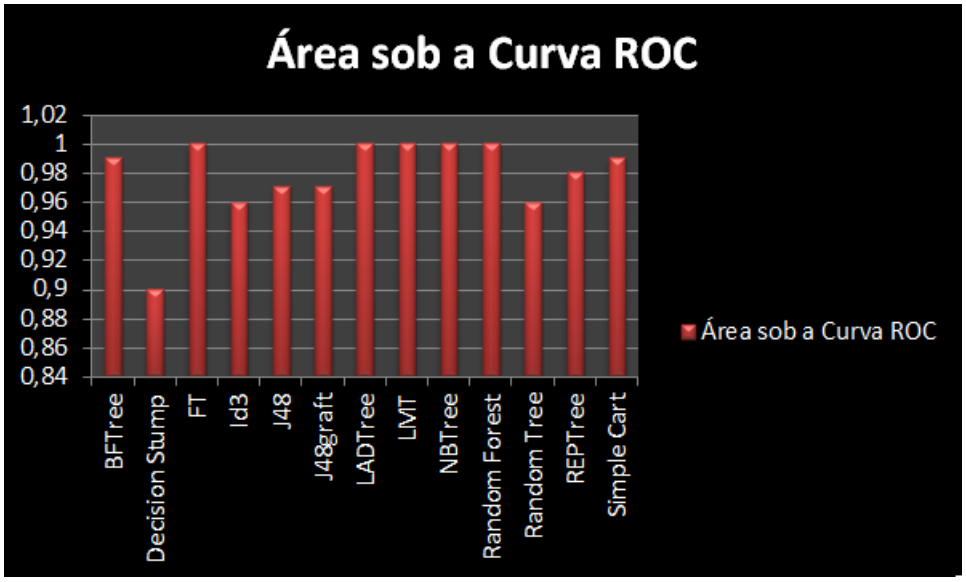

Figura 53: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de dermatologia em relação ao critério da Área sob a Curva ROC.

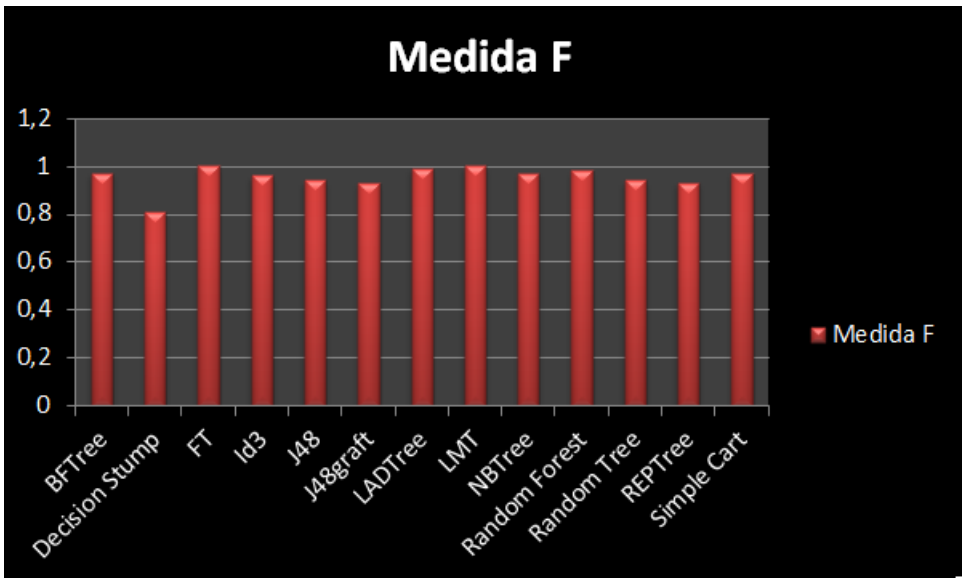

Figura 54: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de dermatologia em relação ao critério da Medida F.

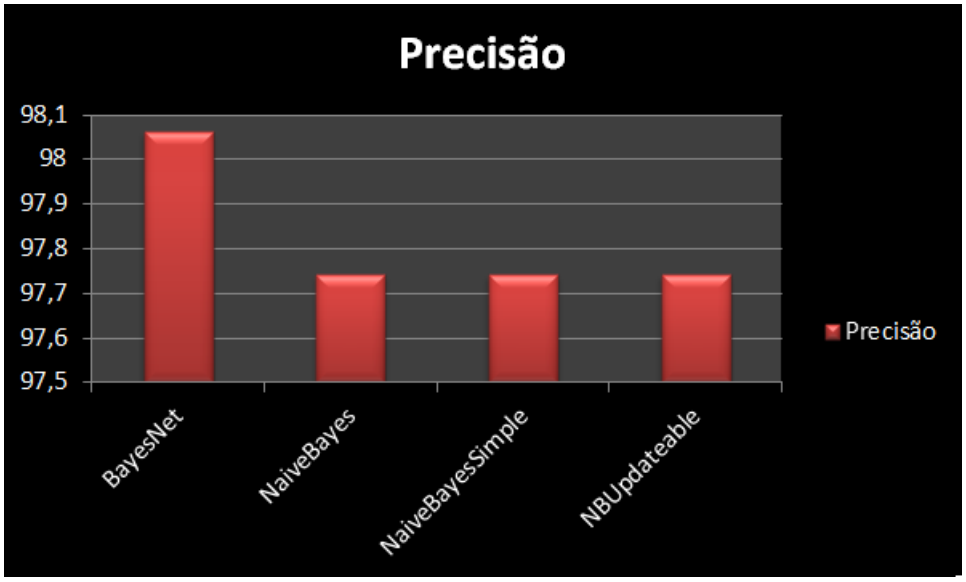

Figura 55: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de dermatologia em relação ao critério de Precisão.

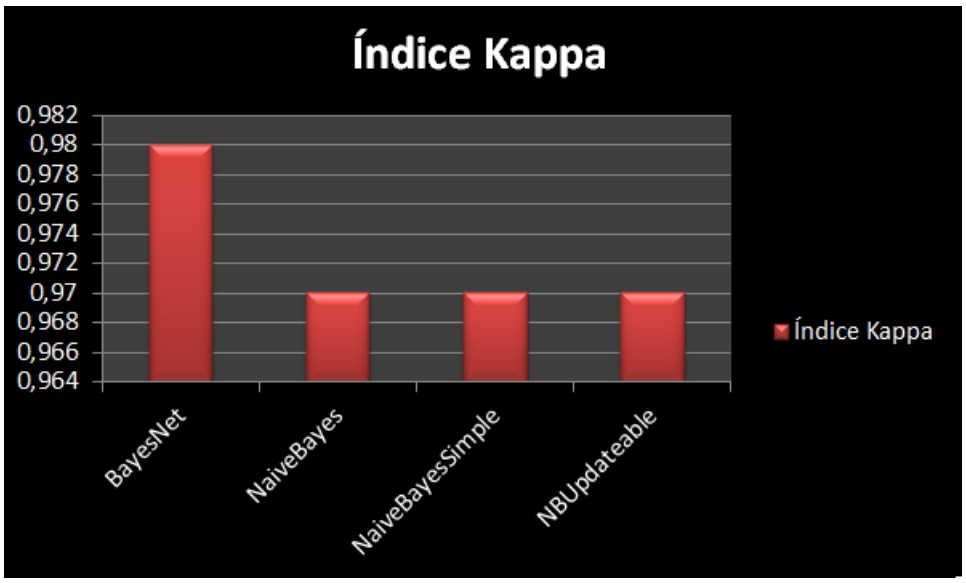

Figura 56: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de dermatologia em relação ao critério do Índice Kappa.

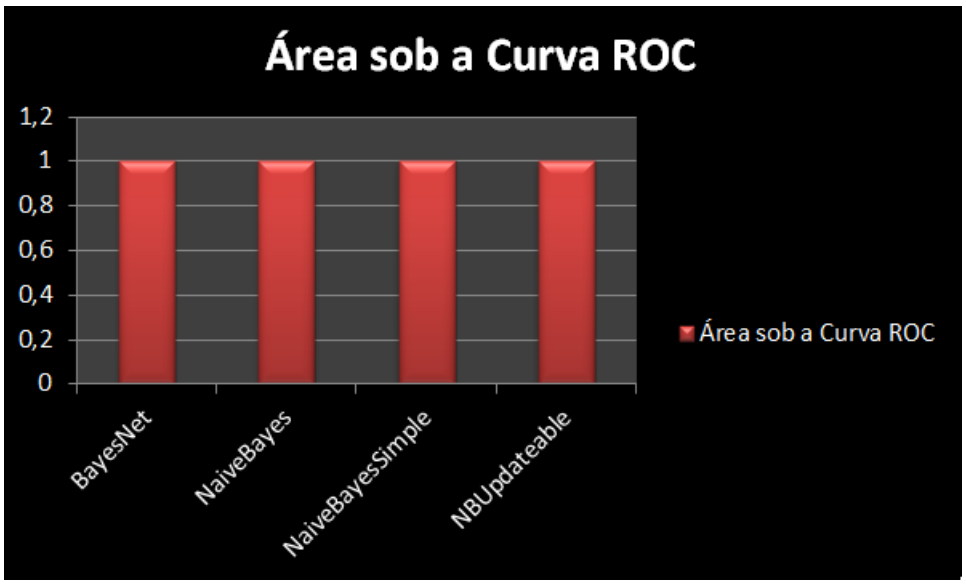

Figura 57: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de dermatologia em relação ao critério da Área sob a Curva ROC.

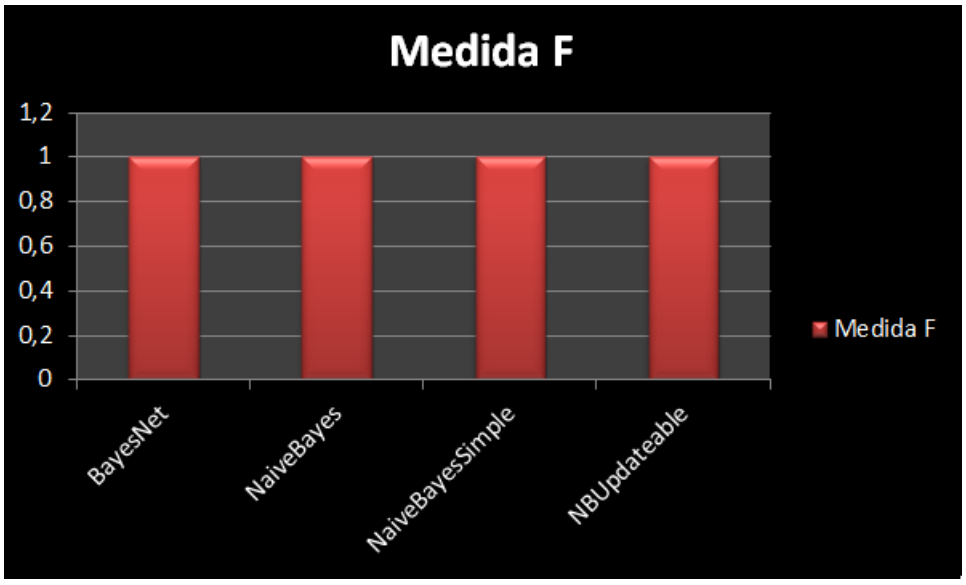

Figura 58: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de dermatologia em relação ao critério da Medida F.

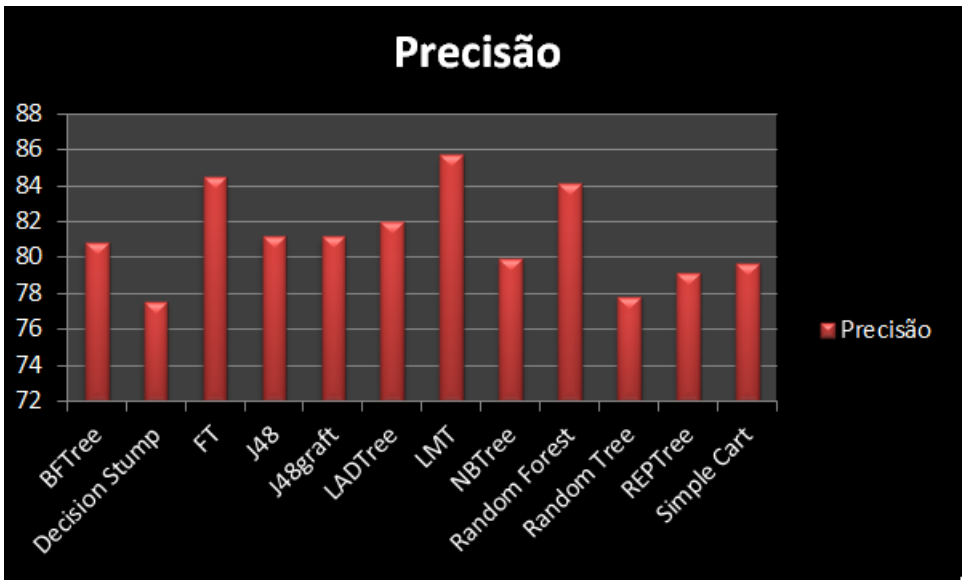

Figura 59: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de coluna vertebral em relação ao critério de Precisão.

