UNIVERSIDADE FEDERAL DOS VALES DO JEQUITINHONHA E MUCURI

Programa de Pós-Graduação em Educação

Rodrigo Magalhães Mota dos Santos

Técnicas de Aprendizagem de Máquina Utilizadas na Previsão de Desempenho Acadêmico

Diamantina

2016

Rodrigo Magalhães Mota dos Santos

Técnicas de Aprendizagem de Máquina Utilizadas na Previsão de Desempenho Acadêmico

Dissertação apresentada ao programa de Pós-Graduação em Educação da Universidade Federal dos Vales do Jequitinhonha e Mucuri, como requisito para obtenção do título de Mestre.

Orientador: Prof. Dr. Cristiano Grijó Pitangui

Diamantina

2016

Aline Pereira da Costa - CRB6/2798

Santos, Rodrigo Magalhães Mota dos
006.312
S237t Técnicas de aprendizagem de máquina utilizadas
na previsão de desempenho acadêmico / Rodrigo Magalhães Mota
dos Santos. – 2016.
90f. : il.
Orientador: Dr. Cristiano Grijó Pitangui.
Dissertação (Mestrado em Educação) – Universidade Federal dos
Vales do Jequitinhonha e Mucuri, 2016. Programa de Pós-Graduação em Educação.

1. Mineração de dados. 2. Mineração de dados educacionais.
3. Educação. 4. Educação à distância. 5. Ambientes virtuais de aprendizagem
I. Pitangui, Cristiano Grijó. II. Universidade Federal dos Vales do
Jequitinhonha e Mucuri. Programa de Pós-Graduação em Educação. III. Título.

Rodrigo Magalhães Mota dos Santos

Técnicas de Aprendizagem de Máquina Utilizadas na Previsão de Desempenho Acadêmico

Dissertação apresentada ao programa de Pós-Graduação em Educação da Universidade Federal dos Vales do Jequitinhonha e Mucuri, como requisito para obtenção do título de Mestre.

Data de aprovação:

Prof. Dr. Cristiano Grijó Pitangui
Orientador

Prof. Dr. Alessandro Vivas Andrade
Departamento de
Computação/FACET/UFVJM

Profa. Dra. Luciana Pereira de Assis
Departamento de
Computação/FACET/UFVJM

Prof. Msc. Rafael Santin
Departamento de
Computação/FACET/UFVJM

Diamantina
2016

Dedico este trabalho a minha mãe Zildete, minha irmã Amanda e a minha avó Dália.

Agradecimentos

Agradeço primeiramente a Deus por tudo de bom que vem proporcionando.
Agradeço a minha família por todo o carinho.
Agradeço a minha namorada Lorena pela paciência, compreensão, e incentivo.
Agradeço ao Professor Cristiano pela excelente orientação.
Agradeço aos Professores Alessandro e Luciana pelos apontamentos valiosos na pesquisa.
Agradeço ao Professor Everton Luiz de Paula pelo apoio a pesquisa.
Agradeço aos colegas de trabalho pelo companherismo.

"Só se pode alcançar um grande êxito quando nos mantemos fiéis a nós mesmos"

Friedrich Nietzsche

Resumo

A tecnologia, presente cada vez mais no ambiente educacional, tem contribuído para o aumento da oferta de cursos à distância. Grande parte dos cursos ofertados nesta modalidade utilizam os Ambientes Virtuais de Aprendizagem (AVA). Estes ambientes ganham espaço no cotidiano dos educadores devido ao fácil manuseio e a grande diversidade de ferramentas disponibilizadas. Tais ferramentas permitem, de forma geral, a administração de cursos totalmente à distância com oferta de múltiplas mídias e recursos (fóruns de discussão, chats, dentre outros) para interações entre professores e alunos. Tais interações criam enormes volumes de dados que podem ser analisados através da aplicação de técnicas de Mineração de Dados Educacionais. Com a aplicação destas técnicas pode-se realizar a previsão de desempenho acadêmico que pode ter grande utilidade para Instituições de Ensino no sentido de auxiliá-las a tomar, de forma antecipada, decisões pedagógicas que possam ajudar os estudantes.

Este trabalho apresenta um estudo de métodos como Seleção de Atributos utilizando a abordagem Wrapper e Classificador em Cascata, ainda não empregados em trabalhos correlatos pesquisados, que visam melhorar os resultados obtidos pelas técnicas de Mineração de Dados Educacionais utilizadas na previsão de desempenho acadêmico de estudantes. Os resultados experimentais indicam uma melhora no desempenho dos algoritmos classificadores utilizados (alguns alcançando a notável marca de 90,2% de acurácia), bem como apontam quais os recursos utilizados no AVA possuem maior influência no desempenho dos estudantes.

Palavras-chaves: Mineração de Dados, Mineração de Dados Educacionais, Educação, Educação à Distância, Ambientes Virtuais de Aprendizagem

Abstract

The technology, which is being increasingly used in the educational environment, has contributed for the popularity of distance courses. Much of the courses offered in this mode uses the so-called Virtual Learning Environments (VLE). These environments are gaining ground in the daily lives of educators due to its easy handling and the wide variety of available tools. These tools allow, in general, the administration of fully distance courses with multiple media and resources (forums, chats, among others) for interactions between teachers and students. These interactions create huge volumes of data that can be analyzed through the application of Educational Data Mining techniques. Such techniques can be used to academic performance prediction that can be very useful for education institutions in order to help them to take, in advance, pedagogical decisions that can help students.

This work presents a study of methods as Feature Selection using the Wrapper approach and Classifier Cascade that were not employed in other works, with the aim to improve the results obtained by Educational Data Mining techniques used in the academic performance prediction. Results showed an improvement in the performance of classifiers (some obtaining the remarkable mark of 90.2% in accuracy results), as well as pointed out what the resources used in VLE that have greater influence on student performance.

Key-words: Data Mining, Education Data Mining, Education, Distance Education, Virtual Learning Environments.

Lista de ilustrações

Figura 1 – Processo básico de classificação 29
Figura 2 – Árvore de decisão 31
Figura 3 – Exemplo simples do funcionamento do algoritmo KNN 33
Figura 4 – Seleção de atributos pela abordagem Filtro (*Filter*) 36
Figura 5 – Seleção de atributos pela abordagem Cápsula (*Wrapper*) 36
Figura 6 – Etapas do processo de KDD 39
Figura 7 – Três principais áreas relacionadas com a EDM 41
Figura 8 – Etapas da Mineração de Dados Educacionais 42
Figura 9 – Tela inicial do *Weka* 49
Figura 10 – Arquivo de treinamento no formato arff 50
Figura 11 – Processo de obtenção dos dados 52
Figura 12 – Comparação dos resultados para a classificação dos conjuntos de dados original e balanceado do Curso de Matemática 58
Figura 13 – Divisão dos períodos da série temporal 62
Figura 14 – Conjunto de dados original mais nota da primeira avaliação 63
Figura 15 – Conjunto de dados original mais as notas da primeira e segunda avaliação . 63
Figura 16 – Abordagem Classificador em Cascata 73

Lista de tabelas

Tabela 1 – Conjunto de dados fictícios para concessão de empréstimos 30
Tabela 2 – Matriz de Confusão para problemas com duas classes 34
Tabela 3 – Relação dos atributos utilizados 54
Tabela 4 – Relação dos conjuntos de dados obtidos e suas respectivas classes 54
Tabela 5 – Resultados para a classificação dos conjuntos de dados originais 56
Tabela 6 – Resultados para a classificação dos conjuntos de dados original e balanceado do Curso de Matemática 57
Tabela 7 – Comparação das F-Measure obtidas pelos algoritmos classificadores na classe Reprovado do Curso de Matemática 58
Tabela 8 – Resultados para a classificação com Seleção de Atributos 60
Tabela 9 – Comparação das acurácias obtidas pelos algoritmos de Classificação com Seleção de Atributos 61
Tabela 10 – Comparação dos resultados para a classificação dos conjuntos original e seleção de atributos pelos métodos cápsula e filtro 62
Tabela 11 – Resultados para a classificação do conjunto original em séries temporais . . 65
Tabela 12 – Resultados para a classificação do conjunto original mais notas parciais em séries temporais 67
Tabela 13 – Resultados para a classificação do conjunto usando a técnica de seleção de atributos pelo método cápsula mais notas parciais em séries temporais . . . 69
Tabela 14 – Comparação dos resultados das três abordagens propostas no experimento com séries temporais 70
Tabela 15 – Relação dos conjuntos de dados obtidos de semestres anteriores e suas respectivas classes 71
Tabela 16 – Resultados para a classificação dos conjuntos de dados originais utilizando modelo construído com dados do semestre anterior 71
Tabela 17 – Atributos retornados pelo algoritmo InfoGainAttributeEval 72
Tabela 18 – Resultados para a classificação no conjunto de dados utilizando classificador em cascata do Curso de Administração 74
Tabela 19 – Comparação dos resultados da classificação no conjunto de dados do curso de Administração utilizando o conjunto de dados original e com classificador em cascata 75
Tabela 20 – Resultados para a classificação no conjunto de dados utilizando classificador em cascata do Curso de Física 76
Tabela 21 – Comparação dos resultados da classificação no conjunto de dados do curso de Física utilizando o conjunto de dados original e com classificador em cascata 77

Tabela 22 – Resultados para a classificação no conjunto de dados utilizando classificador em cascata do Curso de Matemática 78
Tabela 23 – Comparação dos resultados da classificação no conjunto de dados do curso de Matemática utilizando o conjunto de dados original e com classificador em cascata 79
Tabela 24 – Resultados para a classificação no conjunto de dados utilizando classificador em cascata do Curso de Química 80
Tabela 25 – Comparação dos resultados da classificação no conjunto de dados do curso de Química utilizando o conjunto de dados original e com classificador em cascata 81

Lista de abreviaturas e siglas

ARFF	*Attribute Relation File Format*
AVA	Ambiente Virtual de Aprendizagem
DESAFIE	*Workshop* de Desafios da Computação Aplicada à Educação
DM	*Data Mining*
DNA	*DeoxyriboNucleic Acid*
DW	*Data Warehouse*
EDM	*Education Data Mining*
FN	Falsos Negativos (*False Negative*)
FP	Falsos Positivos (*False Positive*)
IE	Informática na Educação
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
KDD	*Knowledge Discovery in Databases*
RBIE	Revista Brasileira de Informática na Educação
SBIE	Simpósio Brasileiro de Informática na Educação
SGA	Sistema de Gerenciamento Acadêmico
SGBD	Sistema Gerenciador de Banco de Dados
SMOTE	*Synthetic Minority Over-sampling Technique*
STI	Sistema Tutor Inteligente
TIC	Tecnologia da Informação e Comunicação
TP	Verdadeiros Positivos (*True Positive*)
TN	Verdadeiros Negativos (*True Negative*)

Sumário

1 INTRODUÇÃO . . . 23
1.1 Relevância da Pesquisa . . . 23
1.2 Objetivos . . . 24
1.3 Metodologia . . . 25
1.4 Estrutura do Trabalho . . . 25

2 APRENDIZADO DE MÁQUINA . . . 27
2.1 Aprendizado Supervisionado . . . 28
2.1.1 Classificação . . . 29
2.1.2 Métricas de Avaliação dos Classificadores . . . 33
2.1.3 Conjunto de Treinamento e Teste . . . 35
2.1.4 Seleção de Atributos . . . 35
2.1.5 Balanceamento de Classes . . . 37
2.2 Aplicações do Aprendizado de Máquina . . . 37
2.3 Descoberta de Conhecimento em Base de Dados . . . 38
2.3.1 Etapas do Processo de KDD . . . 39
2.3.1.1 Pré-Processamento . . . 39
2.3.1.2 Mineração de Dados . . . 40
2.3.1.3 Pós-Processamento . . . 40

3 MINERAÇÃO DE DADOS EDUCACIONAIS . . . 41
3.1 Trabalhos Relacionados . . . 43
3.1.1 Comparação de Algoritmos . . . 43
3.1.2 Balanceamento de Classes . . . 44
3.1.3 Seleção de Atributos . . . 44
3.1.4 Séries Temporais . . . 46

4 FERRAMENTAS, MÉTODOS E EXPERIMENTOS . . . 49
4.1 Weka . . . 49
4.1.1 Arquivos ARFF . . . 50
4.2 Moodle . . . 51
4.3 Algoritmos Utilizados . . . 51
4.4 Base de Dados . . . 52
4.5 Experimentos . . . 55
4.5.1 Experimento 1 - Comparação de Algoritmos . . . 55
4.5.2 Experimento 2 - Balanceamento de Classes . . . 55

4.5.3 Experimento 3 - Seleção de Atributos 58
4.5.4 Experimento 4 - Séries Temporais 62
4.5.5 Experimento 5 - Utilizando Base Antiga para Classificar Novos Conjuntos . . 70
4.5.6 Experimento 6 - Classificador em Cascata 73

5 CONCLUSÕES E TRABALHOS FUTUROS 83

REFERÊNCIAS 87

1 Introdução

A tecnologia, presente cada vez mais no ambiente educacional, tem contribuído para o aumento da oferta de cursos à distância. Grande parte dos cursos ofertados nesta modalidade utilizam os Ambientes Virtuais de Aprendizagem (AVA). Para FRANÇA (2000), AVA é o espaço que organiza os recursos e ferramentas para acesso aos cursos, por meio da interação com os conteúdos, realização de atividades de aprendizagem, interação com o professor e colegas. Estes ambientes ganham espaço no cotidiano dos educadores devido ao fácil manuseio e a grande diversidade de ferramentas disponibilizadas. Tais ferramentas permitem, de forma geral, a administração de cursos totalmente à distância com oferta de múltiplas mídias para interações entre professores e estudantes, e vários recursos, tais como fóruns de discussão, *chat's*, dentre outros.

Com tamanha gama de ferramentas, os AVA's se tornaram detentores de inúmeras informações, registros de acessos, e interações. Este enorme volume de dados armazenado está sendo estudado por profissionais da área de Informática na Educação (IE).

Utilizando técnicas de Mineração de Dados Educacionais (do inglês, *Education Data Mining* – EDM) profissionais de IE podem descobrir padrões e tendências implícitas em grandes conjuntos de dados. Estas informações podem auxiliar professores e gestores na adoção de estratégias pedagógicas com o objetivo de melhorar o processo de ensino e aprendizagem nas Instituições Educacionais. Também é possível verificar se o estudante está desmotivado ou confuso e, assim, personalizar o ambiente e os métodos de ensino para oferecer melhores condições de aprendizagem (BAKER; ISOTANI; CARVALHO, 2011). Desta forma, o uso de técnicas de EDM pode ser útil para se descobrir novos conhecimentos relevantes nesses dados.

Espera-se com a conclusão deste trabalho que os gestores do Ensino à Distância (EAD) de Instituições de Ensino Superior (IFES), possuam informações relevantes que permitam melhorar os processos de ensino/aprendizagem e os mecanismos de gestão acadêmica e pedagógica.

1.1 Relevância da Pesquisa

Um grande número de Instituições Educacionais vêm utilizando AVA's como apoio no processo de ensino e aprendizagem de estudantes, seja em cursos presenciais, semipresenciais ou totalmente à distância. Durante a utilização destes ambientes, os estudantes interagem com inúmeros recursos como *chats*, fóruns, documentos, vídeos entre outras ferramentas. Para Romero-Zaldivar et al. (2012) esta interação é identificada como um processo crucial do processo de aprendizagem, assumindo que uma experiência de aprendizagem bem-sucedida tem um alto nível de interação. Tais interações produzem uma enorme quantidade de dados que

podem ser analisados através da aplicação de técnicas de EDM.

Segundo Baker et al. (2012), a área emergente de EDM procura desenvolver ou adaptar métodos e algoritmos de mineração existentes, de tal modo que se prestem a compreender melhor os dados em contextos educacionais, produzidos principalmente por estudantes e professores, considerando os ambientes nos quais eles interagem, tais como AVA's, Sistemas Tutores Inteligentes (STI's), entre outros. Com tais métodos visa-se, por exemplo, entender melhor o estudante no seu processo de aprendizagem, analisando-se sua interação com o ambiente.

Alguns trabalhos, como Manhães (2015), Kampff et al. (2014), e Cambruzzi (2014), propõem sistemas de monitoramento que auxiliam os professores no acompanhamento e avaliação dos estudantes. Estes sistemas utilizam técnicas de EDM para a identificação de determinados perfis. Outros trabalhos, como Gotardo, Cereda e Junior (2013), Gottardo (2012) e Marquez-Vera, Romero e Ventura (2011), propõem o estudo e aplicação de métodos que possam melhorar o desempenho das técnicas de EDM utilizadas.

Este trabalho tem como proposta a aplicação de métodos que possam melhorar o desempenho das técnicas de EDM utilizadas na previsão do desempenho acadêmico de estudantes. Para isso, utilizam-se atributos que representam o comportamento destes estudantes no AVA, ou seja, a forma como interagem com as atividades e recursos disponibilizados nesta plataforma. Como contribuição, esta pesquisa apresenta a aplicação de técnicas e métodos ainda não utilizados em trabalhos correlatos analisados.

1.2 Objetivos

Buscando contribuir com uma solução capaz de auxiliar a tomada de decisão na gestão do Ensino a Distância, o objetivo geral deste trabalho visa à aplicação de técnicas de EDM na base de dados do AVA, utilizado em uma IFES, para prever o desempenho acadêmico de estudantes através de seus comportamentos durante a realização de um curso.

Dentro desse contexto, definem-se os seguintes objetivos específicos:

- Explorar e analisar os dados contidos na base de dados do AVA *Moodle* para identificar quais informações podem ser utilizadas na predição de desempenho;
- Utilizar técnicas de mineração de dados para gerar inferências que indiquem estimativas de desempenho acadêmico dos estudantes;
- Identificar quais os atributos e, consequentemente, quais recursos influenciam o desempenho acadêmico dos estudantes;
- Propor e aplicar um método de acompanhamento do desempenho acadêmico utilizando o conceito de séries temporais;

- Propor e aplicar um método de predição de desempenho acadêmico utilizando o conceito de classificadores em cascata;
- Apresentar, analisar e discutir os resultados obtidos nos experimentos realizados.

1.3 Metodologia

Segundo Charoux (2006), a pesquisa do tipo exploratória objetiva a formulação de diagnósticos sobre determinado fenômeno ou processo, procurando explorar todas as dimensões possíveis de um problema.

Dado o caráter exploratório deste trabalho, o mesmo teve início com um estudo mais aprofundando sobre as técnicas de EDM seguido da extração do conjunto de dados proveniente de um AVA *Moodle* e de um SGA. Este conjunto de dados foi processado e transformado. Após a preparação dos dados, foram realizados experimentos e os resultados obtidos são apresentados, analisados e discutidos.

1.4 Estrutura do Trabalho

O capítulo 2 apresenta alguns conceitos básicos de Aprendizado de Máquina, suas características e técnicas. Bem como são apresentados os conceitos de descoberta de conhecimento em base de dados e suas etapas.

No capítulo 3 são apresentados os conceitos de Mineração de Dados Educacionais, suas características. Alguns trabalhos relacionados à dissertação em questão são apresentados, descrevendo seus objetivos, suas características e os seus resultados obtidos.

O capítulo 4 apresenta as ferramentas, a descrição das Bases de Dados utilizada neste trabalho, as abordagens propostas, os experimentos realizados, bem como os resultados obtidos.

Por fim, o capítulo 5 traz as considerações finais sobre os resultados obtidos além de apresentar algumas considerações que apontam para trabalhos futuros.

2 Aprendizado de Máquina

Para Mitchell (1997), Aprendizagem de Máquina é o processo que faz com que um programa de computador melhore o seu desempenho em alguma tarefa através da experiência. Por exemplo, um programa de computador que aprende a jogar damas pode melhorar seu desempenho, medido pela sua capacidade de vencer jogos de damas, através da experiência obtida em jogos anteriores. Um problema de aprendizagem bem definido possui três características: a classe de tarefas, a medida de desempenho a ser melhorada, e a fonte de experiência. Assim pode-se definir Aprendizagem de Máquina:

"Um programa de computador aprende a partir da experiência E, em relação a uma classe de tarefas T, com medida de desempenho P, se seu desempenho em T, medido por P, melhora com a experiência E" (MITCHELL, 1997, p. 2).

Para Alpaydin (2010), Aprendizagem de Máquina é o desenvolvimento de programas de computadores capazes de otimizar o seu desempenho utilizando dados de exemplos ou de experiência anterior.

Witten, Frank e Hall (2011) define, operacionalmente, aprendizagem como aprender coisas que mudam seu comportamento de uma forma que melhorem seu desempenho no futuro. Assim, interpretam Aprendizagem de Máquina como a aquisição de descrições estruturais de exemplos.

Para Konar (1999), o Aprendizado de Máquina pode ser amplamente classificado em três categorias i) aprendizagem supervisionada, ii) aprendizagem não supervisionada e iii) aprendizagem por reforço. Estas categorias são brevemente descritas a seguir.

No aprendizado supervisionado os exemplos são rotulados. Os exemplos são ditos supervisionados pois, além de conter a entrada (vetor de atributos), possui também o rótulo ou saída (classe). Estes exemplos rotulados são fornecidos ao aprendiz, que constrói através do treinamento, uma função de mapeamento que tem como objetivo determinar corretamente a classe de novos exemplos não rotulados.

A Aprendizagem não supervisionada, no entanto, não utiliza treinador, pois os exemplos não são rotulados. Assim, o aprendiz tem que construir conceitos através da experimentação sobre o ambiente. Desta forma o aprendiz analisa os exemplos fornecidos e tenta determinar se alguns deles podem ser agrupados, através de características comuns (CHEESEMAN; STUTZ, 1996). Após a determinação dos agrupamentos, normalmente, é necessária uma análise para determinar o que cada agrupamento significa no contexto do problema que está sendo analisado.

Aprendizagem por reforço consiste em usar recompensas ou punições para aprender

uma política ótima (ou quase ótima) para o ambiente. Para isso, agentes inteligentes[1] são inseridos nesses ambientes com a capacidade de perceber e agir de modo autônomo, para atingirem objetivos ou executar tarefas para os quais tenham sido modelados (MAES, 1995). O agente precisa saber que algo de bom aconteceu quando ganhar, e que algo de ruim aconteceu quando perder (RUSSELL; NORVIG, 2004). Segundo Mitchell (1997), no aprendizado por reforço, cada vez que o agente executa uma ação em seu ambiente, um treinador pode fornecer uma recompensa ou penalidade para indicar a conveniência do estado resultante.

Para realizar a predição de desempenho dos estudantes nesta pesquisa se optou pelo aprendizado supervisionado, que será detalhado na próxima seção.

2.1 Aprendizado Supervisionado

Para Maimon e Rokach (2005), métodos supervisionados são métodos que tentam descobrir a relação entre a entrada de atributos (às vezes chamadas de variáveis independentes) e uma classe de destino (por vezes referida como uma variável dependente). A relação descoberta é representada em uma estrutura referida como um modelo. Normalmente, modelos descrevem e explicam os fenômenos que estão escondidos no conjunto de dados, e podem ser usados para prever o valor da classe alvo a partir do conhecimento dos valores dos atributos de entrada.

Segundo Han, Kamber e Pei (2011), na Aprendizagem Supervisionada, a supervisão no aprendizado vem dos exemplos rotulados no conjunto de dados de treinamento. Por exemplo, em um problema de concessão de crédito, um conjunto de características de clientes rotulados como inadimplentes ou não inadimplentes são utilizados como exemplos de treinamento, que supervisionam a aprendizagem do modelo de classificação.

Têm-se dois modelos de aprendizagem supervisionada, os modelos de classificação, onde a predição é feita para um atributo classificador que assume valores discretos, e os modelos de regressão, onde a variável alvo é contínua. O modelo de classificação será utilizado neste trabalho, pois se adapta melhor ao problema, e será detalhado na próxima seção.

O modelo de regressão, segundo Witten, Frank e Hall (2011), é um método simples para a previsão numérica, e tem sido amplamente utilizado em aplicações estatísticas por décadas, além de servir bem como blocos de construção para os métodos de aprendizagem mais complexos. A análise de regressão consiste na realização de uma análise estatística com o objetivo de verificar a existência de uma relação funcional entre uma variável dependente com uma ou mais variáveis independentes.

Alpaydin (2010) exemplifica a regressão da seguinte forma: assuma que busca-se ter um sistema que pode prever o preço de um carro usado. As entradas (X) são os atributos do carro, a marca, ano, cilindrada do motor, quilometragem, e outras informações que afetam o valor,

[1] Um agente inteligente é algo capaz de perceber seu ambiente através de sensores e de agir sobre esse ambiente por meio de atuadores (RUSSELL; NORVIG, 2004).

enquanto a saída (Y) é o preço do carro. Com as informações passadas forma-se o conjunto de dados de treinamento, e o programa de aprendizagem de máquina realiza uma análise estatística com o objetivo de saber Y em função de X.