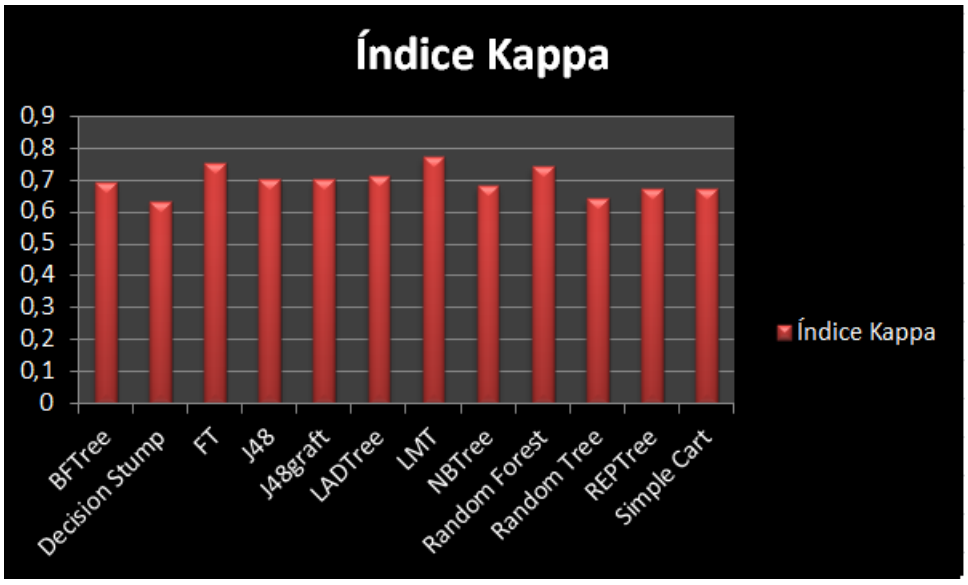

Figura 60: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de coluna vertebral em relação ao critério do Índice Kappa.

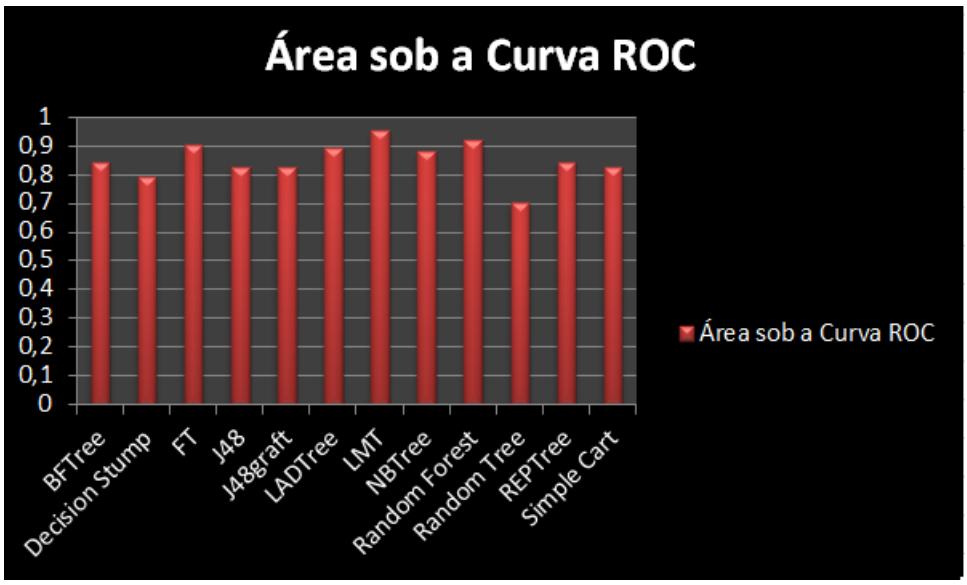

Figura 61: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de coluna vertebral em relação ao critério da Área sob a Curva ROC.

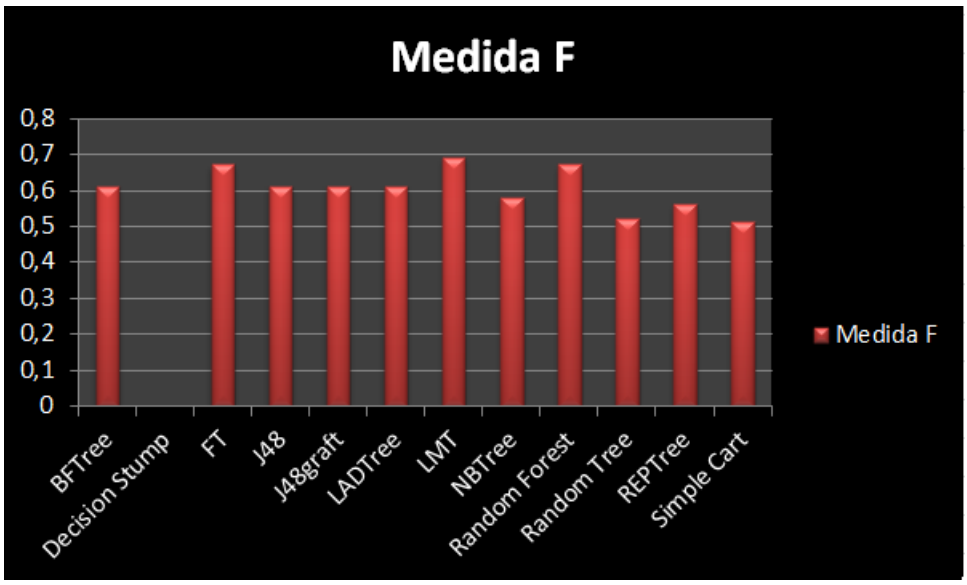

Figura 62: Gráfico dos resultados de desempenho dos algoritmos de árvore de decisão sobre a base de dados de coluna vertebral em relação ao critério da Medida F.

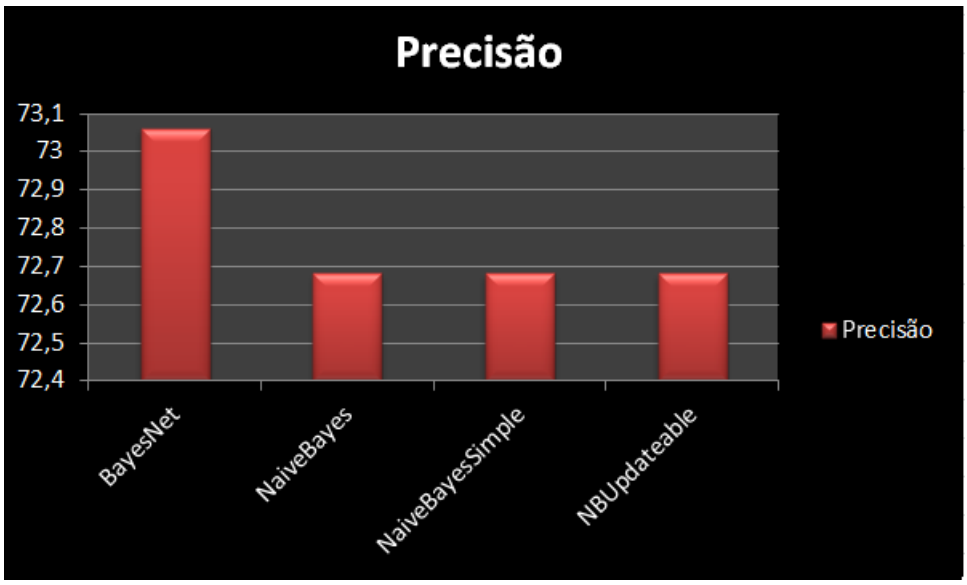

Figura 63: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de coluna vertebral em relação ao critério de Precisão.

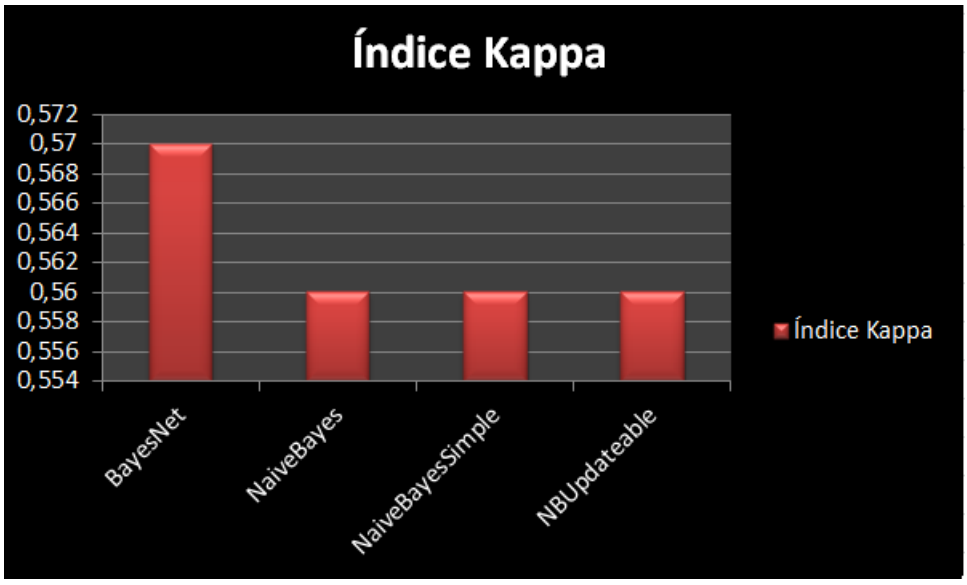

Figura 64: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de coluna vetebral em relação ao critério do Índice Kappa.

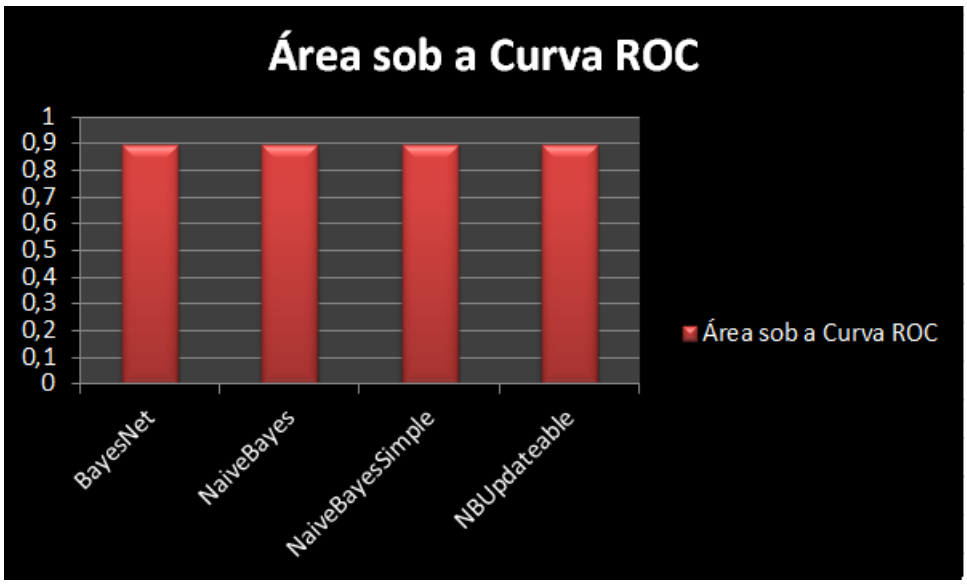

Figura 65: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de coluna vertebral em relação ao critério da Área sob a Curva ROC.

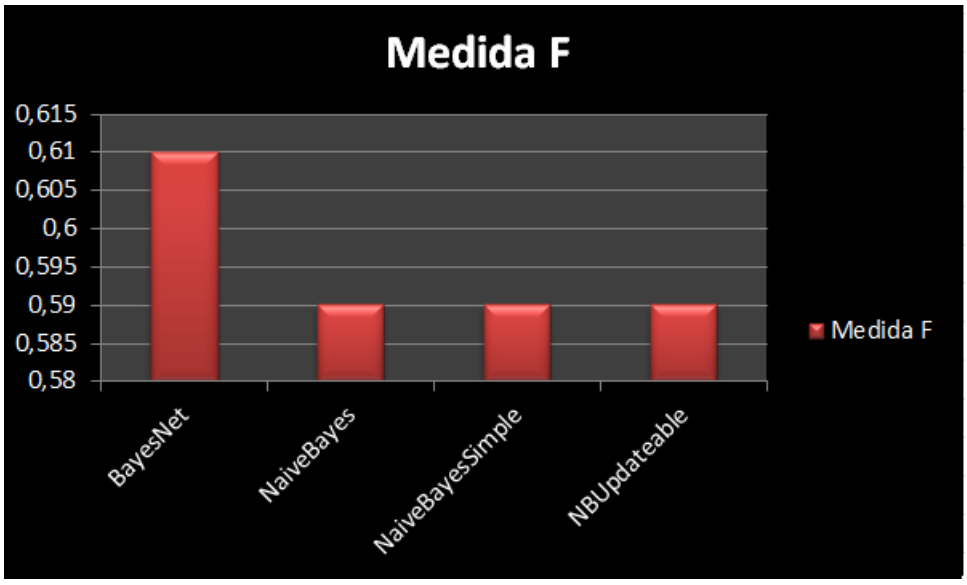

Figura 66: Gráfico dos resultados de desempenho dos algoritmos Bayesianos sobre a base de dados de coluna vertebral em relação ao critério da Medida F.

APÊNDICE B – Regras Extraídas dos Experimentos com Algoritmos de Associação

B.1 Regras de Associação Extraídas da Base de Dados de Câncer de Mama

1. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' 347 ==> Class=2 347 conf:(1)
2. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' Mitoses='(-inf-1.9]' 342 ==> Class=2 342 conf:(1)
3. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 327 ==> Class=2 327 conf:(1)
4. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 322 ==> Class=2 322 conf:(1)
5. UniformityofCellShape='(-inf-1.9]' BareNuclei='(-inf-1.9]' 319 ==> Class=2 319 conf:(1)
6. MarginalAdhesion='(-inf-1.9]' BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 316 ==> Class=2 316 conf:(1)
7. UniformityofCellSize='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 356 ==> Class=2 355 conf:(1)
8. UniformityofCellSize='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 350 ==> Class=2 349 conf:(1)
9. UniformityofCellShape='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 325 ==> Class=2 324 conf:(1)
10. UniformityofCellShape='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 319 ==> Class=2 318 conf:(1)
11. UniformityofCellSize='(-inf-1.9]' SingleEpithelialCellSize='(1.9-2.8]' 317 ==> Class=2 316 conf:(1)
12. BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 359 ==> Class=2 357 conf:(0.99)
13. UniformityofCellShape='(-inf-1.9]' 353 ==> Class=2 351 conf:(0.99)
14. UniformityofCellShape='(-inf-1.9]' Mitoses='(-inf-1.9]' 346 ==> Class=2 344 conf:(0.99)
15. UniformityofCellSize='(-inf-1.9]' MarginalAdhesion='(-inf-1.9]' 333 ==> Class=2 331 conf:(0.99)
16. UniformityofCellSize='(-inf-1.9]' UniformityofCellShape='(-inf-1.9]' 331 ==> Class=2 329 conf:(0.99)
17. SingleEpithelialCellSize='(1.9-2.8]' BareNuclei='(-inf-1.9]' 329 ==> Class=2 327 conf:(0.99)
18. UniformityofCellSize='(-inf-1.9]' MarginalAdhesion='(-inf-1.9]' Mitoses='(-inf-1.9]' 328 ==> Class=2 326 conf:(0.99)
19. UniformityofCellSize='(-inf-1.9]' UniformityofCellShape='(-inf-1.9]' Mitoses='(-inf-1.9]' 325 ==> Class=2 323 conf:(0.99)
20. SingleEpithelialCellSize='(1.9-2.8]' BareNuclei='(-inf-1.9]' Mitoses='(-inf-1.9]' 323 ==> Class=2 321 conf:(0.99)
21. UniformityofCellSize='(-inf-1.9]' Mitoses='(-inf-1.9]' 377 ==> Class=2 374 conf:(0.99)
22. BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 366 ==> Class=2 363 conf:(0.99)
23. MarginalAdhesion='(-inf-1.9]' BareNuclei='(-inf-1.9]' Mitoses='(-inf-1.9]' 339 ==> Class=2 336 conf:(0.99)
24. UniformityofCellSize='(-inf-1.9]' 384 ==> Class=2 380 conf:(0.99)
25. MarginalAdhesion='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 341 ==> Class=2 337 conf:(0.99)
26. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' 347 ==> Mitoses='(-inf-1.9]' 342 conf:(0.99)
27. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' Class=2 347 ==> Mitoses='(-inf-1.9]' 342 conf:(0.99)
28. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' 347 ==> Mitoses='(-inf-1.9]' Class=2 342 conf:(0.99)
29. MarginalAdhesion='(-inf-1.9]' BareNuclei='(-inf-1.9]' 346 ==> Class=2 341 conf:(0.99)
30. MarginalAdhesion='(-inf-1.9]' BareNuclei='(-inf-1.9]' Class=2 341 ==> Mitoses='(-inf-1.9]' 336 conf:(0.99)
31. UniformityofCellSize='(-inf-1.9]' MarginalAdhesion='(-inf-1.9]' 333 ==> Mitoses='(-inf-1.9]' 328 conf:(0.98)
32. UniformityofCellSize='(-inf-1.9]' MarginalAdhesion='(-inf-1.9]' Class=2 331 ==> Mitoses='(-inf-1.9]' 326 conf:(0.98)
33. SingleEpithelialCellSize='(1.9-2.8]' NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 328 ==> Class=2 323 conf:(0.98)
34. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 327 ==> Mitoses='(-inf-1.9]' 322 conf:(0.98)
35. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Class=2 327 ==> Mitoses='(-inf-1.9]' 322 conf:(0.98)
36. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 327 ==> Mitoses='(-inf-1.9]' Class=2 322 conf:(0.98)
37. UniformityofCellSize='(-inf-1.9]' Class=2 380 ==> Mitoses='(-inf-1.9]' 374 conf:(0.98)
38. MarginalAdhesion='(-inf-1.9]' Class=2 375 ==> Mitoses='(-inf-1.9]' 369 conf:(0.98)
39. BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Class=2 363 ==> Mitoses='(-inf-1.9]' 357 conf:(0.98)
40. UniformityofCellSize='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 356 ==> Mitoses='(-inf-1.9]' 350 conf:(0.98)
41. UniformityofCellSize='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Class=2 355 ==> Mitoses='(-inf-1.9]' 349 conf:(0.98)
42. MarginalAdhesion='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 349 ==> Class=2 343 conf:(0.98)
43. BareNuclei='(-inf-1.9]' Class=2 401 ==> Mitoses='(-inf-1.9]' 394 conf:(0.98)

44. MarginalAdhesion='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Class=2 343 ==> Mitoses='(-inf-1.9]' 337 conf:(0.98)
45. UniformityofCellSize='(-inf-1.9]' UniformityofCellShape='(-inf-1.9]' 331 ==> Mitoses='(-inf-1.9]' 325 conf:(0.98)
46. UniformityofCellSize='(-inf-1.9]' 384 ==> Mitoses='(-inf-1.9]' 377 conf:(0.98)
47. SingleEpithelialCellSize='(1.9-2.8]' BareNuclei='(-inf-1.9]' 329 ==> Mitoses='(-inf-1.9]' 323 conf:(0.98)
48. UniformityofCellSize='(-inf-1.9]' UniformityofCellShape='(-inf-1.9]' Class=2 329 ==> Mitoses='(-inf-1.9]' 323 conf:(0.98)
49. SingleEpithelialCellSize='(1.9-2.8]' BareNuclei='(-inf-1.9]' Class=2 327 ==> Mitoses='(-inf-1.9]' 321 conf:(0.98)
50. UniformityofCellShape='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 325 ==> Mitoses='(-inf-1.9]' 319 conf:(0.98)
51. UniformityofCellShape='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' Class=2 324 ==> Mitoses='(-inf-1.9]' 318 conf:(0.98)
52. BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 366 ==> Mitoses='(-inf-1.9]' 359 conf:(0.98)
53. UniformityofCellSize='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 356 ==> Mitoses='(-inf-1.9]' Class=2 349 conf:(0.98)
54. UniformityofCellShape='(-inf-1.9]' 353 ==> Mitoses='(-inf-1.9]' 346 conf:(0.98)
55. NormalNucleoli='(-inf-1.9]' Class=2 402 ==> Mitoses='(-inf-1.9]' 394 conf:(0.98)
56. UniformityofCellShape='(-inf-1.9]' Class=2 351 ==> Mitoses='(-inf-1.9]' 344 conf:(0.98)
57. MarginalAdhesion='(-inf-1.9]' BareNuclei='(-inf-1.9]' 346 ==> Mitoses='(-inf-1.9]' 339 conf:(0.98)
58. SingleEpithelialCellSize='(1.9-2.8]' NormalNucleoli='(-inf-1.9]' 337 ==> Class=2 330 conf:(0.98)
59. UniformityofCellSize='(-inf-1.9]' MarginalAdhesion='(-inf-1.9]' 333 ==> Mitoses='(-inf-1.9]' Class=2 326 conf:(0.98)
60. SingleEpithelialCellSize='(1.9-2.8]' NormalNucleoli='(-inf-1.9]' Class=2 330 ==> Mitoses='(-inf-1.9]' 323 conf:(0.98)
61. UniformityofCellShape='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 325 ==> Mitoses='(-inf-1.9]' Class=2 318 conf:(0.98)
62. MarginalAdhesion='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 349 ==> Mitoses='(-inf-1.9]' 341 conf:(0.98)
63. UniformityofCellSize='(-inf-1.9]' UniformityofCellShape='(-inf-1.9]' 331 ==> Mitoses='(-inf-1.9]' Class=2 323 conf:(0.98)
64. SingleEpithelialCellSize='(1.9-2.8]' BareNuclei='(-inf-1.9]' 329 ==> Mitoses='(-inf-1.9]' Class=2 321 conf:(0.98)
65. BareNuclei='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 366 ==> Mitoses='(-inf-1.9]' Class=2 357 conf:(0.98)
66. SingleEpithelialCellSize='(1.9-2.8]' Class=2 363 ==> Mitoses='(-inf-1.9]' 354 conf:(0.98)
67. UniformityofCellShape='(-inf-1.9]' 353 ==> Mitoses='(-inf-1.9]' Class=2 344 conf:(0.97)
68. UniformityofCellSize='(-inf-1.9]' 384 ==> Mitoses='(-inf-1.9]' Class=2 374 conf:(0.97)
69. SingleEpithelialCellSize='(1.9-2.8]' NormalNucleoli='(-inf-1.9]' 337 ==> Mitoses='(-inf-1.9]' 328 conf:(0.97)
70. BareNuclei='(-inf-1.9]' Mitoses='(-inf-1.9]' 405 ==> Class=2 394 conf:(0.97)
71. Class=2 458 ==> Mitoses='(-inf-1.9]' 445 conf:(0.97)
72. MarginalAdhesion='(-inf-1.9]' BareNuclei='(-inf-1.9]' 346 ==> Mitoses='(-inf-1.9]' Class=2 336 conf:(0.97)
73. BareNuclei='(-inf-1.9]' 418 ==> Mitoses='(-inf-1.9]' 405 conf:(0.97)
74. MarginalAdhesion='(-inf-1.9]' NormalNucleoli='(-inf-1.9]' 349 ==> Mitoses='(-inf-1.9]' Class=2 337 conf:(0.97)
75. SingleEpithelialCellSize='(1.9-2.8]' 386 ==> Mitoses='(-inf-1.9]' 372 conf:(0.96)
76. NormalNucleoli='(-inf-1.9]' 443 ==> Mitoses='(-inf-1.9]' 426 conf:(0.96)
77. BareNuclei='(-inf-1.9]' 418 ==> Class=2 401 conf:(0.96)
78. SingleEpithelialCellSize='(1.9-2.8]' NormalNucleoli='(-inf-1.9]' 337 ==> Mitoses='(-inf-1.9]' Class=2 323 conf:(0.96)
79. MarginalAdhesion='(-inf-1.9]' 407 ==> Mitoses='(-inf-1.9]' 388 conf:(0.95)
80. SingleEpithelialCellSize='(1.9-2.8]' Mitoses='(-inf-1.9]' 372 ==> Class=2 354 conf:(0.95)
81. MarginalAdhesion='(-inf-1.9]' Mitoses='(-inf-1.9]' 388 ==> Class=2 369 conf:(0.95)
82. BareNuclei='(-inf-1.9]' 418 ==> Mitoses='(-inf-1.9]' Class=2 394 conf:(0.94)
83. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' 347 ==> NormalNucleoli='(-inf-1.9]' 327 conf:(0.94)
84. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' Class=2 347 ==> NormalNucleoli='(-inf-1.9]' 327 conf:(0.94)
85. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' 347 ==> NormalNucleoli='(-inf-1.9]' Class=2 327 conf:(0.94)
86. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' Mitoses='(-inf-1.9]' 342 ==> NormalNucleoli='(-inf-1.9]' 322 conf:(0.94)
87. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' Mitoses='(-inf-1.9]' Class=2 342 ==> NormalNucleoli='(-inf-1.9]' 322 conf:(0.94)
88. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' Mitoses='(-inf-1.9]' 342 ==> NormalNucleoli='(-inf-1.9]' Class=2 322 conf:(0.94)
89. SingleEpithelialCellSize='(1.9-2.8]' 386 ==> Class=2 363 conf:(0.94)
90. UniformityofCellShape='(-inf-1.9]' Mitoses='(-inf-1.9]' 346 ==> UniformityofCellSize='(-inf-1.9]' 325 conf:(0.94)
91. UniformityofCellShape='(-inf-1.9]' Mitoses='(-inf-1.9]' Class=2 344 ==> UniformityofCellSize='(-inf-1.9]' 323 conf:(0.94)
92. UniformityofCellShape='(-inf-1.9]' 353 ==> UniformityofCellSize='(-inf-1.9]' 331 conf:(0.94)
93. UniformityofCellShape='(-inf-1.9]' Class=2 351 ==> UniformityofCellSize='(-inf-1.9]' 329 conf:(0.94)
94. UniformityofCellSize='(-inf-1.9]' Class=2 380 ==> NormalNucleoli='(-inf-1.9]' 355 conf:(0.93)
95. UniformityofCellShape='(-inf-1.9]' Mitoses='(-inf-1.9]' 346 ==> UniformityofCellSize='(-inf-1.9]' Class=2 323 conf:(0.93)
96. UniformityofCellSize='(-inf-1.9]' Mitoses='(-inf-1.9]' Class=2 374 ==> NormalNucleoli='(-inf-1.9]' 349 conf:(0.93)
97. UniformityofCellShape='(-inf-1.9]' 353 ==> UniformityofCellSize='(-inf-1.9]' Class=2 329 conf:(0.93)
98. UniformityofCellSize='(-inf-1.9]' Mitoses='(-inf-1.9]' 377 ==> NormalNucleoli='(-inf-1.9]' 350 conf:(0.93)
99. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' 347 ==> NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 322 conf:(0.93)
100. UniformityofCellSize='(-inf-1.9]' BareNuclei='(-inf-1.9]' Class=2 347 ==> NormalNucleoli='(-inf-1.9]' Mitoses='(-inf-1.9]' 322 conf:(0.93)