2.1.1 Classificação

A classificação visa identificar a qual classe um novo exemplo pertence. Através de um conjunto de dados rotulados (classes), chamado conjunto de treinamento, um modelo de classificação é construído por um algoritmo. Quando um novo registro é submetido ao modelo, este é capaz de classificá-lo. Por exemplo, para conceder crédito, uma instituição financeira deve saber se o cliente é capaz ou não de honrar tal compromisso. O banco possui um histórico de clientes com dados como histórico financeiro, renda, reserva financeira, profissão, idade, dependentes, garantias e se pagaram ou não a dívida. Através desses dados passados é possível construir um modelo de classificação que informará em que perfil um novo cliente se encaixará.

Na classificação, o objetivo principal é a predição, uma vez que tem-se uma regra que se ajusta aos dados anteriores e, se o futuro é semelhante ao passado, então pode-se fazer previsões corretas para novos casos (ALPAYDIN, 2010).

Han, Kamber e Pei (2011) descrevem a abordagem geral para a classificação como um processo de duas etapas. No primeiro passo, é construído um modelo de classificação, com base em dados anteriores, através do conjunto de treinamento. Na segunda etapa, através do conjunto de teste, determina-se se a precisão do modelo é aceitável, e se assim for, o modelo é usado para classificar novos dados. Este processo pode ser observado na Figura 1.

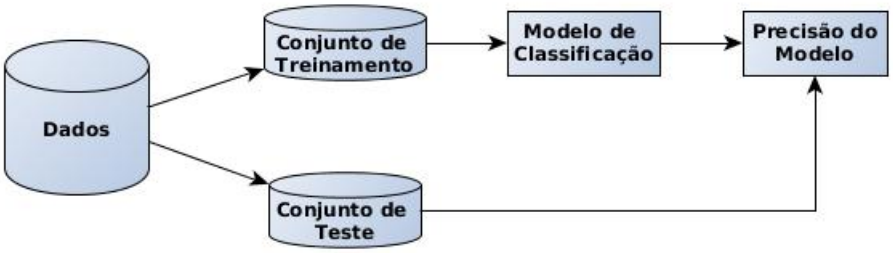

Figura 1 – Processo básico de classificação

Fonte: Adaptado de Han, Kamber e Pei (2011)

De forma geral, a ideia da classificação é colocar um objeto em uma classe ou categoria, com base nas suas características. Na educação, professores e instrutores estão a todo tempo classificando seus alunos de acordo com o seu conhecimento, motivação e comportamento. Avaliar as respostas de uma prova e determinar uma pontuação de acordo com certos critérios também é uma tarefa de classificação. Classificação automática é uma parte inevitável de Sistemas Tutores Inteligentes (STI's), que são sistemas instrucionais baseados em computador com modelos de conteúdo instrucional que especificam 'o que' ensinar, e estratégias de

ensino que especificam ‘como’ ensinar (WENGER, 1987). Antes do sistema selecionar qualquer medida de adaptação como a seleção de tarefas, material de aprendizagem, ou conselho, este deve primeiro classificar a situação atual do aluno. Para este fim, precisa-se de um classificador, um modelo, que prevê o valor de classe de acordo com atributos, características. Por exemplo, pode-se derivar o nível de motivação do estudante a partir de suas ações no sistema de tutoria, bem como prever o seu desempenho. Tais previsões são igualmente úteis no ensino tradicional, mas em sistemas de aprendizagem informatizados pode-se recolher mais dados para derivar classificadores (HÄMÄLÄINEN; VINNI, 2010).

Algumas das técnicas utilizadas na tarefa de classificação são Árvores de Decisão (WITTEN; FRANK; HALL, 2011), Redes Neurais (WITTEN; FRANK; HALL, 2011), Regras de Classificação (HAN; KAMBER; PEI, 2011), Aprendizado Bayesiano (HAN; KAMBER; PEI, 2011), Máquinas de Vetores (MAIMON; ROKACH, 2005) e Aprendizado Tardio (WITTEN; FRANK; HALL, 2011). A seguir, cada técnica será brevemente descrita.

Árvores de Decisão (WITTEN; FRANK; HALL, 2011) são um fluxograma com uma estrutura de árvore, onde cada nó interno (não nó folha) indica um ensaio de um atributo, cada ramo representa um resultado de teste, e cada nó folha (ou nó terminal) contém um rótulo de classe. O nó superior em uma árvore é o nó raiz. A Figura 2 exemplifica a árvore de decisão gerada a partir do conjunto de dados fictícios da Tabela 1.

Tabela 1 – Conjunto de dados fictícios para concessão de empréstimos

Cliente	**Renda**	**Reserva Financeira**	**Liberar Empréstimo**
1	Alta	Possui	Liberar
2	Baixa	Não possui	Não liberar
3	Alta	Possui	Liberar
4	Alta	Não possui	Não liberar
5	Baixa	Possui	Não liberar

A Tabela 1 apresenta os dados de clientes, de uma insituição financeira, que desejam ter os seus empréstimos liberados. A coluna Cliente identifica cada cliente da instituição, as colunas Renda e Reserva Fincanceira são os atributos (as características dos clientes), e por fim, a coluna Liberar Empréstimo representa a classe, o rótulo, que assume os valores discretos *Liberar* e *Não liberar*.

Em uma árvore de decisão, o conhecimento é representado em cada nó que, ao ser testado, pode conduzir a busca a um de seus filhos. Na Figura 2, no nível 1 da árvore tem-se o atributo ‘Renda’ de onde partem dois ramos, que representam cada um dos valores que o atributo Renda pode assumir. Caso o valor seja *Baixa* tem-se o rótulo da classe *Não Liberar*, pois independente do valor do atributo ‘Reserva Financeira’ a ordem é não liberar empréstimos para os clientes de baixa renda. No outro ramo, onde se tem o valor *Alta*, parte-se para o nível 2 da árvore onde o atributo ‘Reserva Financeira’ é testado. Ao assumir o valor *Possui* tem-se o rótulo da classe *Liberar*, caso este último assuma o valor *Não possui*, o rótulo é *Não liberar*.

Figura 2 – Árvore de decisão

Fonte: Elaborado pelo autor

As Redes Neurais (WITTEN; FRANK; HALL, 2011) são modelos de computação para o processamento de informações e são particularmente úteis para a identificação da relação fundamental entre um conjunto de variáveis ou padrões nos dados. Elas se desenvolveram a partir da pesquisa em Inteligência Artificial, especificamente, a partir das tentativas para imitar a aprendizagem das Redes Neurais biológicas - especialmente aquelas no cérebro humano, que podem conter mais do que 10^{11} neurônios altamente interligados. As Redes Neurais são capazes de resolver os problemas que têm padrões imprecisos ou dados contendo informações incompletas e ruidosas com um grande número de variáveis. Esta característica de tolerância a falhas é atraente para os problemas de Mineração de Dados, pois os conjuntos de dados são geralmente ruidosos e não seguem claras estruturas de probabilidade que tipicamente são requeridas pelos modelos estatísticos (MAIMON; ROKACH, 2005).

Regras de Classificação (HAN; KAMBER; PEI, 2011) são uma alternativa popular para Árvores de Decisão, e se baseiam na estrutura: *SE* condição *ENTÃO* conclusão. O antecedente, ou pré-condição de uma regra é uma série de ensaios tal como os testes nos nós em Árvores de Decisão, enquanto que o consequente, ou conclusão, fornecem à classe ou classes que se aplicam a casos abrangidos por essa regra (WITTEN; FRANK; HALL, 2011). Exemplos de regras geradas para o conjunto de dados fictícios da Tabela 1 são:

Regra 1: SE(Renda = ‘Alta’ E Reserva Financeira = ‘Possui’) ENTÃO ‘Liberar’;
Regra 2: SE(Renda = ‘Alta’ E Reserva Financeira = ‘Não possui’) ENTÃO ‘Não liberar’;
Regra 3: SE(Renda = ‘Baixa’) ENTÃO ‘Não liberar’.

O Aprendizado Bayesiano (HAN; KAMBER; PEI, 2011) fornece uma abordagem probabilística para a inferência baseada no Teorema de *Bayes* (MITCHELL, 1997). Segundo o Teorema de *Bayes*, é possível encontrar a probabilidade de certo evento (h) ocorrer, dada a probabilidade de outro evento (D) que já ocorreu:

$$P(h|D) = \frac{P(D|h) * P(h)}{P(D)} \tag{2.1}$$

Onde:

- P(h|D) probabilidade a posteriori de h condicional a D;
- P(D|h) probabilidade a posteriori de D condicional a h;
- P(h) probabilidade a priori do evento h ocorrer;
- P(D) probabilidade a priori do evento D ocorrer.

Os algoritmos Bayesianos são importantes para a Aprendizagem de Máquina porque calculam explicitamente as probabilidades para as hipóteses, além de fornecer base para outros algoritmos que não manipulam explicitamente probabilidades (MITCHELL, 1997).

SVM (*Support Vector Machines*) (MAIMON; ROKACH, 2005) proporcionam um método de aprendizagem que tem sido altamente bem-sucedido em uma variedade de aplicações. A origem do SVMs surge de duas ideias-chave. A primeira idéia é realizar um mapeamento não linear do conjunto de dados em um espaço de dimensão mais elevada, e, em seguida, utilizar classificadores lineares neste novo espaço.

Aprendizado Tardio (*Lazy Learning*) (WITTEN; FRANK; HALL, 2011) é o método de aprendizagem que adia o processamento de dados de treinamento até que haja uma necessidade de consulta a ser respondida. Isto normalmente envolve armazenar os dados na memória, e encontrar dados relevantes na base de dados para responder a uma consulta particular. O típico exemplo deste método é a técnica dos Vizinhos mais Próximo (*k-nearest neighbor* - KNN) que possui três elementos principais: um conjunto de exemplos rotulados (por exemplo, um conjunto de registros armazenados), uma distância ou semelhança métrica para calcular a distância entre exemplos, e o valor de k, o número de vizinhos mais próximos. Para classificar um exemplo não marcado, os seus k vizinhos mais próximos são identificados, e o rótulo de classe predominante desses vizinhos é então usado para determinar a classe deste objeto. Um exemplo simples de como funciona a classificação realizada pelo algoritmo KNN pode ser observado na Figura 3.

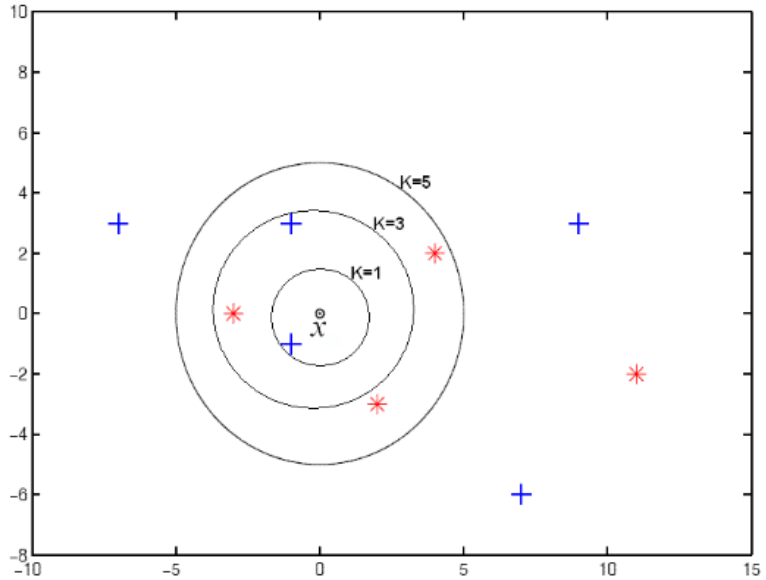

Figura 3 – Exemplo simples do funcionamento do algoritmo KNN

Fonte: Elaborado pelo autor

A tarefa do KNN é classificar o padrão X como sendo pertencente a uma das classes (+ ou *) exibidas no exemplo. Dependendo do número de K-vizinhos, X será classificado da seguinte forma: se K=1, X será classificado como "+", se K=3, X será classificado como "+", e se K=5, X será classificado como "*".