B.2 Regras de Associação Extraídas da Base de Dados de Dermatologia

1. follicular='(-inf-0.75]' horn='(-inf-0.75]' 330 ==> perifollicular='(-inf-0.75]' 330 conf:(1)
2. vacuolisation='(-inf-0.75]' 294 ==> melanin='(-inf-0.75]' 294 conf:(1)
3. oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' 294 conf:(1)
4. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' 294 conf:(1)
5. polygonal='(-inf-0.75]' oral='(-inf-0.75]' 294 ==> melanin='(-inf-0.75]' 294 conf:(1)
6. follicular='(-inf-0.75]' family='(-inf-0.25]' horn='(-inf-0.75]' 294 ==> perifollicular='(-inf-0.75]' 294 conf:(1)

```
 7. oral='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' 293    conf:(1)
 8. polygonal='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' 293    conf:(1)
 9. oral='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' 293    conf:(1)
10. polygonal='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' 293    conf:(1)
11. oral='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' 293    conf:(1)
12. polygonal='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' 293    conf:(1)
13. polygonal='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
14. polygonal='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
15. melanin='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' 293    conf:(1)
16. polygonal='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
17. oral='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
18. oral='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
19. melanin='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' 293    conf:(1)
20. oral='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
21. focal='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
22. oral='(-inf-0.75]' melanin='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' 293    conf:(1)
23. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' 293    conf:(1)
24. polygonal='(-inf-0.75]' oral='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
25. oral='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
26. polygonal='(-inf-0.75]' focal='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
27. oral='(-inf-0.75]' melanin='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' 293    conf:(1)
28. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' 293    conf:(1)
29. polygonal='(-inf-0.75]' oral='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
30. oral='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
31. polygonal='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
32. oral='(-inf-0.75]' melanin='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' 293    conf:(1)
33. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' 293    conf:(1)
34. polygonal='(-inf-0.75]' oral='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> melanin='(-inf-0.75]' 293    conf:(1)
35. melanin='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' oral='(-inf-0.75]' 293    conf:(1)
36. oral='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
37. polygonal='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 ==> oral='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
38. horn='(-inf-0.75]' 344 ==> perifollicular='(-inf-0.75]' 343    conf:(1)
39. follicular='(-inf-0.75]' 333 ==> perifollicular='(-inf-0.75]' 332    conf:(1)
40. family='(-inf-0.25]' horn='(-inf-0.75]' 308 ==> perifollicular='(-inf-0.75]' 307    conf:(1)
41. eosinophils='(-inf-0.5]' horn='(-inf-0.75]' 302 ==> perifollicular='(-inf-0.75]' 301    conf:(1)
42. follicular='(-inf-0.75]' family='(-inf-0.25]' 297 ==> perifollicular='(-inf-0.75]' 296    conf:(1)
43. focal='(-inf-0.75]' 295 ==> melanin='(-inf-0.75]' 294    conf:(1)
44. vacuolisation='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' 293    conf:(1)
45. sawtooth='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' 293    conf:(1)
46. vacuolisation='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' 293    conf:(1)
47. sawtooth='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' 293    conf:(1)
48. sawtooth='(-inf-0.75]' 294 ==> melanin='(-inf-0.75]' 293    conf:(1)
49. vacuolisation='(-inf-0.75]' 294 ==> focal='(-inf-0.75]' 293    conf:(1)
50. polygonal='(-inf-0.75]' oral='(-inf-0.75]' 294 ==> focal='(-inf-0.75]' 293    conf:(1)
51. polygonal='(-inf-0.75]' oral='(-inf-0.75]' 294 ==> vacuolisation='(-inf-0.75]' 293    conf:(1)
52. vacuolisation='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' oral='(-inf-0.75]' 293    conf:(1)
53. polygonal='(-inf-0.75]' oral='(-inf-0.75]' 294 ==> sawtooth='(-inf-0.75]' 293    conf:(1)
54. sawtooth='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' oral='(-inf-0.75]' 293    conf:(1)
55. melanin='(-inf-0.75]' focal='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' 293    conf:(1)
56. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> focal='(-inf-0.75]' 293    conf:(1)
57. melanin='(-inf-0.75]' vacuolisation='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' 293    conf:(1)
58. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> vacuolisation='(-inf-0.75]' 293    conf:(1)
59. vacuolisation='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
60. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> sawtooth='(-inf-0.75]' 293    conf:(1)
61. sawtooth='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
62. melanin='(-inf-0.75]' focal='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' 293    conf:(1)
63. oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> focal='(-inf-0.75]' 293    conf:(1)
64. melanin='(-inf-0.75]' vacuolisation='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' 293    conf:(1)
65. oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> vacuolisation='(-inf-0.75]' 293    conf:(1)
66. vacuolisation='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
67. oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> sawtooth='(-inf-0.75]' 293    conf:(1)
68. sawtooth='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' melanin='(-inf-0.75]' 293    conf:(1)
69. melanin='(-inf-0.75]' vacuolisation='(-inf-0.75]' 294 ==> focal='(-inf-0.75]' 293    conf:(1)
70. melanin='(-inf-0.75]' focal='(-inf-0.75]' 294 ==> vacuolisation='(-inf-0.75]' 293    conf:(1)
71. vacuolisation='(-inf-0.75]' 294 ==> melanin='(-inf-0.75]' focal='(-inf-0.75]' 293    conf:(1)
72. polygonal='(-inf-0.75]' oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> focal='(-inf-0.75]' 293    conf:(1)
73. melanin='(-inf-0.75]' focal='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' oral='(-inf-0.75]' 293    conf:(1)
74. oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' focal='(-inf-0.75]' 293    conf:(1)
75. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' focal='(-inf-0.75]' 293    conf:(1)
76. polygonal='(-inf-0.75]' oral='(-inf-0.75]' 294 ==> melanin='(-inf-0.75]' focal='(-inf-0.75]' 293    conf:(1)
77. polygonal='(-inf-0.75]' oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> vacuolisation='(-inf-0.75]' 293    conf:(1)
78. melanin='(-inf-0.75]' vacuolisation='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' oral='(-inf-0.75]' 293    conf:(1)
79. oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293    conf:(1)
```

80. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 conf:(1)
81. polygonal='(-inf-0.75]' oral='(-inf-0.75]' 294 ==> melanin='(-inf-0.75]' vacuolisation='(-inf-0.75]' 293 conf:(1)
82. vacuolisation='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' oral='(-inf-0.75]' melanin='(-inf-0.75]' 293 conf:(1)
83. polygonal='(-inf-0.75]' oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> sawtooth='(-inf-0.75]' 293 conf:(1)
84. oral='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 conf:(1)
85. polygonal='(-inf-0.75]' melanin='(-inf-0.75]' 294 ==> oral='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 conf:(1)
86. polygonal='(-inf-0.75]' oral='(-inf-0.75]' 294 ==> melanin='(-inf-0.75]' sawtooth='(-inf-0.75]' 293 conf:(1)
87. sawtooth='(-inf-0.75]' 294 ==> polygonal='(-inf-0.75]' oral='(-inf-0.75]' melanin='(-inf-0.75]' 293 conf:(1)
88. perifollicular='(-inf-0.75]' 345 ==> horn='(-inf-0.75]' 343 conf:(0.99)
89. follicular='(-inf-0.75]' perifollicular='(-inf-0.75]' 332 ==> horn='(-inf-0.75]' 330 conf:(0.99)
90. family='(-inf-0.25]' perifollicular='(-inf-0.75]' 309 ==> horn='(-inf-0.75]' 307 conf:(0.99)
91. eosinophils='(-inf-0.5]' perifollicular='(-inf-0.75]' 303 ==> horn='(-inf-0.75]' 301 conf:(0.99)
92. melanin='(-inf-0.75]' 296 ==> polygonal='(-inf-0.75]' 294 conf:(0.99)
93. melanin='(-inf-0.75]' 296 ==> oral='(-inf-0.75]' 294 conf:(0.99)
94. melanin='(-inf-0.75]' 296 ==> focal='(-inf-0.75]' 294 conf:(0.99)
95. melanin='(-inf-0.75]' 296 ==> vacuolisation='(-inf-0.75]' 294 conf:(0.99)
96. melanin='(-inf-0.75]' 296 ==> polygonal='(-inf-0.75]' oral='(-inf-0.75]' 294 conf:(0.99)
97. follicular='(-inf-0.75]' family='(-inf-0.25]' perifollicular='(-inf-0.75]' 296 ==> horn='(-inf-0.75]' 294 conf:(0.99)
98. focal='(-inf-0.75]' 295 ==> polygonal='(-inf-0.75]' 293 conf:(0.99)
99. focal='(-inf-0.75]' 295 ==> oral='(-inf-0.75]' 293 conf:(0.99)
100. focal='(-inf-0.75]' 295 ==> vacuolisation='(-inf-0.75]' 293 conf:(0.99)

B.3 Regras de Associação Extraídas da Base de Dados de Coluna Vertebral

1. class=Normal 100 ==> degree_spondylolisthesis='(-inf-74.862073]' 100 conf:(1)
2. pelvic_incidence='(-inf-46.885145]' 81 ==> degree_spondylolisthesis='(-inf-74.862073]' 81 conf:(1)
3. pelvic_tilt='(4.642414-15.839776]' sacral_slope='(34.979458-56.591985]' 75 ==> degree_spondylolisthesis='(-inf-74.862073]' 75 conf:(1)
4. lumbar_lordosis_angle='(-inf-36.348477]' 74 ==> degree_spondylolisthesis='(-inf-74.862073]' 74 conf:(1)
5. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' 64 ==> degree_spondylolisthesis='(-inf-74.862073]' 64 conf:(1)
6. pelvic_incidence='(-inf-46.885145]' sacral_slope='(-inf-34.979458]' 62 ==> degree_spondylolisthesis='(-inf-74.862073]' 62 conf:(1)
7. pelvic_incidence='(46.885145-67.622369]' lumbar_lordosis_angle='(36.348477-58.696954]' sacral_slope='(34.979458-56.591985]' 61 ==>
degree_spondylolisthesis='(-inf-74.862073]' 61 conf:(1)
8. class=Hernia 60 ==> degree_spondylolisthesis='(-inf-74.862073]' 60 conf:(1)
9. sacral_slope='(34.979458-56.591985]' class=Normal 59 ==> degree_spondylolisthesis='(-inf-74.862073]' 59 conf:(1)
10. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' sacral_slope='(34.979458-56.591985]' 59 ==>
degree_spondylolisthesis='(-inf-74.862073]' 59 conf:(1)
11. lumbar_lordosis_angle='(36.348477-58.696954]' class=Normal 58 ==> degree_spondylolisthesis='(-inf-74.862073]' 58 conf:(1)
12. sacral_slope='(-inf-34.979458]' pelvic_radius='(107.277961-125.875654]' 57 ==> degree_spondylolisthesis='(-inf-74.862073]' 57 conf:(1)
13. lumbar_lordosis_angle='(-inf-36.348477]' sacral_slope='(-inf-34.979458]' 56 ==> degree_spondylolisthesis='(-inf-74.862073]' 56 conf:(1)
14. pelvic_tilt='(4.642414-15.839776]' class=Normal 55 ==> degree_spondylolisthesis='(-inf-74.862073]' 55 conf:(1)
15. pelvic_radius='(107.277961-125.875654]' class=Normal 53 ==> degree_spondylolisthesis='(-inf-74.862073]' 53 conf:(1)
16. pelvic_incidence='(46.885145-67.622369]' class=Normal 51 ==> degree_spondylolisthesis='(-inf-74.862073]' 51 conf:(1)
17. lumbar_lordosis_angle='(36.348477-58.696954]' sacral_slope='(34.979458-56.591985]' pelvic_radius='(107.277961-125.875654]' 50 ==>
degree_spondylolisthesis='(-inf-74.862073]' 50 conf:(1)
18. pelvic_incidence='(-inf-46.885145]' pelvic_tilt='(4.642414-15.839776]' 49 ==> degree_spondylolisthesis='(-inf-74.862073]' 49 conf:(1)
19. pelvic_tilt='(4.642414-15.839776]' lumbar_lordosis_angle='(36.348477-58.696954]' sacral_slope='(34.979458-56.591985]' 48 ==>
degree_spondylolisthesis='(-inf-74.862073]' 48 conf:(1)
20. pelvic_incidence='(-inf-46.885145]' lumbar_lordosis_angle='(-inf-36.348477]' 47 ==> degree_spondylolisthesis='(-inf-74.862073]' 47 conf:(1)
21. lumbar_lordosis_angle='(-inf-36.348477]' pelvic_radius='(107.277961-125.875654]' 47 ==> degree_spondylolisthesis='(-inf-74.862073]' 47 conf:(1)
22. pelvic_radius='(107.277961-125.875654]' class=Hernia 46 ==> degree_spondylolisthesis='(-inf-74.862073]' 46 conf:(1)
23. sacral_slope='(-inf-34.979458]' class=Hernia 45 ==> degree_spondylolisthesis='(-inf-74.862073]' 45 conf:(1)
24. pelvic_incidence='(46.885145-67.622369]' lumbar_lordosis_angle='(36.348477-58.696954]' pelvic_radius='(107.277961-125.875654]' 44 ==>
degree_spondylolisthesis='(-inf-74.862073]' 44 conf:(1)
25. pelvic_incidence='(46.885145-67.622369]' sacral_slope='(34.979458-56.591985]' class=Normal 44 ==>
degree_spondylolisthesis='(-inf-74.862073]' 44 conf:(1)
26. pelvic_incidence='(-inf-46.885145]' pelvic_radius='(107.277961-125.875654]' 42 ==> degree_spondylolisthesis='(-inf-74.862073]' 42 conf:(1)
27. pelvic_tilt='(15.839776-27.037139]' lumbar_lordosis_angle='(36.348477-58.696954]' 42 ==> degree_spondylolisthesis='(-inf-74.862073]' 42 conf:(1)
28. pelvic_incidence='(-inf-46.885145]' lumbar_lordosis_angle='(-inf-36.348477]' sacral_slope='(-inf-34.979458]' 42 ==>
degree_spondylolisthesis='(-inf-74.862073]' 42 conf:(1)
29. pelvic_radius='(125.875654-144.473347]' class=Normal 41 ==> degree_spondylolisthesis='(-inf-74.862073]' 41 conf:(1)
30. pelvic_tilt='(4.642414-15.839776]' sacral_slope='(34.979458-56.591985]' pelvic_radius='(107.277961-125.875654]' 41 ==>
degree_spondylolisthesis='(-inf-74.862073]' 41 conf:(1)
31. lumbar_lordosis_angle='(36.348477-58.696954]' sacral_slope='(34.979458-56.591985]' class=Normal 41 ==>
degree_spondylolisthesis='(-inf-74.862073]' 41 conf:(1)
32. pelvic_incidence='(-inf-46.885145]' class=Normal 40 ==> degree_spondylolisthesis='(-inf-74.862073]' 40 conf:(1)
33. pelvic_tilt='(4.642414-15.839776]' sacral_slope='(-inf-34.979458]' 40 ==> degree_spondylolisthesis='(-inf-74.862073]' 40 conf:(1)
34. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' lumbar_lordosis_angle='(36.348477-58.696954]' 39 ==>

degree_spondylolisthesis='(-inf-74.862073]' 39 conf:(1)
35. pelvic_tilt='(4.642414-15.839776]' lumbar_lordosis_angle='(-inf-36.348477]' 38 ==> degree_spondylolisthesis='(-inf-74.862073]' 38 conf:(1)
36. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' pelvic_radius='(107.277961-125.875654]' 38 ==>
degree_spondylolisthesis='(-inf-74.862073]' 38 conf:(1)
37. lumbar_lordosis_angle='(-inf-36.348477]' class=Hernia 37 ==> degree_spondylolisthesis='(-inf-74.862073]' 37 conf:(1)
38. sacral_slope='(-inf-34.979458]' class=Normal 37 ==> degree_spondylolisthesis='(-inf-74.862073]' 37 conf:(1)
39. pelvic_incidence='(-inf-46.885145]' pelvic_tilt='(4.642414-15.839776]' sacral_slope='(-inf-34.979458]' 37 ==>
degree_spondylolisthesis='(-inf-74.862073]' 37 conf:(1)
40. lumbar_lordosis_angle='(-inf-36.348477]' sacral_slope='(-inf-34.979458]' pelvic_radius='(107.277961-125.875654]' 37 ==>
degree_spondylolisthesis='(-inf-74.862073]' 37 conf:(1)
41. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' lumbar_lordosis_angle='(36.348477-58.696954]'
sacral_slope='(34.979458-56.591985]' 37 ==> degree_spondylolisthesis='(-inf-74.862073]' 37 conf:(1)
42. sacral_slope='(-inf-34.979458]' pelvic_radius='(107.277961-125.875654]' class=Hernia 36 ==>
degree_spondylolisthesis='(-inf-74.862073]' 36 conf:(1)
43. pelvic_incidence='(46.885145-67.622369]' lumbar_lordosis_angle='(36.348477-58.696954]' sacral_slope='(34.979458-56.591985]'
pelvic_radius='(107.277961-125.875654]' 36 ==> degree_spondylolisthesis='(-inf-74.862073]' 36 conf:(1)
44. pelvic_tilt='(15.839776-27.037139]' sacral_slope='(-inf-34.979458]' 35 ==>
degree_spondylolisthesis='(-inf-74.862073]' 35 conf:(1)
45. pelvic_tilt='(4.642414-15.839776]' lumbar_lordosis_angle='(36.348477-58.696954]' class=Normal 35 ==>
degree_spondylolisthesis='(-inf-74.862073]' 35 conf:(1)
46. pelvic_tilt='(4.642414-15.839776]' sacral_slope='(34.979458-56.591985]' class=Spondylolisthesis 35 ==>
degree_spondylolisthesis='(-inf-74.862073]' 35 conf:(1)
47. sacral_slope='(34.979458-56.591985]' pelvic_radius='(107.277961-125.875654]' class=Normal 35 ==>
degree_spondylolisthesis='(-inf-74.862073]' 35 conf:(1)
48. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' sacral_slope='(34.979458-56.591985]'
pelvic_radius='(107.277961-125.875654]' 35 ==> degree_spondylolisthesis='(-inf-74.862073]' 35 conf:(1)
49. pelvic_incidence='(-inf-46.885145]' sacral_slope='(-inf-34.979458]' pelvic_radius='(107.277961-125.875654]' 34 ==>
degree_spondylolisthesis='(-inf-74.862073]' 34 conf:(1)
50. pelvic_tilt='(4.642414-15.839776]' sacral_slope='(34.979458-56.591985]' class=Normal 34 ==> degree_spondylolisthesis='(-inf-74.862073]' 34 conf:(1)
51. pelvic_incidence='(-inf-46.885145]' lumbar_lordosis_angle='(36.348477-58.696954]' 33 ==> degree_spondylolisthesis='(-inf-74.862073]' 33 conf:(1)
52. pelvic_tilt='(15.839776-27.037139]' class=Normal 33 ==> degree_spondylolisthesis='(-inf-74.862073]' 33 conf:(1)
53. lumbar_lordosis_angle='(-inf-36.348477]' class=Normal 33 ==> degree_spondylolisthesis='(-inf-74.862073]' 33 conf:(1)
54. pelvic_incidence='(46.885145-67.622369]' lumbar_lordosis_angle='(36.348477-58.696954]' class=Normal 33 ==>
degree_spondylolisthesis='(-inf-74.862073]' 33 conf:(1)
55. pelvic_incidence='(46.885145-67.622369]' pelvic_radius='(107.277961-125.875654]' class=Normal 33 ==>
degree_spondylolisthesis='(-inf-74.862073]' 33 conf:(1)
56. pelvic_incidence='(-inf-46.885145]' class=Hernia 32 ==> degree_spondylolisthesis='(-inf-74.862073]' 32 conf:(1)
57. lumbar_lordosis_angle='(-inf-36.348477]' sacral_slope='(-inf-34.979458]' class=Hernia 32 ==> degree_spondylolisthesis='(-inf-74.862073]' 32 conf:(1)
58. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' class=Spondylolisthesis 31 ==>
degree_spondylolisthesis='(-inf-74.862073]' 31 conf:(1)
59. pelvic_tilt='(4.642414-15.839776]' 129 ==>
degree_spondylolisthesis='(-inf-74.862073]' 128 conf:(0.99)
60. pelvic_incidence='(46.885145-67.622369]' sacral_slope='(34.979458-56.591985]' 94 ==> degree_spondylolisthesis='(-inf-74.862073]' 93 conf:(0.99)
61. lumbar_lordosis_angle='(36.348477-58.696954]' sacral_slope='(34.979458-56.591985]' 90 ==> degree_spondylolisthesis='(-inf-74.862073]' 89 conf:(0.99)
62. pelvic_incidence='(46.885145-67.622369]' lumbar_lordosis_angle='(36.348477-58.696954]' 76 ==>
degree_spondylolisthesis='(-inf-74.862073]' 75 conf:(0.99)
63. pelvic_incidence='(46.885145-67.622369]' pelvic_radius='(107.277961-125.875654]' 76 ==>
degree_spondylolisthesis='(-inf-74.862073]' 75 conf:(0.99)
64. lumbar_lordosis_angle='(36.348477-58.696954]' pelvic_radius='(107.277961-125.875654]' 72 ==> degree_spondylolisthesis='(-inf-74.862073]' 71 conf:(0.99)
65. pelvic_tilt='(4.642414-15.839776]' lumbar_lordosis_angle='(36.348477-58.696954]' 68 ==>
degree_spondylolisthesis='(-inf-74.862073]' 67 conf:(0.99)
66. pelvic_tilt='(4.642414-15.839776]' pelvic_radius='(107.277961-125.875654]' 68 ==>
degree_spondylolisthesis='(-inf-74.862073]' 67 conf:(0.99)
67. pelvic_tilt='(15.839776-27.037139]' sacral_slope='(34.979458-56.591985]' 59 ==>
degree_spondylolisthesis='(-inf-74.862073]' 58 conf:(0.98)
68. pelvic_incidence='(46.885145-67.622369]' sacral_slope='(34.979458-56.591985]' pelvic_radius='(107.277961-125.875654]' 56 ==>
degree_spondylolisthesis='(-inf-74.862073]' 55 conf:(0.98)
69. pelvic_tilt='(4.642414-15.839776]' class=Spondylolisthesis 49 ==>
degree_spondylolisthesis='(-inf-74.862073]' 48 conf:(0.98)
70. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(15.839776-27.037139]' 48 ==>
degree_spondylolisthesis='(-inf-74.862073]' 47 conf:(0.98)
71. lumbar_lordosis_angle='(36.348477-58.696954]' 133 ==>
degree_spondylolisthesis='(-inf-74.862073]' 130 conf:(0.98)
72. pelvic_incidence='(46.885145-67.622369]' sacral_slope='(34.979458-56.591985]' class=Spondylolisthesis 40 ==>
degree_spondylolisthesis='(-inf-74.862073]' 39 conf:(0.98)
73. lumbar_lordosis_angle='(36.348477-58.696954]' sacral_slope='(34.979458-56.591985]' class=Spondylolisthesis 40 ==>
degree_spondylolisthesis='(-inf-74.862073]' 39 conf:(0.98)
74. pelvic_tilt='(15.839776-27.037139]' sacral_slope='(34.979458-56.591985]' pelvic_radius='(107.277961-125.875654]' 34 ==>
degree_spondylolisthesis='(-inf-74.862073]' 33 conf:(0.97)
75. lumbar_lordosis_angle='(36.348477-58.696954]' sacral_slope='(-inf-34.979458]' 33 ==>
degree_spondylolisthesis='(-inf-74.862073]' 32 conf:(0.97)
76. sacral_slope='(34.979458-56.591985]' pelvic_radius='(88.680268-107.277961]' 33 ==>
degree_spondylolisthesis='(-inf-74.862073]' 32 conf:(0.97)

```
 77. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(15.839776-27.037139]' pelvic_radius='(107.277961-125.875654]' 32 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 31    conf:(0.97)
 78. pelvic_tilt='(15.839776-27.037139]' sacral_slope='(34.979458-56.591985]' class=Spondylolisthesis 32 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 31    conf:(0.97)
 79. pelvic_incidence='(46.885145-67.622369]' 125 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 121    conf:(0.97)
 80. pelvic_tilt='(15.839776-27.037139]' 114 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 110    conf:(0.96)
 81. pelvic_radius='(88.680268-107.277961]' 52 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 50    conf:(0.96)
 82. sacral_slope='(34.979458-56.591985]' pelvic_radius='(107.277961-125.875654]' 97 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 93    conf:(0.96)
 83. pelvic_incidence='(67.622369-88.359593]' lumbar_lordosis_angle='(58.696954-81.045431]' 48 ==> class=Spondylolisthesis 46    conf:(0.96)
 84. pelvic_radius='(107.277961-125.875654]' 179 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 171    conf:(0.96)
 85. pelvic_incidence='(67.622369-88.359593]' lumbar_lordosis_angle='(58.696954-81.045431]'
 degree_spondylolisthesis='(-inf-74.862073]' 41 ==> class=Spondylolisthesis 39    conf:(0.95)
 86. pelvic_radius='(88.680268-107.277961]' class=Spondylolisthesis 40 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 38    conf:(0.95)
 87. pelvic_tilt='(15.839776-27.037139]' pelvic_radius='(107.277961-125.875654]' 78 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 74    conf:(0.95)
 88. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' lumbar_lordosis_angle='(36.348477-58.696954]' 39 ==>
 sacral_slope='(34.979458-56.591985]' 37    conf:(0.95)
 89. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]' lumbar_lordosis_angle='(36.348477-58.696954]'
 degree_spondylolisthesis='(-inf-74.862073]' 39 ==>  sacral_slope='(34.979458-56.591985]' 37    conf:(0.95)
 90. pelvic_incidence='(46.885145-67.622369]' pelvic_tilt='(4.642414-15.839776]'
 lumbar_lordosis_angle='(36.348477-58.696954]' 39 ==> sacral_slope='(34.979458-56.591985]'
 degree_spondylolisthesis='(-inf-74.862073]' 37    conf:(0.95)
 91. sacral_slope='(-inf-34.979458]' 95 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 90    conf:(0.95)
 92. lumbar_lordosis_angle='(36.348477-58.696954]' class=Spondylolisthesis 53 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 50    conf:(0.94)
 93. pelvic_tilt='(15.839776-27.037139]' lumbar_lordosis_angle='(58.696954-81.045431]' 33 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 31    conf:(0.94)
 94. sacral_slope='(34.979458-56.591985]' 172 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 161    conf:(0.94)
 95. pelvic_incidence='(67.622369-88.359593]' pelvic_tilt='(15.839776-27.037139]' 43 ==>
 degree_spondylolisthesis='(-inf-74.862073]' 40    conf:(0.93)
 96. pelvic_incidence='(67.622369-88.359593]' pelvic_radius='(107.277961-125.875654]' 54 ==> class=Spondylolisthesis 50    conf:(0.93)
 97. pelvic_tilt='(4.642414-15.839776]' sacral_slope='(-inf-34.979458]' 40 ==> pelvic_incidence='(-inf-46.885145]' 37    conf:(0.93)
 98. pelvic_tilt='(4.642414-15.839776]' sacral_slope='(-inf-34.979458]'
 degree_spondylolisthesis='(-inf-74.862073]' 40 ==> pelvic_incidence='(-inf-46.885145]' 37    conf:(0.93)
 99. pelvic_tilt='(4.642414-15.839776]' sacral_slope='(-inf-34.979458]' 40 ==> pelvic_incidence='(-inf-46.885145]'
 degree_spondylolisthesis='(-inf-74.862073]' 37    conf:(0.93)
100. sacral_slope='(34.979458-56.591985]' pelvic_radius='(107.277961-125.875654]' class=Spondylolisthesis 52 ==>
degree_spondylolisthesis='(-inf-74.862073]' 48    conf:(0.92)
```