2.1.2 Métricas de Avaliação dos Classificadores

Para avaliar o desempenho dos algoritmos classificadores se faz necessário o uso de métricas de avaliação para verificar se o resultado da predição é satisfatório. Algumas destas métricas são apresentadas a seguir.

A Matriz de Confusão pode ser usada como ponto de partida para uma série de técnicas estatísticas descritivas e analíticas. Ela contém o número de elementos que foram corretamente ou incorretamente classificadas para cada classe. Na Tabela 2 pode-se observar, na diagonal em destaque, o número de exemplos que foram corretamente classificados para cada classe, enquanto os elementos fora da diagonal indicam o número de exemplos que foram classificados incorretamente (MAIMON; ROKACH, 2005).

A Matriz de Confusão é uma ferramenta útil para analisar o quão bem o seu classificador pode reconhecer exemplos de diferentes classes. Dada n classes (onde $n >= 2$), uma matriz de confusão é uma tabela de (ao menos) tamanho n por n (HAN; KAMBER; PEI, 2011). Em problemas com duas classes, uma das classes é referida como positiva, enquanto a outra é chamada de negativa.

Tabela 2 – Matriz de Confusão para problemas com duas classes

	Predição Positiva	**Predição Negativa**
Classe Positiva	Verdadeiro Positivo (TP)	Falso Negativo (FN)
Classe Negativa	Falso Positivo (FP)	Verdadeiro Negativo (TN)

Para realizar o cálculo de algumas métricas de avaliação, se faz necessário compreender alguns termos utilizados (HAN; KAMBER; PEI, 2011).

- Verdadeiros positivos (*True Positive - TP*): referem-se aos exemplos positivos que foram corretamente rotulados pelo classificador. Logo *TP* é o número de verdadeiros positivos.
- Verdadeiros negativos (*True Negative - TN*): referem-se aos exemplos negativos que foram corretamente rotulados pelo classificador. Logo *TN* é o número de verdadeiros negativos.
- Falsos positivos (*False Positive - FP*): referem-se aos exemplos negativos que foram incorretamente rotulados como exemplos positivos pelo classificador. Logo *FP* é o número de falsos positivos.
- Falsos negativos (*False Negative - FN*): referem-se aos exemplos positivos que foram incorretamente rotulados como exemplos negativos pelo classificador. Logo *FN* é o número de falsos negativos.

Acurácia é a porcentagem de exemplos do conjunto de teste que foram classificados corretamente, ou seja, é a taxa de acerto global.

$$Acurácia = \frac{TP + TN}{P + N} \tag{2.2}$$

Error Rate é a taxa de erro ou taxa de classificação incorreta de um classificador, que é simplesmente 1 - *Acurácia*.

$$Error\ Rate = \frac{FP + FN}{P + N} \tag{2.3}$$

Precision é a porcentagem de exemplos classificados corretamente como positivos (*TP*) dentre todos os que foram classificados como positivos (*TP* + *FP*).

$$Precision = \frac{TP}{TP + FP} \tag{2.4}$$

Recall é a porcentagem de exemplos classificados corretamente como positivos (*TP*) dentre todos os que realmente são positivos (*TP* + *FN*).

$$Recall = \frac{TP}{TP+FN} = \frac{TP}{P} \tag{2.5}$$

F-Measure é a média harmônica de *Precision* e *Recall*, onde o F tem seu melhor valor em 1 e o pior em 0.

$$F = \frac{2 * Precision * Recall}{Precision + Recall} \tag{2.6}$$

Neste trabalho, utilizou-se a Acurácia para medir a taxa de acerto global. Tal métrica foi utilizada em alguns trabalhos relacionados, tais como Zafra, Romero e Ventura (2011), Marquez-Vera, Romero e Ventura (2011), Manhães (2015). Utilizou-se também a *F-Measure* para avaliar o desempenho dos classificadores em cada classe dos conjuntos de dados, fato que se justifica devido ao desbalanceamento de classes identificado em um dos conjuntos utilizados. Esta métrica foi utilizada em alguns trabalhos relacionados tais como Dekker, Pechenizkiy e Vleeshouwers (2009), Talebi e Sayficar (2014), e Luo et al. (2015).

2.1.3 Conjunto de Treinamento e Teste

Para a realização da tarefa de classificação e predição de desempenho, é necessário que o algoritmo utilizado na construção do modelo de aprendizado seja treinado. Para realizar este treinamento, e posterior teste, é necessário que o conjunto de dados seja dividido em dois conjuntos distintos, treinamento e teste. Neste trabalho foi utilizado o método *cross validation*, que funciona da seguinte forma: o conjunto de dados é dividido em *n* partes (*folds*) iguais, geralmente 10. Cada uma destas n partes é utilizada para teste e o restante para treinamento, ou seja, utiliza-se 9/10 dos dados para treinamento e 1/10 dos dados para o teste. Este procedimento se repete 10 vezes, de modo que, ao final, cada parte seja utilizada apenas uma vez para testes (HAN; KAMBER; PEI, 2011). Outra forma de divisão é a realizada pela funcionalidade *Percentage split* presente na ferramenta *Weka*, onde a porcentagem informada representa o conjunto de treinamento.

2.1.4 Seleção de Atributos

O objetivo da Seleção de Atributos é encontrar um subconjunto de características[2] de um conjunto de dados original, de modo que um algoritmo de indução que seja executado em dados contendo apenas esses atributos gere um classificador com a maior precisão possível. São duas as abordagens principais Filtro (*Filter*) e Cápsula (*Wrapper*) (KOHAVI; SOMMERFIELD; DOUGHERTY, 1996).

A abordagem Filtro, Figura 4, introduz um processo separado, o qual ocorre antes da aplicação do algoritmo de aprendizagem (indução) propriamente dito. A ideia é filtrar atrib-

[2] Os termos atributos (*features*) e características são usados neste trabalho de maneira intercambiável.

utos irrelevantes, segundo algum critério, antes do aprendizado ocorrer. Essa etapa do pré-processamento considera características gerais do conjunto de dados para selecionar alguns atributos e excluir outros. Assim, métodos de filtro são independentes do algoritmo de aprendizado que, simplesmente, recebem um conjunto de exemplos e selecionam um subconjunto de atributos considerados importantes pelo Filtro (FREITAS, 1998).

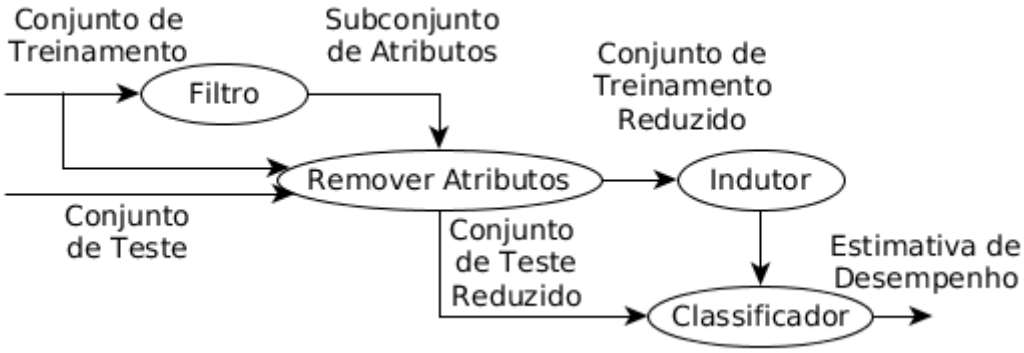

Figura 4 – Seleção de atributos pela abordagem Filtro (*Filter*)

Fonte: Adaptado de Freitas (1998)

Na abordagem Cápsula (*Wrapper*), Figura 5, a seleção do subconjunto de características é realizada usando a capacidade preditiva do algoritmo de indução como uma caixa-preta (ou seja, não é necessário nenhum conhecimento do algoritmo, apenas a interface). O algoritmo de Seleção de Atributos realiza uma pesquisa dinâmica por um bom subconjunto de atributos 'candidatos' usando o algoritmo de indução para avaliar a qualidade do subconjunto atualmente selecionado (KOHAVI; SOMMERFIELD; DOUGHERTY, 1996).

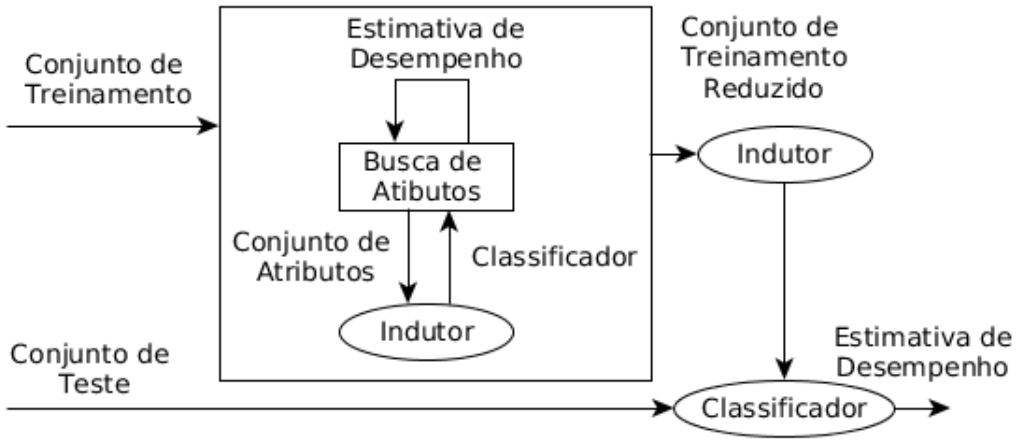

Figura 5 – Seleção de atributos pela abordagem Cápsula (*Wrapper*)

Fonte: Adaptado de Freitas (1998)

2.1.5 Balanceamento de Classes

Alguns aspectos podem influenciar o desempenho de um modelo de classificação criado por um sistema de aprendizado supervisionado. Um desses aspectos é o desbalanceamento de classes, definido por Chawla et al. (2002) como um conjunto de dados onde as classes não estão igualmente representadas. Quando essa diferença é grande, os sistemas de aprendizado tendem a valorizar classes predominantes em detrimento à classe minoritária. Em muitos dos problemas reais há uma grande desproporção no número de casos pertencentes a cada uma das classes, como, por exemplo, detecção de fraudes em chamadas telefônicas, transações com cartão de créditos, gestão de telecomunicações e classificação de textos (CHAWLA et al., 2002). Alguns métodos podem ser aplicados a conjuntos de dados para solucionar o problema de classes desbalanceadas. Existem basicamente duas classes de métodos para o balanceamento da distribuição das classes, a saber:

- *Under-sampling* que tem como objetivo balancear o conjunto de dados pela eliminação de exemplos da classe majoritária;
- *Over-sampling* que replicam exemplos da classe minoritária com o objetivo de obter uma distribuição mais balanceada.

Ambos os métodos têm problemas correlacionados. *Under-sampling* pode eliminar exemplos potencialmente úteis nos dados e *over-sampling* pode aumentar a probabilidade de ocorrer um *overfitting*[3], uma vez que a maioria dos métodos de *over-sampling* fazem cópias exatas dos exemplos da classe minoritária (PRATI et al., 2003).

A técnica escolhida para ser utilizada neste trabalho foi a *SMOTE* (*Synthetic Minority Over-sampling Technique*) (CHAWLA et al., 2002) que é definida por Witten, Frank e Hall (2011) como um filtro de dados supervisionado que altera a distribuição de uma classe. Basicamente, a técnica consiste em inserir instâncias sintéticas usando a abordagem de k-vizinho mais próximo. O usuário pode informar qual classe deseja inserir novas instâncias, o percentual de sobre amostragem e o número de vizinhos. O fato de realizar a inserção de instâncias pela interpolação de exemplos da classe minoritária que estão próximos contorna a possibilidade de *overfiting*.

2.2 Aplicações do Aprendizado de Máquina

Áreas como, astronomia, marketing, financeiro, detecção de fraudes, produção industrial, e telecomunicações trabalham há algum tempo com a ideia de descobrir padrões em dados

[3] Quando o modelo estatístico se adapta demasiadamente ao conjunto de dados ou amostra de treinamento. É comum que a amostra apresente desvios causados por erros de medição ou fatores aleatórios, ocorre o *overfitting* quando o modelo se ajusta a estes. Um modelo com *overfitting* apresenta alta precisão quando testado com seu conjunto de dados, porém tal modelo não é uma boa representação da realidade.

de forma automática e utilizá-los para predizer eventos (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996).