APÊNDICE C – Refinamento dos Parâmetros de Configuração dos Algoritmos de Classificação sobre as Bases de Dados

C.1 Refinamento dos Parâmetros de Configuração do Algoritmo BayesNet Sobre a Base de Dados de Câncer de Mama

```
=== Run information ===

Scheme:weka.classifiers.bayes.BayesNet -D -Q
weka.classifiers.bayes.net.search.local.K2 -- -P 1 -S BAYES -E
weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5

Instances:    466
Attributes:   10
              ClumpThickness
              UniformityofCellSize
              UniformityofCellShape
              MarginalAdhesion
              SingleEpithelialCellSize
              BareNuclei
              BlandChromatin
              NormalNucleoli
              Mitoses
              Class
Test mode:user supplied test set: size unknown (reading incrementally)

=== Classifier model (full training set) ===

Bayes Network Classifier
not using ADTree
#attributes=10 #classindex=9
Network structure (nodes followed by parents)
ClumpThickness(10): Class
UniformityofCellSize(10): Class
UniformityofCellShape(10): Class
MarginalAdhesion(10): Class
SingleEpithelialCellSize(10): Class
BareNuclei(10): Class
```

```
BlandChromatin(10): Class
NormalNucleoli(10): Class
Mitoses(10): Class
Class(2):
LogScore Bayes: -5655.852063208376
LogScore BDeu: -6134.12702964486
LogScore MDL: -6140.3774113426625
LogScore ENTROPY: -5639.626282161422
LogScore AIC: -5802.626282161422


Time taken to build model: 0.02 seconds

=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances         226               96.9957 %
Incorrectly Classified Instances         7                3.0043 %
Kappa statistic                          0.9328
Mean absolute error                      0.0339
Root mean squared error                  0.1761
Relative absolute error                  7.5393 %
Root relative squared error             37.4885 %
Total Number of Instances              233

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.962     0.013      0.993     0.962     0.977      0.991    2
                 0.987     0.038      0.926     0.987     0.955      0.991    4
Weighted Avg.    0.97      0.021      0.971     0.97      0.97       0.991

=== Confusion Matrix ===

   a   b   <-- classified as
 151   6 |   a = 2
   1  75 |   b = 4

================ End of Run information ===================
```

C.2 Refinamento dos Parâmetros de Configuração do Algoritmo BayesNet Sobre a Base de Dados de Dermatologia

```
=== Run information ===

Scheme:weka.classifiers.bayes.BayesNet -D -Q
weka.classifiers.bayes.net.search.local.TAN -- -S BAYES -E
weka.classifiers.bayes.net.estimate.SimpleEstimator -- -A 0.5

Instances:    244
Attributes:   35
              erythema
              scaling
              definite
              itching
              koebner
              polygonal
              follicular
              oral
              knee
              scalp
              family
              melanin
              eosinophils
              PNL
```

```
              fibrosis
              exocytosis
              acanthosis
              hyperkeratosis
              parakeratosis
              clubbing
              elongation
              thinning
              spongiform
              munro
              focal
              disappearance
              vacuolisation
              spongiosis
              sawtooth
              horn
              perifollicular
              inflammatory
              band-like
              Age
              class
Test mode:user supplied test set: size unknown (reading incrementally)

=== Classifier model (full training set) ===

Bayes Network Classifier
not using ADTree
#attributes=35 #classindex=34
Network structure (nodes followed by parents)
erythema(4): class follicular
scaling(4): class definite
definite(4): class oral
itching(4): class
koebner(4): class itching
polygonal(4): class itching
follicular(4): class itching
oral(4): class parakeratosis
knee(4): class clubbing
scalp(4): class clubbing
family(4): class inflammatory
melanin(4): class scaling
eosinophils(4): class exocytosis
PNL(4): class perifollicular
fibrosis(4): class definite
exocytosis(4): class perifollicular
acanthosis(4): class elongation
hyperkeratosis(4): class follicular
parakeratosis(4): class clubbing
clubbing(4): class hyperkeratosis
elongation(4): class clubbing
thinning(4): class parakeratosis
spongiform(4): class thinning
munro(4): class PNL
focal(4): class oral
disappearance(4): class koebner
vacuolisation(4): class oral
spongiosis(4): class oral
sawtooth(4): class vacuolisation
horn(4): class hyperkeratosis
perifollicular(4): class horn
inflammatory(4): class perifollicular
band-like(4): class inflammatory
Age(10): class clubbing
class(6):
LogScore Bayes: -6552.142183246518
LogScore BDeu: -17611.17463624836
LogScore MDL: -15196.131293699455
LogScore ENTROPY: -8206.481895239005
LogScore AIC: -10749.481895239052


Time taken to build model: 0.07 seconds
```

```
=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances         121               99.1803 %
Incorrectly Classified Instances         1                0.8197 %
Kappa statistic                          0.9898
Mean absolute error                      0.0103
Root mean squared error                  0.0638
Relative absolute error                  3.8502 %
Root relative squared error             17.3096 %
Total Number of Instances              122

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.971     0          1         0.971     0.985      1        1
                 1         0.011      0.964     1         0.982      0.999    2
                 1         0          1         1         1          1        3
                 1         0          1         1         1          1        4
                 1         0          1         1         1          1        5
                 1         0          1         1         1          1        6
Weighted Avg.    0.992     0.002      0.992     0.992     0.992      1

=== Confusion Matrix ===

  a  b  c  d  e  f   <-- classified as
 33  1  0  0  0  0 |  a = 1
  0 27  0  0  0  0 |  b = 2
  0  0 20  0  0  0 |  c = 3
  0  0  0 13  0  0 |  d = 4
  0  0  0  0 21  0 |  e = 5
  0  0  0  0  0  7 |  f = 6

================= End of Run information ===================
```

C.3 Refinamento dos Parâmetros de Configuração do Algoritmo FT Sobre a Base de Dados da Coluna Vertebral

```
=== Run information ===

Scheme:weka.classifiers.trees.FT -I 15 -F 1 -M 15 -W 0.0
Instances:    206
Attributes:   7
              pelvic_incidence
              pelvic_tilt
              lumbar_lordosis_angle
              sacral_slope
              pelvic_radius
              degree_spondylolisthesis
              class
Test mode:user supplied test set: size unknown (reading incrementally)

=== Classifier model (full training set) ===

FT Leaves tree
------------------
: FT_1:15/15 (206)

Number of Leaves  :  1

Size of the Tree :  1
FT_1:
Class 0 :
11.4 +
```

```
[pelvic_tilt] * 0.09 +
[lumbar_lordosis_angle] * -0.02 +
[sacral_slope] * -0.14 +
[pelvic_radius] * -0.06 +
[degree_spondylolisthesis] * -0.04

Class 1 :
-9.86 +
[lumbar_lordosis_angle] * 0.04 +
[sacral_slope] * 0.05 +
[pelvic_radius] * 0.01 +
[degree_spondylolisthesis] * 0.33

Class 2 :
-7.6 +
[pelvic_tilt] * -0.07 +
[lumbar_lordosis_angle] * 0.04 +
[sacral_slope] * 0    +
[pelvic_radius] * 0.07 +
[degree_spondylolisthesis] * -0.06


Time taken to build model: 0.07 seconds

=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances          91               87.5    %
Incorrectly Classified Instances        13               12.5    %
Kappa statistic                          0.794
Mean absolute error                      0.1215
Root mean squared error                  0.2692
Relative absolute error                 29.2875 %
Root relative squared error             59.6281 %
Total Number of Instances              104

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.733     0.067      0.647     0.733     0.688      0.936    Hernia
                 0.962     0.038      0.962     0.962     0.962      0.989    Spondylolisthesis
                 0.811     0.075      0.857     0.811     0.833      0.939    Normal
Weighted Avg.    0.875     0.056      0.879     0.875     0.876      0.963

=== Confusion Matrix ===

  a  b  c   <-- classified as
 11  0  4 |  a = Hernia
  1 50  1 |  b = Spondylolisthesis
  5  2 30 |  c = Normal

================ End of Run information ==================
```