Além das aplicações citadas anteriormente, a aprendizagem de máquina é utilizada na área educacional, como podemos observar nos trabalhos a seguir. Utilizando mineração de dados, Kampff (2009) identifica comportamentos e características de alunos com risco de evasão ou reprovação e, então, alerta o professor. Na *Utah State University*, Huang (2011) utiliza modelos matemáticos, mineração de dados, para prever o desempenho de estudantes na disciplina *Engineering Dynamics*. Nandeshwar, Menzies e Nelson (2011) utilizam dados socioeconômicos e dados do ensino médio para prever a retenção de alunos na graduação. Cambruzzi (2014) e Portal (2016) utilizam mineração de dados e *Learning Analytics* para acompanhamento de alunos com o objetivo principal de reduzir a evasão na EAD. Fonseca e Namen (2016) aplicam mineração em bases de dados do INEP (Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira) com o intuito de identificar fatores que relacionam o perfil de professores que lecionam Matemática com a proficiência obtida por seus alunos.

Para se descobrir conhecimento em grandes bases de dados, deve-se identificar informações relevantes, processá-las e agregá-las ao conhecimento prévio do domínio com o qual se trabalha para resolver determinada situação ou problema (WIVES, 2002). O processo de obtenção desta informação é fortemente ligado a como esta enorme quantidade de dados são processados, logo mecanismos automáticos tornam estes processos mais eficientes. Neste contexto surge a área de pesquisa conhecida como Descoberta de Conhecimento em Base de Dados (GOLDSCHMIDT; PASSOS, 2005), que aplica técnicas de Aprendizagem de Máquina em grandes bases de dados, e será descrita a seguir.

2.3 Descoberta de Conhecimento em Base de Dados

Todos os dias milhares de pessoas, através de suas atividades cotidianas disponibilizam uma quantidade enorme de dados seja em atividades comerciais, de lazer ou educacionais, impulsionados, principalmente, pelos recursos tecnológicos. Esta enorme quantidade de dados pode ser explorada para a geração de conhecimento útil para suporte à tomada de decisão através da aplicação do processo de descoberta de conhecimento em base de dados (do inglês *Knowledge Discovery in Databases* – KDD).

O processo de KDD preocupa-se com o desenvolvimento de métodos e técnicas para dar sentido aos dados. O núcleo do processo é a aplicação de métodos de mineração de dados específicos para a descoberta e extração de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996).

Segundo Han, Kamber e Pei (2011), o processo de KDD é dividido em sete etapas: limpeza dos dados, integração dos dados, seleção dos dados, transformação dos dados, miner-

ação dos dados, avaliação dos modelos encontrados e apresentação do conhecimento adquirido.

2.3.1 Etapas do Processo de KDD

Na Figura 6, podemos observar as etapas do processo de KDD e sob um nível mais alto de abstração este processo pode ser dividido em 3 etapas, pré-processamento, mineração de dados e pós-processamento.

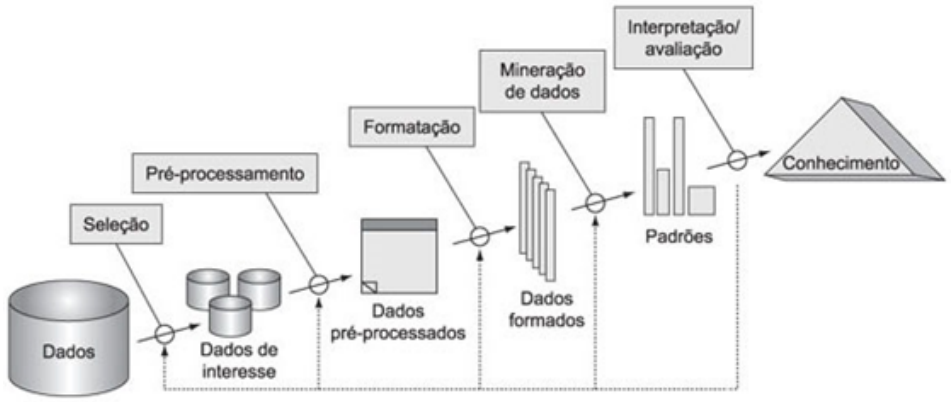

Figura 6 – Etapas do processo de KDD

Fonte: Adaptado de Fayyad, Piatetsky-Shapiro e Smyth (1996)

2.3.1.1 Pré-Processamento

Esta etapa engloba as tarefas de seleção, pré-processamento e transformação. Nesta etapa o primeiro passo é desenvolver um entendimento do domínio de aplicação, identificar o objetivo e o que é importante para o cliente (o que se pretende descobrir).

Em segundo lugar devemos selecionar os dados, que podem ser originados de diversas fontes, criar uma base de dados de destino ou *Data Warehouse* (DW) que é um repositório de informações coletadas de várias fontes, armazenado sob um esquema unificado Han, Kamber e Pei (2011).

O terceiro passo é o pré-processamento e a limpeza dos dados que geralmente envolvem filtrar, combinar e preencher valores vazios. Além de detectar ruídos (é um erro ou variância aleatória na medição de uma variável) e *outiliers* (valores extremos que estão fora dos limites de um intervalo de dados ou estão destoando da tendência dos valores de um determinado atributo).

O quarto passo é a transformação onde os dados precisam ser modificados ou consolidados, ou seja, passam pelo processo de transformação para que estejam em uma forma adequada e possam ser melhores entendidos pelos algoritmos de mineração. Por exemplo, transformar o atributo nota (que possuem valores numéricos contínuos de 0 a 100) em atributos categóricos (Aprovado ou Reprovado).

2.3.1.2 Mineração de Dados

A Mineração de Dados (do inglês *Data Mining* - DM) é a área específica do KDD que trata das técnicas e algoritmos utilizados na detecção dos padrões de dados.

São várias as definições sobre DM encontradas na literatura. Segundo Cardoso e Machado (2008), mineração de dados faz parte de uma das etapas da descoberta de conhecimento em bancos de dados. Ela é capaz de revelar o conhecimento que está implícito em grandes quantidades de informações armazenadas nas bases de dados de uma organização. DM é a análise de grandes conjuntos de dados a fim de encontrar relacionamentos inesperados e de resumir os dados de uma forma que eles sejam tanto úteis quanto compreensíveis ao dono dos dados (HAND; MANNILA; SMYTH, 2001). Mineração de dados segundo Witten, Frank e Hall (2011) é usada para extrair informações a partir dos dados brutos em bases de dados de informação que é expresso numa forma compreensível e pode ser usado para uma variedade de fins.

Alguns autores (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996; MAIMON; ROKACH, 2005) dividem está etapa em 3, escolha da tarefa de mineração, escolha dos algoritmos e a aplicação propriamente dita dos algoritmos em busca dos padrões nos dados.

2.3.1.3 Pós-Processamento

O objetivo nesta etapa é a avaliação, interpretação e explanação do conhecimento extraído da base de dados. A avaliação é realizada através de medidas técnicas como a acurácia, precisão, matriz de confusão, *F-Measure* dentre outros, onde se procura obter a confiabilidade nos modelos gerados e padrões extraídos. Se as medidas técnicas não forem boas o suficiente durante a avaliação, pode ser necessário o retorno a etapas anteriores do processo de KDD. A interpretação é a fase de visualização dos dados extraídos, da análise do modelo descoberto, onde os padrões identificados pelo sistema são interpretados em conhecimento. É durante a explanação que o conhecimento extraído deve se tornar compreensível ao usuário, ou seja, é nesta fase que há o detalhamento dos resultados obtidos, seja em forma de relatórios, gráficos ou documentos, visando facilitar o entendimento do cliente, do interessado na descoberta.

3 Mineração de Dados Educacionais

Com a expansão dos cursos mediados por meios computacionais, muitos pesquisadores na área da Informática na Educação (IE) têm utilizado DM para explorar dados originados em ambientes educacionais, com o objetivo de encontrar padrões e contribuir com os processos de aprendizagem e, consequente, melhora das ações pedagógicas e de materiais didáticos. Dentro deste contexto, surgiu a área de pesquisa conhecida como EDM. Segundo Baker, Isotani e Carvalho (2011), EDM é definida como a área de pesquisa que tem como principal foco o desenvolvimento de métodos para explorar conjuntos de dados coletados em ambientes educacionais. Para Romero e Ventura (2010), EDM lida com o desenvolvimento de métodos para explorar os dados originários de um contexto educacional. Assim, é possível compreender de forma mais eficaz e adequada os alunos, como eles aprendem, o papel do contexto na qual a aprendizagem ocorre, além de outros fatores que influenciam a aprendizagem. Segundo Romero e Ventura (2013), EDM é a combinação de três áreas, ciência da computação, estatística e educação, conforme pode ser observado na Figura 7.

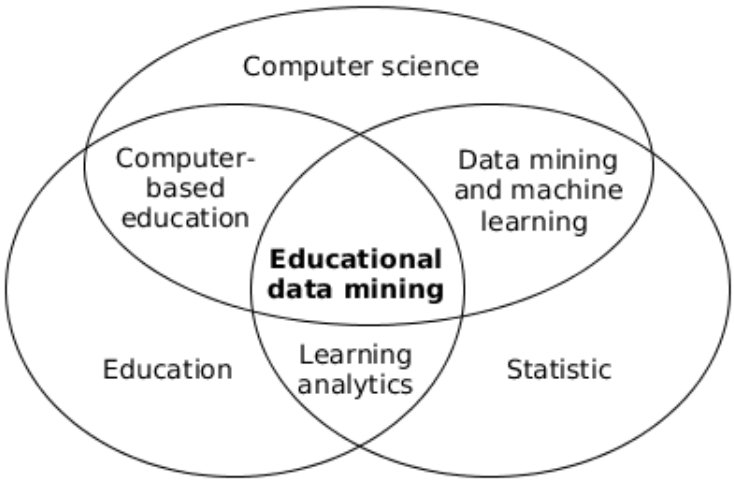

Figura 7 – Três principais áreas relacionadas com a EDM

Fonte: Romero e Ventura (2013)

O processo de EDM converte os dados brutos de Sistemas Educacionais em informação útil que pode ser usada por desenvolvedores de *software*, professores, pesquisadores educacionais, etc. Para Romero, Ventura e Bra (2004), a EDM não difere do típico processo de descoberta de conhecimento, que é mostrado na Figura 8.

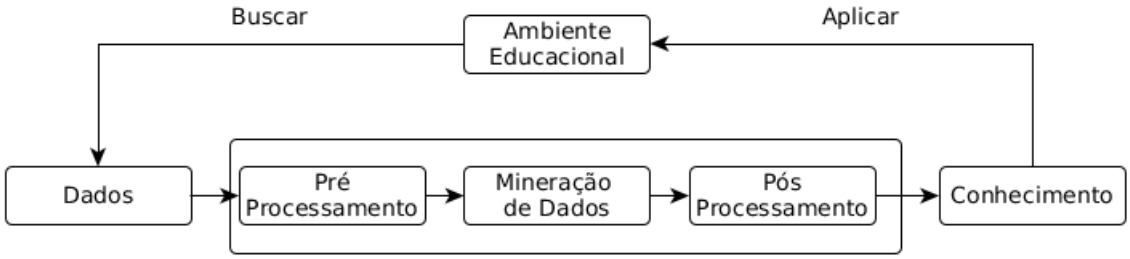

Figura 8 – Etapas da Mineração de Dados Educacionais

Fonte: Adaptado de García et al. (2011)

Segundo Romero e Ventura (2013), as origens da EDM estão em uma série de oficinas organizadas em conferências relacionadas, que começou em 2000. A primeira oficina, denominada '*Educational Data Mining*', ocorreu em 2005 e culminou em 2008 com a criação da Conferência Internacional Anual sobre *Education Data Mining* organizada pelo *International Working Group on Educational Data Mining*. A primeira conferência realizada foi a EDM2008 em Montreal no Canadá, desde então vem sendo realizada de forma anual.

Existem várias linhas de pesquisa na área de EDM. Muitas delas derivadas diretamente da área de mineração de dados. Baker, Isotani e Carvalho (2011) apresentam a taxonomia das principais subáreas de pesquisa em EDM:

1. Predição
    - Classificação
    - Regressão
    - Estimação de Densidade
2. Agrupamento
3. Mineração de Relações
    - Mineração de Regras de associação
    - Mineração de Correlações
    - Mineração de Padrões Sequenciais
    - Mineração de Causas
4. Destilação de dados
5. Descobrimento com modelos

Na predição o que se busca é aprender critérios de decisão para classificar casos desconhecidos e prever o que poderá ocorrer, a meta é desenvolver modelos que deduzam aspectos

específicos dos dados. A análise preditiva é capaz de descobrir padrões ocultos nos dados que o especialista humano não pode ver.