APÊNDICE D – Telas Demonstrativas da Interface *do* Software Weka

Figura 67: Tela inicial do *Weka*

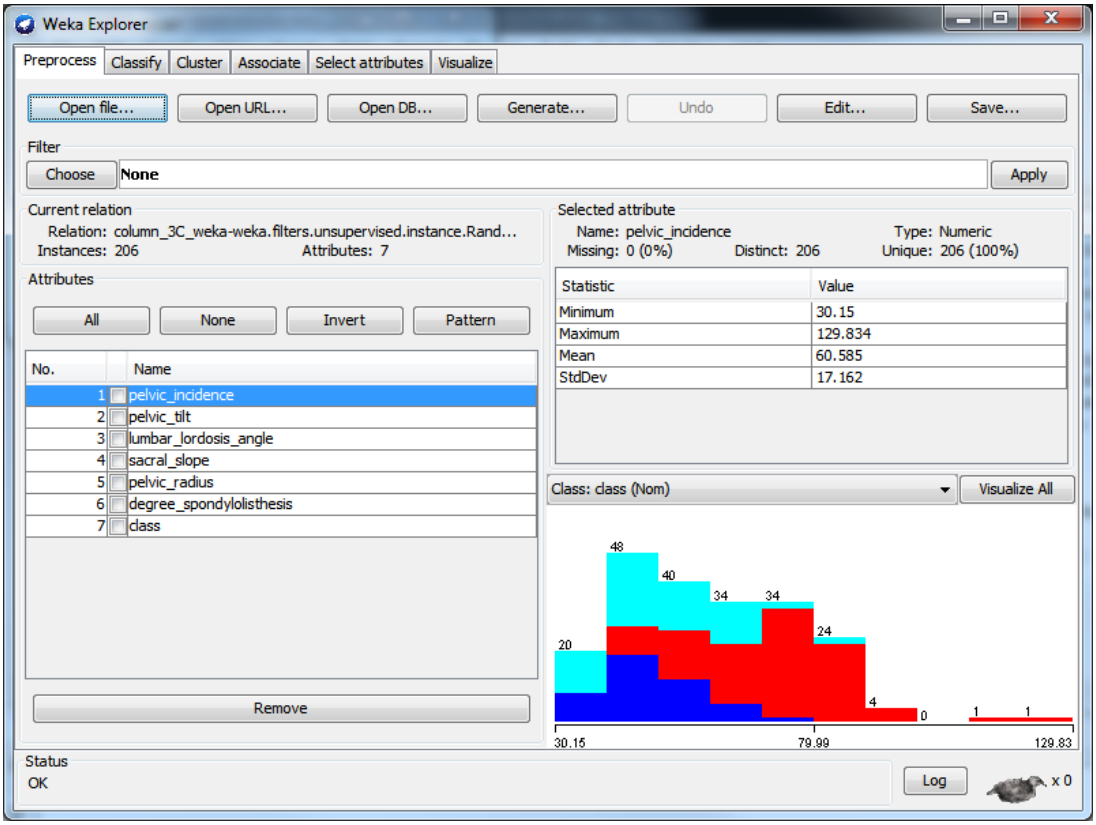

Figura 68: Tela da aba *Preprocess* do modo *Explorer* do *Weka*

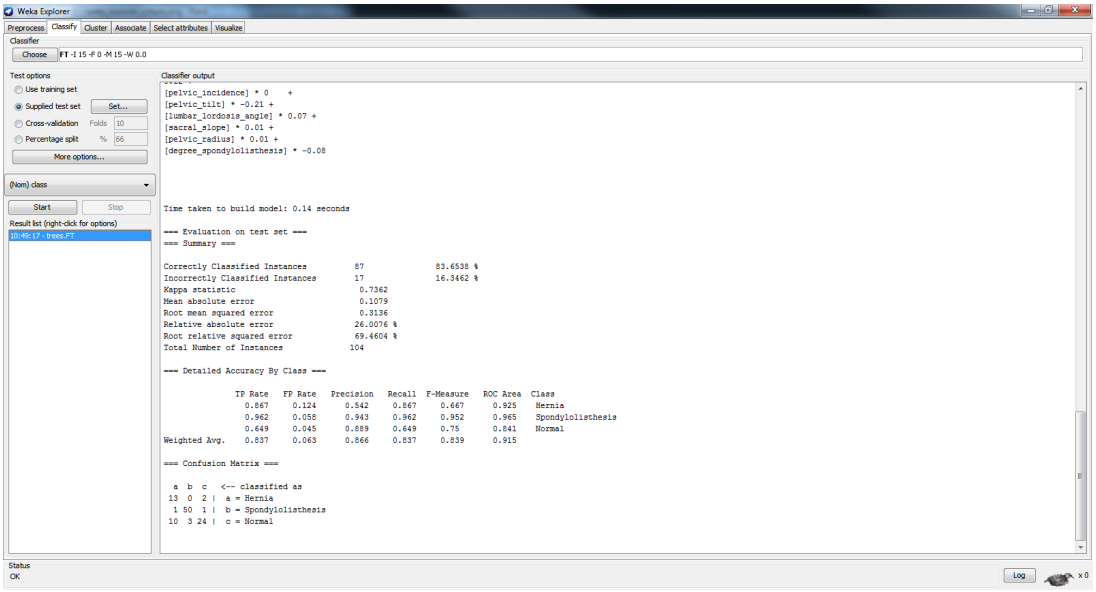

Figura 69: Tela da aba *Classify* do modo *Explorer* do *Weka*

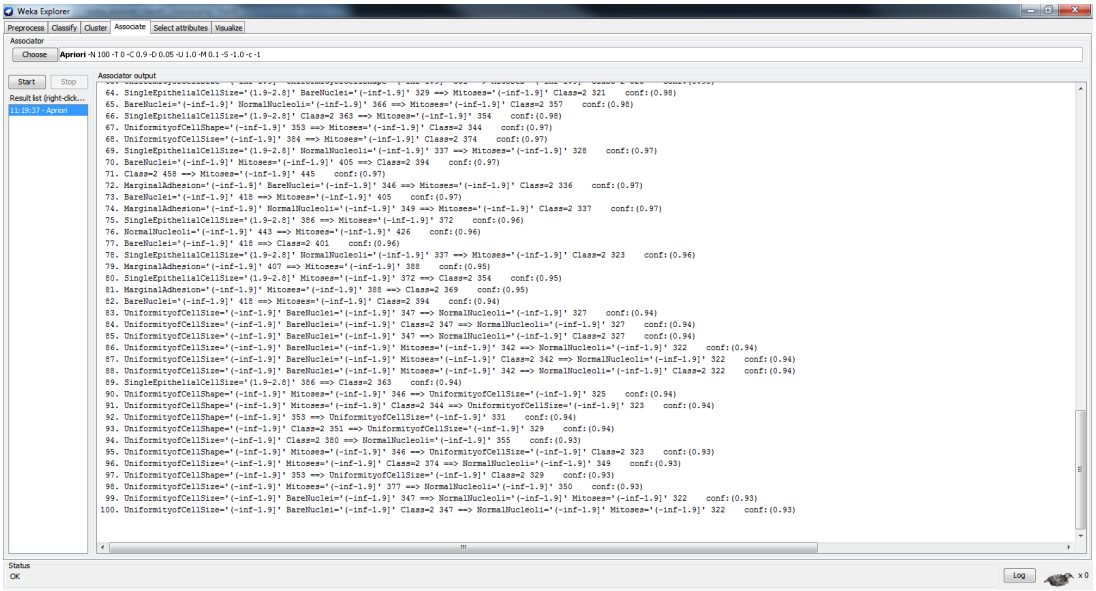

Figura 70: Tela da aba *Associate* do modo *Explorer* do *Weka*

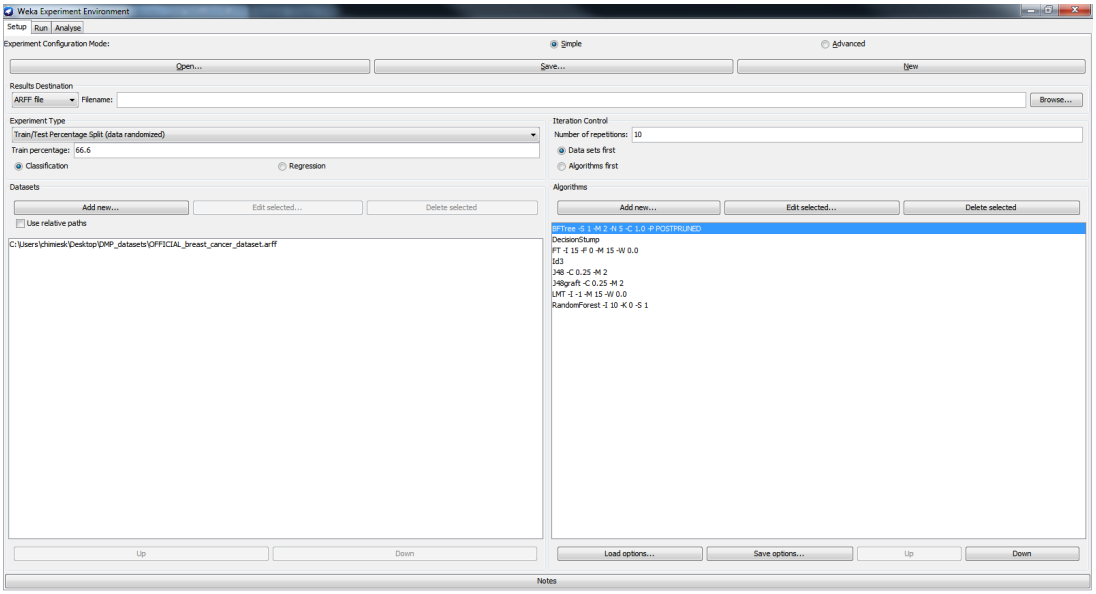

Figura 71: Tela da aba *Setup* do modo *Experimenter* do *Weka*

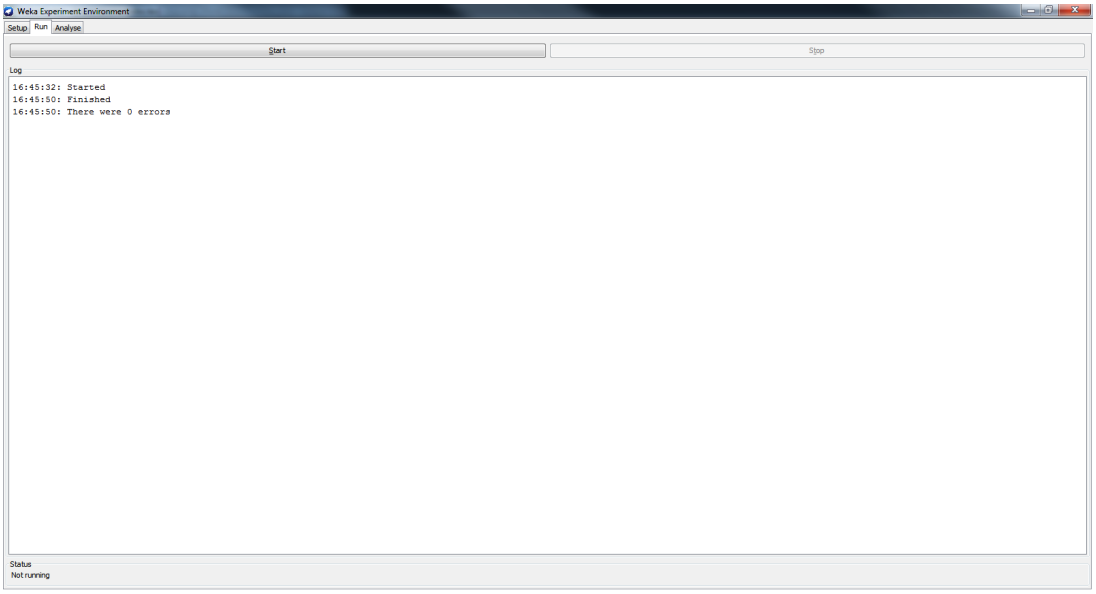

Figura 72: Tela da aba *Run* do modo *Experimenter* do *Weka*

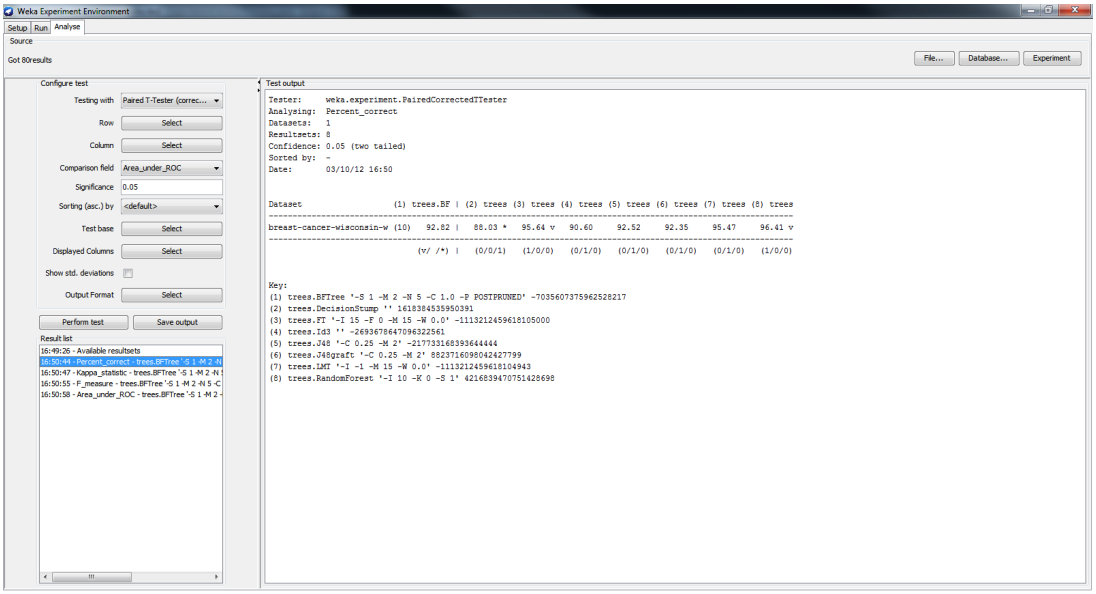

Figura 73: Tela da aba *Analyse* do modo *Experimenter* do *Weka*

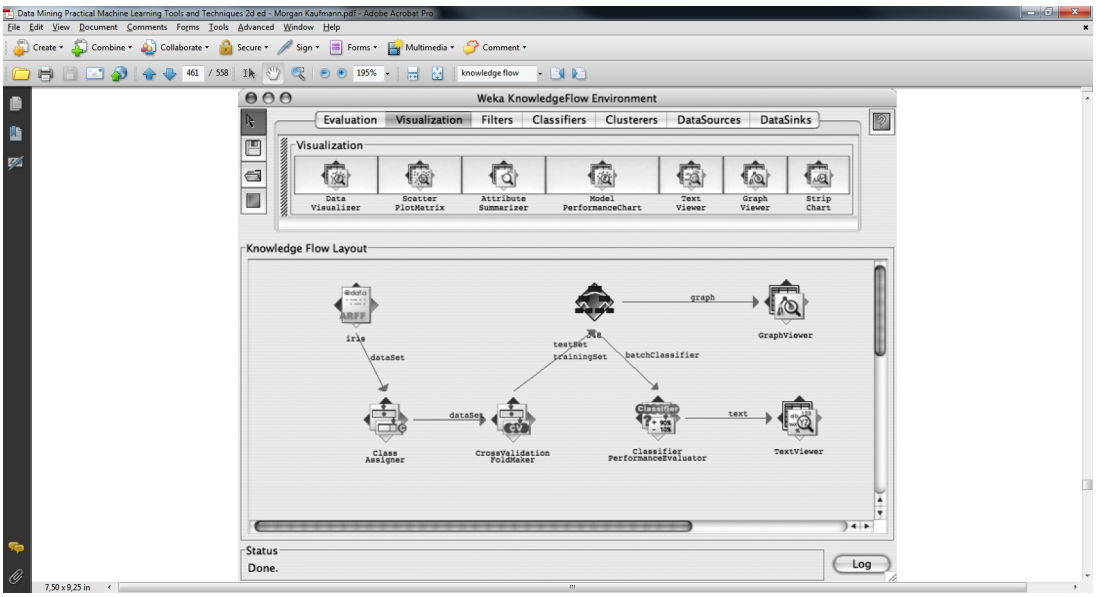

Figura 74: Tela do modo *Knowledge Flow* do *Weka*

APÊNDICE E – Gráficos dos Níveis de Uniformidade do Tamanho e da Forma das Células de Tumor de Mama

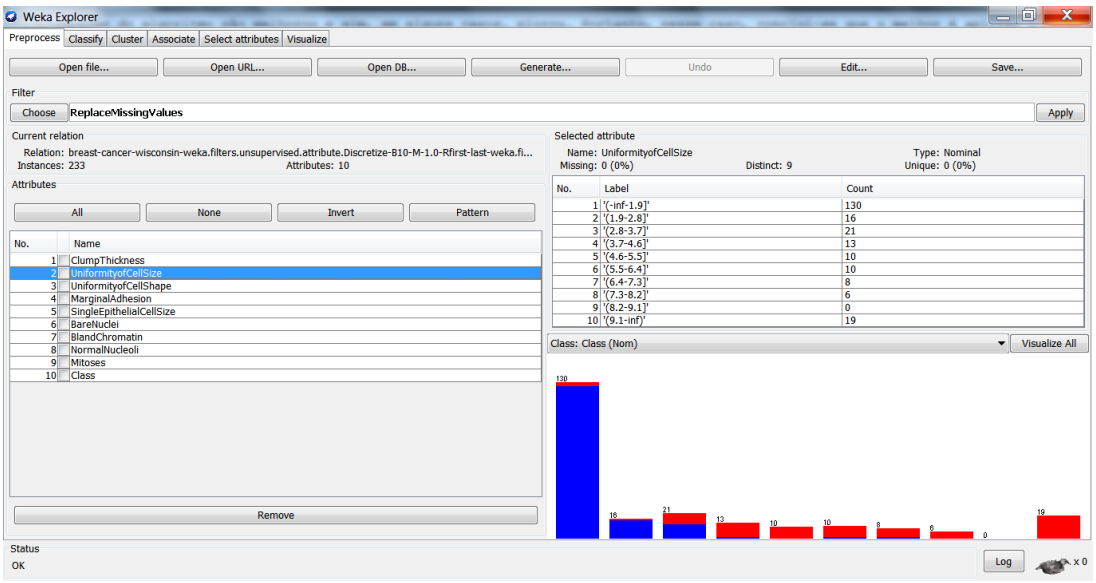

Figura 75: Gráfico da relação entre os níveis de uniformidade do tamanho da célula e seus diagnósticos associados.

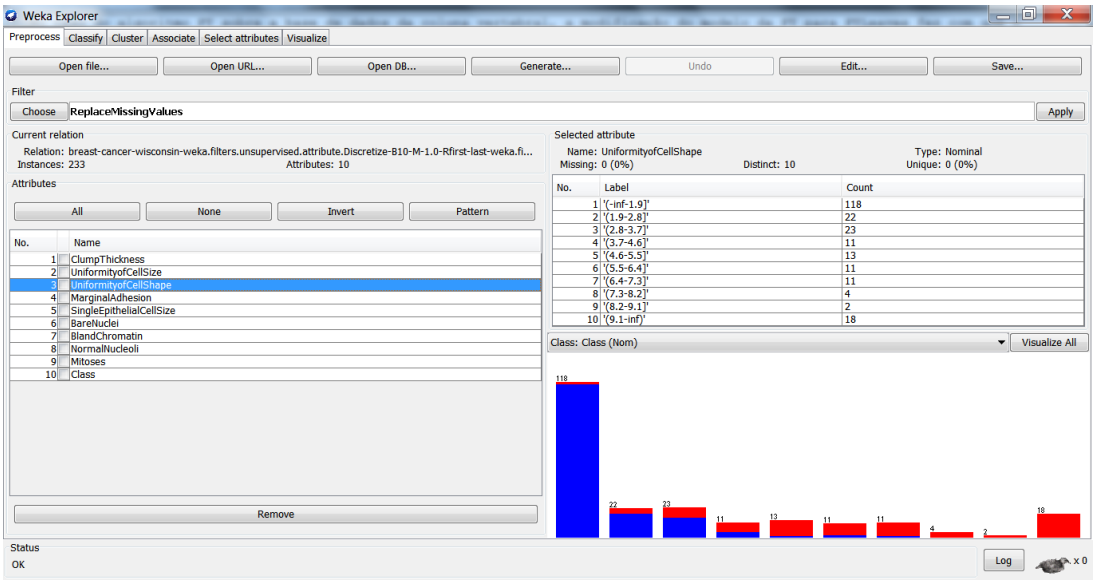

Figura 76: Gráfico da relação entre os níveis de uniformidade da forma da célula e seus diagnósticos associados.

Referências

[1] C. Carroll, P. Marsden, P. Soden, E. Naylor, J. New, and T. Dornan, "Involving users in the design and usability evaluation of a clinical decision support system," *Computer Methods and Programs in Biomedicine*, pp. 123–135, 2002.

[2] C. W. Hanson, *Healthcare informatics*. McGraw-Hill, 2006.

[3] S. Tsumoto and S. Hirano, "Hospital Management Based on Data Mining," *2008 Eighth International Conference on Intelligent Systems Design and Applications*, pp. 257–262, Nov. 2008.

[4] J. C. B. A. A. L. Filho, José Rodrigues Xavier, "A tecnologia da informação na Área hospitalar: um caso de implementação de um sistema de registro de pacientes," *Revista de Administração Contemporânea*, vol. Volume 5, pp. 105–120, 2001.

[5] J. Favela, M. Rodríguez, A. Preciado, and V. M. González, "Integrating context-aware public displays into a mobile hospital information system.," *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 8, pp. 279–86, Sept. 2004.

[6] U. Johansson, "Obtaining accurate and comprehensible data mining models: An evolutionary approach," no. 1086, 2007.

[7] M. Persson and N. Lavesson, "Identification of Surgery Indicators by Mining Hospital Data: A Preliminary Study," *2009 20th International Workshop on Database and Expert Systems Application*, pp. 323–327, 2009.

[8] S. Nirkhi, "Potential use of Artificial Neural Network in Data Mining," *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, pp. 339–343, Feb. 2010.

[9] S. Dreiseitl and M. Binder, "Do physicians value decision support? A look at the effect of decision support systems on physician opinion.," *Artificial intelligence in medicine*, vol. 33, pp. 25–30, Jan. 2005.

[10] S. Kramer, "Mobile devices: Changing healthcare forever," June 2012. http://www.v3im.com/2012/06/mobile-devices-changing-healthcare-forever/#axzz24kP8K2QQ, Acessado em Nov. 12, 2012.