No agrupamento, o objetivo principal é achar dados que se agrupam naturalmente, organizando os dados em grupos de acordo com alguma medida de similaridade. Os padrões pertencentes a um dado grupo são mais similares entre si do que em relação a padrões pertencentes a outro grupo.

A mineração de relações tem como objetivo descobrir possíveis relações entre variáveis em uma base de dados com muitas variáveis. Estas relações são identificadas descobrindo quais variáveis estão fortemente conectadas.

Na destilação de dados são realizadas pesquisas que têm como objetivo apresentar dados complexos de forma mais simples e de fácil compreensão e que possam expor suas características mais importantes (BAKER; ISOTANI; CARVALHO, 2011).

Em descoberta com modelos, parte-se de um modelo gerado por um método de predição, tal como classificação, ou por um método de agrupamento, ou ainda manualmente, por meio de engenharia de conhecimento. Em seguida, esse modelo é utilizado como componente, ou ponto de partida, em outra análise com técnicas de predição ou mineração de relações (BAKER et al., 2012).

A comunidade de EDM vem crescendo rapidamente, fato que pode ser verificado nos trabalhos listados a seguir. Romero e Ventura (2010) conceituam como uma área de investigação interdisciplinar emergente. Baker, Isotani e Carvalho (2011) apresentaram um breve histórico do surgimento desta comunidade, e teve como objetivo divulgar esta área no Brasil. Baker et al. (2012) apresentam os principais conceitos, técnicas e algoritmos de mineração de dados aplicados a conjuntos de dados educacionais. Peña-Ayala (2014) afirma que a EDM emerge como um paradigma orientado para projetar modelos, tarefas, métodos e algoritmos para explorar dados de ambientes educacionais.

3.1 Trabalhos Relacionados

Nesta seção são apresentados alguns trabalhos correlatos estudados durante a realização deste trabalho. Os trabalhos foram analisados e separados em categorias de acordo com seu objetivo.

3.1.1 Comparação de Algoritmos

Manhães (2015) propõe uma arquitetura chamada EDM WAVE que engloba todo o processo de descoberta de conhecimento em dados (pré-processamento, mineração de dados e pós-processamento) para predição do desempenho acadêmico de graduandos. O objetivo do trabalho é fornecer aos gestores educacionais das Universidades públicas brasileiras, não especialistas

em EDM, uma abordagem que oferece informações úteis sobre o desempenho acadêmico dos graduandos e predizer os que estão em risco de abandonar o sistema de ensino. Em seu estudo de caso 1 a autora realiza uma avaliação de 12 algoritmos classificadores (*AdaBoost*, *Bayes Net*, *Decision Table*, J48, JRip, *Multilayer Perceptron*, *Naive Bayes*, OneR, *Random Forest*, *Simple Logistic*, SVM (*PolyKernel*) e SVM (*RBFKernel*)), todos disponíveis na ferramenta *Weka*. Sendo o melhor desempenho alcançado pelo algoritmo OneR.

Zafra, Romero e Ventura (2011) propõem uma nova abordagem baseada na aprendizagem de instâncias múltiplas (do inglês, *Multiple Instance Learning* - MIL) para prever o desempenho acadêmico de estudantes em Ambiente Virtual de Aprendizagem (AVA) e melhorar os resultados obtidos usando a abordagem clássica. Segundo os autores, os resultados obtidos na *Cordoba University*, utilizando a abordagem MIL, apresentaram melhores soluções com uma confiança de 95%, de acordo com o Teste de *Wilcoxon*.

Kotsiantis, Pierrakeas e Pintelas (2003) tem como objetivo realizar um estudo sobre o uso de técnicas de Aprendizado de Máquina para lidar com a desistência estudantil e posterior identificação do algoritmo de aprendizado mais adequado para a previsão de evasão de estudantes no EAD. Foram utilizados dados de estudantes provenientes do curso de Informática da *Hellenic Open University*, os algoritmos utilizados foram *Naive Bayes*, C4.5, *Back Propagation*, MLE (*Maximum Likelihood Estimation*), SVM (*Support Vector Machine*) e 3NN, onde o autor chegou a conclusão que o *Naive Bayes* foi mais adequado para a aplicação.

Assim como os trabalhos citados anteriormente, esta dissertação realiza uma comparação de algoritmos classificadores com o objetivo de avaliar qual classificador melhor se adapta ao problema de previsão de desempenho de estudantes no EAD. Além de identificar o algortimo que melhor se adapta ao problema, os resultados obtidos com tal comparação servem como base para avalição de novos métodos de predição propostos.

3.1.2 Balanceamento de Classes

Com o objetivo geral de investigar a possibilidade de geração de inferências relativas ao desempenho futuro de estudantes, Gottardo (2012) utiliza técnicas de mineração de dados utilizando atributos disponíveis em uma base de dados do AVA *Moodle*. Em seu quarto experimento, o autor utiliza o algoritmo de balanceamento de classes SMOTE disponível na ferramenta *Weka*. O conjunto de dados balanceado foi submetido aos algoritmos *Multilayer Perceptron* e *Random Forest*, os resultados mostraram um aumento considerável no desempenho da classe minoritária balanceada.

3.1.3 Seleção de Atributos

Ferreira (2015) tem como foco a identificação de fatores relacionados à conclusão do Ensino Fundamental utilizando técnicas de mineração de dados aplicadas aos Micro-dados

do Censo Escolar da Educação Básica coletados pelo INEP (Instituto Nacional de Estudos e Pesquisas Educacionais) no ano de 2014. Foi aplicado o algoritmo de classificação J48 e o filtro *CfsSubsetEval*, ambos da ferramenta *Weka* (WAIKATO, 2016). Os resultados apresentaram uma redução no desempenho do classificador, quando utilizada à técnica de filtro. Porém, a Árvore de Decisão gerada foi menor. Esta menor quantidade de nodos permite uma melhor visualização das regras obtidas. Algumas regras que associaram internet banda larga, laboratório de ciências, auditório na escola e ensino privado, como os fatores mais influentes na conclusão do Ensino Fundamental.

Com o objetivo de prever o desempenho de estudantes na Universidade de Calcutá, Acharya e Sinha (2014) utilizam a técnica de Seleção de Atributos para elevar o desempenho dos classificadores e criar modelos com menor complexidade computacional. Os autores utilizam as técnicas de Filtro e Cápsula, onde os algoritmos *Correlation Based FS*, *Chi-Square Based Feature Evaluation*, *Information Gain Attribute Evaluation* são baseados no método Filtro, e os algoritmos C 4.5, *Naive Bayes*, e 1-NN são utilizados na técnica em Cápsula. Foram utilizados os dados socioeconômicos de 309 estudantes com um total de 14 atributos. Os melhores resultados foram alcançados com o conjunto de dados selecionado pelo algoritmo de filtro *Correlation Based FS* com 8 atributos.

Com o objetivo geral de investigar a possibilidade de geração de inferências relativas ao desempenho futuro de estudantes, Gottardo (2012) utiliza técnicas de mineração de dados utilizando atributos disponíveis em uma base de dados do AVA *Moodle*. Em seu terceiro experimento, o autor utiliza o algoritmo de Seleção de Atributos supervisionado padrão da ferramenta *Weka*. Estes atributos selecionados foram submetidos aos algoritmos *Multilayer Perceptron* e *Random Forest* e obtiveram 75% e 73,6% de acurácia respectivamente, contra 80,7% e 74,3% do conjunto de dados original.

Marquez-Vera, Romero e Ventura (2011) propõem a aplicação de técnicas de mineração de dados para prever o desempenho escolar através de dados reais de 670 estudantes do ensino médio de Zacatecas, no México. Na tentativa de melhorar a predição de desempenho final dos estudantes, o autor utiliza a técnica de Seleção de Atributos. Ele utiliza dez algoritmos de Seleção de Atributos para selecionar os melhores dos 77 que o seu conjunto de dados original possui. Os algoritmos utilizados, todos da ferramenta *Weka*, foram *Cfs Subset Eval*, *Chi Squared Attribute Eval*, *Consistency Subset Eval*, *Filtered Attribute Eval*, *OneR Attribute Eval*, *Filtered Subset Eval*, *Gain Ratio Attribute Eval*, *Info Gain Attribute Eval*, *Relief Attribute Eval*, *Symmetrical Uncert Attribute Eval*. Após aplicação da técnica de seleção, por meio de ranqueamento, foram selecionados somente os 15 melhores atributos. Os resultados obtidos com a Seleção de Atributos melhoraram os valores de verdadeiros negativos (*true negatives - TN*) e a média geométrica, mas não o valor da acurácia, em relação aos testes com todos os atributos.

Ramaswami e Bhaskaran (2009) utilizam dados socioeconômicos de estudantes para realizar um estudo sobre Seleção de Atributos utilizando o método de Filtro. Os algoritmos uti-

lizados são *Correlation-based Attribute Evaluation*, *Chi-Square Attribute*, *Gain-Ratio Attribute Evaluation*, *Information-Gain Attribute Evaluation*, *Relief Attribute* e *Symmetrical Uncertainty Attribute Evaluation*. O melhor resultado obtido foi alcançado pelo filtro *Information-Gain Attribute Evaluation* que selecionou sete atributos.

O presente trabalho diferencia-se dos demais utilizando a abordagem Cápsula(*Wrapper*) de Seleção de Atributos, técnica ainda não utilizada em base de dados extraídas de cursos de graduação a distância ofertados em AVA's.

3.1.4 Séries Temporais

Guércio et al. (2014) propõem a criação de um modelo conceitual que auxilie o professor na análise do comportamento dos alunos em diferentes períodos de tempo de determinada disciplina, permitindo assim, avaliar se um aluno possui tendência a aprovação ou reprovação antes do término da disciplina. Foi analisado o desempenho em disciplinas ministradas em curso de graduação. Os dados foram extraídos do AVA *Moddle* e do SGA utilizado na instituição de ensino. Foram utilizados os algoritmos *Random Forest*, *Random Tree* e J48 presentes na ferramenta *Weka*. Segundo os autores a acurácia média encontrada foi de 73% na classificação do desempenho e foi possível observar a possibilidade de transformar os dados armazenados na base de dados da plataforma *Moodle* em conhecimento, gerando regras extremamente úteis para o apoio à tomada de decisões.

Santos, Siebra e Oliveira (2014) acreditam que é importante identificar primeiramente quais são as disciplinas que mais influenciam na evasão do curso. Depois, utilizar essas disciplinas para predição de desempenho dos alunos ao longo das semanas. Com a predição de desempenho antecipada nessas disciplinas, será evitada a evasão do curso de graduação a distância. Foram analisados o desempenho em disciplinas ofertadas em cursos de graduação a distância. Os dados foram obtidos no AVA *Moodle* e no SCA[1] da instituição de ensino. Os algoritmos utilizados foram *SimpleCart*, J48 e o *ADTree* presentes na ferramenta *Weka*. A partir dos experimentos realizados, os autores constataram que, já ao final do primeiro período, é possível prever o risco de um aluno evadir do curso de graduação a distância com acurácia maior do que 80%.

Com o objetivo geral de investigar a possibilidade de geração de inferências relativas ao desempenho futuro de estudantes, Gottardo (2012) utiliza técnicas de mineração de dados utilizando atributos disponíveis em uma base de dados do AVA *Moodle*. Em seu quinto experimento, o autor utiliza o conceito de séries temporais obtendo taxas de acurácia próximas a 72%, mesmo em etapas iniciais de realização do curso. Segundo o autor, estas informações poderiam ser úteis para o desenvolvimento de ações envolvendo estudantes da turma em andamento e não apenas de turmas futuras.

[1] Sigla SCA não identificada pelo autor.

Este trabalho propõe uma abordagem baseada no conceito de séries temporais dividindo o período de oferta do curso em 8 partes. A abordagem proposta prevê ainda a inserção das notas de avaliações presenciais, no decorrer do curso, e a utilização da técnica cápsula de seleção de atributos, o que a diferencia das demais.

4 Ferramentas, Métodos e Experimentos

Neste capítulo serão descritos as ferramentas, a descrição da base de dados utilizada, as abordagens propostas e os experimentos realizados bem como os resultados obtidos. Começamos apresentando a ferramenta *Weka* utilizada no processo de mineração de dados, assim como o formato do arquivo utilizado. Em seguida uma breve descrição do AVA utilizado para a extração dos dados, os algoritmos classificadores empregados nos experimentos, bem como a descrição dos conjuntos de dados extraídos, além dos experimentos e seus resultados.

4.1 Weka

Weka (*Waikato Enviroment for Knowledge Analysis*) (WAIKATO, 2016) é um conjunto de algoritmos de Aprendizado de Máquina para tarefas de Mineração de Dados. Foi desenvolvido na Universidade de Waikato, na Nova Zelândia e é distribuído sob os termos da GNU *General Public License* (GPL). O sistema é escrito na linguagem Java e devido a sua portabilidade é possível utilizar o *Weka* em diversas plataformas. Os algoritmos podem ser aplicados diretamente a um conjunto de dados ou chamado a partir de seu próprio código Java. *Weka* contém ferramentas para pré-processamento de dados, classificação, regressão, clustering, regras de associação, e visualização. É também bem adequada para o desenvolvimento de novos sistemas de Aprendizagem de Máquina (WAIKATO, 2016).

A Figura 9, apresenta a tela inicial da ferramenta Weka na sua versão 3.7.13, versão utilizada nos experimentos. O *software Weka* está disponível para *download* [1] no *site* do desenvolvedor.

Figura 9 – Tela inicial do *Weka*

Fonte: Waikato (2016)

[1] http://www.cs.waikato.ac.nz/ml/weka/downloading.html

Este trabalho utilizou a opção *Weka Explorer* que é um ambiente que permite ao usuário executar apenas um algoritmo por vez apresentando diversas estatísticas a respeito do seu desempenho. É neste ambiente que são executadas as tarefas de pré-processamento de dados, classificação, regressão, clustering, regras de associação, seleção de atributos, e visualização (BOUCKAERT et al., 2015).

4.1.1 Arquivos ARFF

O *Weka* utiliza o formato de arquivo *Attribute Relation File Format* (arff) para organizar os dados de forma que possam ser aplicadas às técnicas de Mineração de Dados.

A Figura 10 mostra um exemplo de arquivo de treinamento, no formato arff e sua estrutura. As informações presentes no arquivo são organizadas através de marcações. As linhas precedidas do símbolo '%' representa um comentário, não possuindo qualquer influência no processo de mineração. A tag '@ *relation*' define o título, ou seja, o nome do conjunto de dados. A tag '@*attribute*' identifica cada atributo deste conjunto de dados assim como o tipo do dado, neste exemplo '*REAL*'. O último atributo declarado é a classe a ser predita e possui entre '{}' os seus possíveis valores. A sequência dos atributos nesta lista tem importância fundamental, pois é esta ordem que indica a posição da coluna na seção de dados do arquivo. Por exemplo, se um atributo é o terceiro a ser declarado, o *Weka* espera que todos os valores da terceira coluna da seção de dados sejam deste atributo. A tag '@ *data*' define os dados, cada linha representa um exemplo, ou seja, uma instância. Os valores desconhecidos no conjunto de dados são representados pelo sinal de '?'.

```
% 1. Title: Iris Plants Database

@RELATION iris

@ATTRIBUTE sepallength        REAL
@ATTRIBUTE sepalwidth         REAL
@ATTRIBUTE petallength        REAL
@ATTRIBUTE petalwidth         REAL
@ATTRIBUTE class      {Iris-setosa,Iris-versicolor,Iris-virginica}

@DATA
5.1, ? ,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
7.0,3.2,4.7,1.4,Iris-versicolor
 ? ,3.2,4.5,1.5,Iris-versicolor
6.9,3.1,4.9,1.5,Iris-versicolor
6.3,3.3,6.0,2.5,Iris-virginica
5.8,2.7,5.1,1.9,Iris-virginica
7.1,3.0, ? ,2.1,Iris-virginica
```

Figura 10 – Arquivo de treinamento no formato arff

Fonte: Elaborado pelo autor.

4.2 Moodle

Segundo Moodle (2016), o *Modular Object-Oriented Dynamic Learning Environment* (*Moodle*) é um Sistema *Open Source* de Gerenciamento de Cursos - *Course Management System* (CMS), também conhecido como *Learning Management System* (LMS) ou um Ambiente Virtual de Aprendizagem (AVA). Sua concepção foi iniciada nos anos 90, por Martin Dougiamas, com base nas abordagens pedagógicas do construtivismo e do construcionismo social, onde o aluno contribui ativamente no processo de ensino-aprendizagem.

O objetivo desse projeto, desde sua concepção, era de suportar a criação e administração de cursos com enfoque no trabalho colaborativo, em um ambiente de simples e intuitiva utilização. Atualmente o *Moodle* possui 77.210 sites registrados em 233 países com um total de 88.573.191 usuários. No *ranking* de sites registrados o Brasil ocupa o terceiro lugar atrás dos Estados Unidos (11.199) e Espanha (7.298) com 4.414 registros (MOODLE, 2016).

4.3 Algoritmos Utilizados

Os algoritmos de classificação são utilizados para gerar modelos que descrevem classes ou conceitos que são utilizados com o propósito de identificar exemplos que ainda não foram classificados. O modelo é criado a partir do treinamento do classificador através de um conjunto de dados, corretamente rotulados, denominado conjunto de treinamento. A performance da classificação é obtida quando um conjunto de teste, conjunto de dados sem rótulos, é submetido ao modelo. Redes neurais, árvores de decisão, regras de decisão, métodos estatísticos são algumas das técnicas empregadas na tarefa de classificação.

Os resultados teóricos mostram que não há um único algoritmo que pode ser uniformemente mais preciso do que os outros em todos os domínios. Embora tais teoremas são de aplicabilidade limitada na prática, muito pouco se sabe sobre quais algoritmos escolher para problemas específicos (KOHAVI; SOMMERFIELD; DOUGHERTY, 1996). Ainda em Kohavi, Sommerfield e Dougherty (1996), é mostrado experimentalmente que a escolha por um único bom algoritmo para todas as tarefas de mineração de dados não é trivial. Por isso, a escolha de vários algoritmos para realizar a tarefa desejada pode ser feita, levando à obtenção de diversos modelos.

Este trabalho optou por escolher e testar os seguintes algoritmos de classificação, *AdaBoost* (FREUND; SCHAPIRE et al., 1996), *BayesNet* (HECKERMAN, 1995), IBk (AHA; KIBLER; ALBERT, 1991), J48 (QUINLAN, 1993), *Random Forest* (BREIMAN, 2001), JRip (COHEN, 1995), *Multilayer Perceptron* (RUMELHART; HINTON; WILLIAMS, 1986) e SVM (PLATT, 1998), utilizados ou indicados em alguns trabalhos pesquisados Kotsiantis, Pierrakeas e Pintelas (2003), Wu et al. (2008), Witten, Frank e Hall (2011) e Manhães (2015).

4.4 Base de Dados

A base de dados utilizada neste trabalho foi extraída de um AVA *Moodle* e um Sistema de Gerenciamento Acadêmico (SGA) utilizados por uma IFES que oferta 4 cursos de graduação na modalidade a distância, Administração, Física, Matemática e Química. Todo o processo, desde o entendimento do domínio da aplicação, até a geração do arquivo para submissão ao algoritmo de classificação, passando pelo tratamento dos dados, está representado na Figura 11.

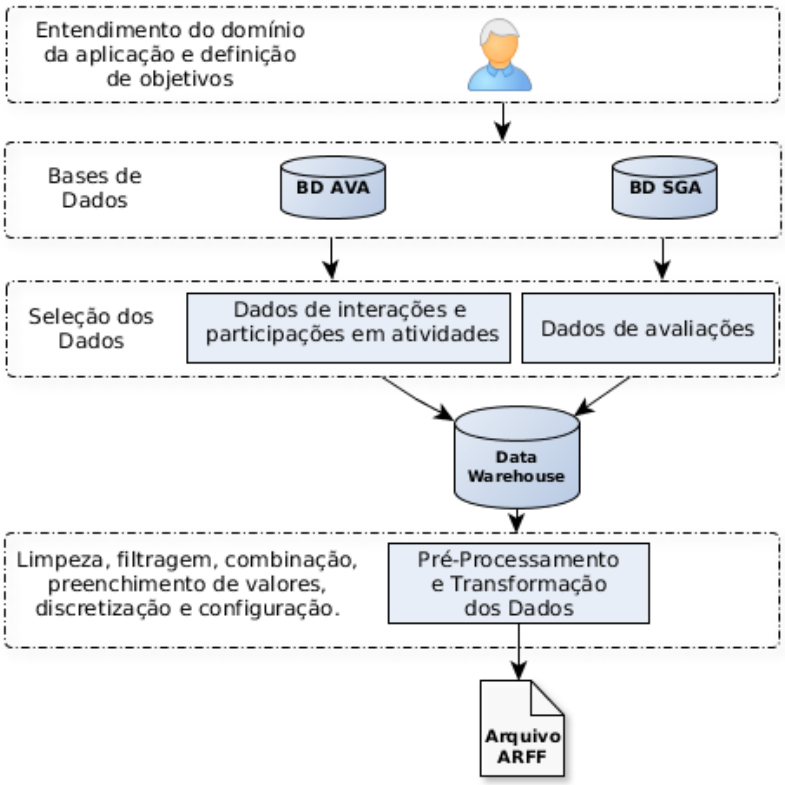

Figura 11 – Processo de obtenção dos dados

Fonte: Elaborado pelo autor.

O banco de dados do AVA *Moodle* utilizado na instituição educacional possui exatamente 291 tabelas onde é possível extrair os mais diversos atributos que podem ser utilizados para representar os estudantes. Neste trabalho optou-se por utilizar atributos que representem o comportamento na plataforma, ou seja, a forma como o estudante interage com as atividades e recursos disponibilizados. A partir da tabela de *log* do sistema, foram extraídos dados quantitativos sobre o comportamento e participação dos alunos durante a realização do curso. Do SGA foram retirados os dados relativos às notas de avaliações parciais e notas finais dos estudantes durante a realização do curso. O processo de escolha das disciplinas para representar cada curso foi baseado nos seguintes critérios:

- Disciplina específica do curso ofertada no primeiro ano da graduação;

- Número de alunos matriculados;
- Quantidade de recursos utilizados;
- Quantidade de interações dos estudantes com o AVA *Moodle*.

Após o processo de seleção das disciplinas os dados foram extraídos dos bancos de dados do AVA *Moodle* e do SGA, através da criação de consultas e funções de inserção, e salvos em um *Data Warehouse* com o objetivo de unificar a fonte de informações e mantê-las em um único repositório, o sistema gerenciador de banco de dados (SGDB) utilizado na instituição e neste trabalho é o *PostgreSQL* na sua versão 9.4.

Com os dados reunidos em um único repositório temos as etapas de pré-processamento e transformação dos dados, que compreende algumas ações como limpeza, filtragem, combinação de valores e discretização. O objetivo destas ações é que sejam eliminadas as inconsistências, contradições, duplicidades e outros problemas existentes, bem como criação ou alteração de tabelas, visões e funções. A inconsistência surge quando, para um mesmo atributo, existem valores diferentes que representam a mesma coisa, como por exemplo, Minas Gerais, MG, Minas. A duplicidade ocorre quando um mesmo registro é armazenado mais de uma vez na base de dados, por exemplo, um mesmo registro de um estudante armazenado duas vezes, porém com identificações diferentes. A contradição ocorre quando há uma incoerência entre duas informações, por exemplo, um registro com idade 18 anos e escolaridade Mestrado seria um caso de contradição. A discretização, neste caso, é a obtenção da classe a ser predita, onde a nota final, que é dada em números sequenciais, é transformada em valores discretos, como por exemplo 'Aprovado', 'Reprovado', 'Classe A'.

Durante a etapa de pré-processamento e transformação dos dados foram obtidos 45 atributos que estão listados na Tabela 3.