[11] M. Gentry, "Mobile device options for the healthcare professional," February 2012. http://www.med.yale.edu/library/services/computing/pdahardware.html, Acessado em Nov. 12 , 2012.

[12] C. Wang, "The research of Android System architecture and application programming," *Proceedings of 2011 International Conference on Computer Science and Network Technology*, pp. 785–790, Dec. 2011.

[13] N. Kahn, "Hospital information systems: An aid to decision making," *Third International Conference on Emerging Trends in Engineering and Technology*, pp. 657–663, 2010.

[14] K. A. Wager, F. W. Lee, J. P. Glaser, and L. R. Burns, *Health Care Information Systems: A Practical Approach for Health Care Management*. Jossey-bass, 2009.

[15] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI magazine*, vol. 17, no. 3, pp. 37–54, 1996.

[16] I. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*. Elsevier, 2005.

[17] J. A. P. Vieira, "Algorithm development for physiological signals analysis and cardiovascular disease diagnosis - a data mining approach," Master's thesis, 2011.

[18] N. Landwehr, M. Hall, and E. Frank, "Logistic Model Trees," *Machine Learning*, vol. 59, pp. 161–205, May 2005.

[19] J. Gama, "Functional trees for classification," *Proceedings 2001 IEEE International Conference on Data Mining*, pp. 147–154, 2001.

[20] C. Souza, "Análise de poder discriminativo através de curvas roc." http://crsouza.blogspot.com.br/2009/07/analise-de-poder-discriminativo-atraves.html.

[21] E. Martinez, F. Louzada-Neto, and B. Pereira, "A curva ROC para testes diagnósticos," *Cad Saúde Coletiva*, 2003.

[22] P. Tighe, S. Laduzenski, D. Edwards, N. Ellis, A. P. Boezaart, and H. Aygtug, "Use of machine learning theory to predict the need for femoral nerve block following ACL repair.," *Pain medicine (Malden, Mass.)*, vol. 12, pp. 1566–75, Oct. 2011.

[23] M. Othman and T. Yau, "Comparison of different classification techniques using WEKA for breast cancer," *3rd Kuala Lumpur International Conference on Biomedical Engineering*, vol. 15, pp. 520–523, 2007.

[24] L. Natis, "Equações de Estimação para a Estatística KAPPA," pp. 2895–2906, 2001.

[25] D. Xhemali, C. Hinde, and R. Stone, "Naïve Bayes vs. Decision Trees vs. Neural Networks in the classification of training web pages," *IJCSI*, vol. 4, no. 1, pp. 16–23, 2009.

[26] J. P. Lucas, "Mineração de dados apoiada pela descoberta de subgrupos através do pós-processamento de regras de associação," Master's thesis, Universidade Federal de Pelotas, Instituto de Física e Matemática, Departamento de Informática, 2006.

[27] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, *Top 10 algorithms in data mining*, vol. 14. Dec. 2007.

[28] O. Mangasarian, "UCI machine learning repository," 1992. http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29, Acessado em Nov. 12, 2012.

[29] H. A. Guvenir, "UCI machine learning repository," 1998. http://archive.ics.uci.edu/ml/datasets/Dermatology, Acessado em Nov. 12, 2012.

[30] G. d. A. Barreto and H. A. F. d. M. Filho, "UCI machine learning repository," 2011. http://archive.ics.uci.edu/ml/datasets/Vertebral+Column, Acessado em Nov. 12, 2012.