Tabela 3 – Relação dos atributos utilizados

Grupo	Atributo	Descrição
Tarefas	up_tarefas	Envio de atividades
	all_tarefas	Acesso a todas as atividades disponíveis no curso
	page_tarefas	Acesso à página de determinada atividade
Blog	ad_blog	Adição de blog
	acesso_blog	Acesso ao blog
Chat	acesso_chat	Acesso à sala do chat
	msg_chat	Mensagens enviadas no chat
	rel_chat	Acesso aos chat's realizados
Questionário Múltipla Escolha	resp_quest_nov	Responde questionário novamente
	resp_quest	Responde questionário
	acesso_quest	Acesso ao questionário
	acesso_quest_all	Acesso a todos os questionários
Curso	acesso_curso	Acesso ao curso
Materiais	acesso_material_all	Acesso a todas as pastas de conteúdo
	acesso_material	Acesso à pasta de conteúdos
Fórum	del_post_forum	Quantidade de posts de uma discussão apagadas
	ad_discussao_forum	Adição de uma discussão no fórum
	insc_forum_all	Se inscrever para receber atualizações de todos os fóruns
	enc_forum	Encerrar inscrição para receber atualizações de um fórum
	acesso_forum	Acesso à página principal do fórum de discussão
	del_discussao_forum	Quantidade de discussões de um fórum apagadas
	enc_forum_all	Encerrar inscrição para receber atualizações de todos os fóruns
	upd_post_forum	Atualização de post dentro de um fórum
	acesso_discussao_forum	Acesso a determinada discussão dentro de um fórum
	ad_post_forum	Adição de post na discussão do fórum
	acesso_foruns	Acesso a todos os fóruns disponibilizados no curso
	insc_forum	Se inscrever para receber atualizações de um fórum
Informações	acesso_inf	Acesso à página de informações do curso(ementa, bibliografia)
Recursos	acesso_lista_recursos	Acesso a todos os materiais disponibilizados no curso
	acesso_recurso	Visualização de apostilas ou vídeos disponibilizados
Vídeos	acesso_video	Acesso à vídeos em sites externos
	acesso_video_all	Acesso à listagem de todos os vídeos em sites externos
Usuários	upd_perfil	Atualização do perfil
	acesso_user_all	Acesso à página com todos os usuários do curso
	acesso_user	Acesso a determinado usuário
Wiki	comentario_wiki	Acesso a determinado comentário da wiki
	acesso_dif_wiki	Acesso a comparações entre versões da wiki
	hist_wiki	Histórico de modificações na wiki
	acesso_wiki	Acesso à wiki
	restaurar_wiki	Restaurar versão da wiki
	ad_item_wiki	Adição de um novo item na wiki
	comentarios_wiki	Acesso a todos os comentários da wiki
	acesso_mapa_wiki	Acesso ao mapa da wiki
	upd_wiki	Edição de um item da wiki
	classe	Classe a ser prevista pelos algoritmos de classificação

As bases dos Cursos foram divididas em 2 classes (Aprovado e Reprovado) através do processo conhecido como discretização (estudantes foram alocados em determinada classe de acordo com a sua nota final). A classe Aprovado compreende os estudantes que obtiveram notas entre 60 e 100, enquanto na classe Reprovado estão os alunos com notas entre 0 e 59 pontos. A distribuição completa dos conjuntos de dados obtidos pode ser observada na Tabela 4.

Tabela 4 – Relação dos conjuntos de dados obtidos e suas respectivas classes

Conjunto	Número total de estudantes	Número de estudantes na classe Aprovado	Número de estudantes na classe Reprovado
Administração	248 (100%)	120 (48,4%)	128 (52,6%)
Física	122 (100%)	75 (61,5%)	47 (38,5%)
Matemática	69 (100%)	47 (68,1%)	22 (31,9%)
Química	31 (100%)	16 (51,6%)	15 (48,4%)

4.5 Experimentos

Esta seção descreve os resultados experimentais realizados. Os testes foram realizados em um computador com processador Intel(R) Core(TM) i7 com 2.0 GHz de frequência de *clock*, 8GB de memória RAM e com o sistema operacional Ubuntu de 64 *bits*. Para facilitar e simplificar a tabulação dos dados, os nomes dos algoritmos *AdaBoost*, *Bayes Net*, *Multilayer Perceptron* e *Random Forest* foram abreviados para AB, BN, MP e RF respectivamente. A divisão dos conjuntos de treinamento e teste, quando não mencionado outra forma, foi realizado através do processo *k-fold cross validation* com $k = 10$. Para avaliar o desempenho dos algoritmos de classificação, este trabalho utiliza as métricas Acurácia (Ac.), que possui valores entre 0 e 100%, e a *F-Measure* (F) que tem o seu melhor valor em 1 e o pior em 0, e foi utilizada para avaliar o desempenho de predição do classificador em cada classe individualmente, devido ao problema de desbalanceamento de algumas classes. Nos resultados dos experimentos são apresentados três valores para a *F-Measure* que representam, respectivamente, as classes Aprovado, Reprovado, e a Média dos valores considerando ambas as classes.

4.5.1 Experimento 1 - Comparação de Algoritmos

Neste experimento foram utilizados os conjuntos de dados originais, de cada um dos quatro cursos analisados, com todos os atributos conforme apresentados na Tabela 3 e divisão das classes de acordo com a Tabela 4. Os melhores valores obtidos para cada curso estão destacados em negrito.

O objetivo deste primeiro experimento é verificar o desempenho dos algoritmos de mineração de dados utilizando duas classes.

Observa-se na Tabela 5 que o algoritmo *Random Forest* foi o mais eficiente, obtendo melhor desempenho em 3 dos 4 conjuntos. Apenas no Curso de Física este algoritmo não obteve o melhor desempenho, cabendo ao *Multilayer Perceptron* a liderança com acurácia de 90,2%, classificando corretamente 110 dos 122 alunos. No Curso de Química, mesmo com poucos exemplos (estudantes), os algoritmos *Adaboost*, *BayesNet* e *Random Forest* obtiveram 80,6% de acurácia. Os classificadores obtiveram as menores acurácias no Curso de Administração, onde o maior valor atingindo foi 72,2% com o *Random Forest*.

Mesmo com dados desbalanceados no Curso de Matemática, os classificadores obtiveram bons valores da *F-Measure* na classe minoritária (Reprovado), chegando a 0,780 com o algoritmo *Random Forest*.

4.5.2 Experimento 2 - Balanceamento de Classes

O problema de classes desbalanceadas ocorre quando o número de exemplos de uma classe é muito inferior ao número de exemplos da outra classe. Esse problema pode ser resolvido

Tabela 5 – Resultados para a classificação dos conjuntos de dados originais

Algoritmo	**Administração**		**Física**		**Matemática**		**Química**		**Média**
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*
AB	67,3%	0,687 0,658 0,672	82,0%	0,855 0,761 0,819	78,3%	0,845 0,634 0,778	**80,6%**	0,824 0,786 0,805	77,1%
BN	71,8%	0,722 0,713 0,718	85,2%	0,882 0,804 0,852	82,6%	0,872 0,727 0,826	**80,6%**	0,824 0,786 0,805	80,1%
IBk	66,9%	0,634 0,699 0,667	78,7%	0,824 0,729 0,788	79,7%	0,848 0,696 0,799	74,2%	0,750 0,733 0,742	74,9%
J48	63,3%	0,629 0,637 0,633	82,0%	0,857 0,756 0,818	73,9%	0,713 0,571 0,736	74,2%	0,778 0,692 0,736	73,4%
JRip	66,1%	0,667 0,656 0,661	81,1%	0,854 0,736 0,808	76,8%	0,826 0,652 0,771	74,2%	0,778 0,692 0,736	74,6%
MP	62,5%	0,614 0,635 0,625	**90,2%**	0,921 0,870 0,901	78,3%	0,835 0,681 0,786	67,7%	0,688 0,667 0,677	74,7%
RF	**72,2%**	0,729 0,714 0,721	86,9%	0,899 0,814 0,866	**87,0%**	0,907 0,780 0,867	**80,6%**	0,824 0,786 0,805	**81,7%**
SVM	67,3%	0,633 0,705 0,671	86,1%	0,892 0,805 0,858	84,1%	0,882 0,756 0,841	74,2%	0,733 0,750 0,741	77,9%

aplicando a técnica de Balanceamento de Classes para equilibrar o número de exemplos de cada classe. Neste experimento, foi utilizado o conjunto de dados do Curso de Matemática, devido ao problema de desbalanceamento identificado no Experimento 1, onde a classe Reprovado possui quantidade de exemplos significativamente inferior em relação a classe Aprovado (47 exemplos da classe Aprovado (68,1%) e 22 exemplos da classe Reprovado (31,9%)). A técnica de balanceamento de classes utilizada foi a *SMOTE* e os parâmetros aplicados foram, taxa de sobre-amostragem de 100% e número de vizinhos igual a 5. Após o balanceamento, a base de dados se constituiu de, 47 exemplos da classe Aprovado (51,6%) e 44 exemplos da classe Reprovado (48,4%).

O objetivo deste segundo experimento é verificar o desempenho da classificação utilizando os conjuntos de dados balanceados.

A Tabela 6, apresenta os resultados obtidos pelos classificadores utilizando os conjuntos de dados original e balanceado do Curso de Matemática. Pode-se observar um aumento da acurácia em quase todos os algoritmos, em especial os algoritmos J48, JRip e IBk, que tiveram um aumento, respectivamente, de 10,7%, 8,9% e 7,1% na acurácia.

Tabela 6 – Resultados para a classificação dos conjuntos de dados original e balanceado do Curso de Matemática

Algoritmo	**Conjunto Original do Curso de Matemática**		**Conjunto Balanceado do Curso de Matemática**	
	Ac.	*F*	*Ac.*	*F*
AB	78,3%	0,845 0,634 0,778	82,4%	0,826 0,822 0,824
BN	82,6%	0,872 0,727 0,826	84,6%	0,860 0,829 0,845
IBk	79,7%	0,848 0,696 0,799	**86,8%**	0,864 0,872 0,868
J48	73,9%	0,713 0,571 0,736	84,6%	0,848 0,844 0,846
JRip	76,8%	0,826 0,652 0,771	85,7%	0,863 0,851 0,857
MP	78,3%	0,835 0,681 0,786	83,5%	0,828 0,842 0,835
RF	**87,0%**	0,907 0,780 0,867	**86,8%**	0,880 0,854 0,867
SVM	84,1%	0,882 0,756 0,841	83,5%	0,842 0,828 0,835

As exceções foram os algoritmos SVM e *Random Forest*, que tiveram uma diminuição do desempenho, mas estes dois tiveram um aumento significativo na predição da classe Reprovado, onde suas *F-Measure* passaram de 0,780 para 0,854 no caso do *Random Forest* e de 0,756 para 0,828 com o algoritmo SVM

A Figura 12, apresenta a comparação das acurácias obtidas pelos classificadores nos conjuntos de dados original e balanceado, evidenciando claramente o aumento de desempenho quando se utiliza a técnica de Balanceamento de Classes.

Figura 12 – Comparação dos resultados para a classificação dos conjuntos de dados original e balanceado do Curso de Matemática

A classe Reprovado representa os estudantes com maior probabilidade de terem um rendimento abaixo do esperado e provável reprovação. O aumento nos acertos de classificação das instâncias desta classe significa uma maior predição de estudantes com provável rendimento abaixo do esperado identificados. Através da identificação destes estudantes, podem-se realizar ações que previnam a sua reprovação ou melhorem o seu desempenho. A Tabela 7 mostra a comparação da *F-Measure* obtida pelos classificadores na classe Reprovado. A coluna Média representa o aumento médio da *F-Measure* obtida pelos classificadores utilizando a técnica de Balanceamento de Classes em relação ao conjunto de dados original para a classe Reprovado.

Tabela 7 – Comparação das F-Measure obtidas pelos algoritmos classificadores na classe Reprovado do Curso de Matemática

Bases	**AB**	**BN**	**IBk**	**J48**	**JRip**	**MP**	**RF**	**SVM**	**Média**
Conjunto Original	0,634	0,727	0,696	0,571	0,652	0,681	0,780	0,756	—
Conjunto Balanceado	**0,822**	**0,829**	**0,872**	**0,844**	**0,851**	**0,842**	**0,854**	**0,828**	**0,156**

É possível observar na Tabela 7 que todos os algoritmos obtiveram aumento de desempenho na classificação da classe Reprovado, resultando em um aumento médio de 0,156 da *F-Measure*.

4.5.3 Experimento 3 - Seleção de Atributos

Os objetivos deste terceiro experimento são:

- Descobrir quais os atributos e, consequentemente, quais recursos influenciam no desempenho dos estudantes.
- Verificar se há um aumento no desempenho do algoritmo de Mineração de Dados quando se utiliza a seleção de atributos.

Neste experimento foi aplicado o processo de Seleção de Atributos pelo método Cápsula (*Wrapper*), porque, dentre os trabalhos pesquisados, nenhum fez uso deste método em conjuntos de dados retirados de cursos de graduação à distância ofertado em AVA's, e também pelo fato deste método alcançar excelentes resultados em outros "tipos" de bases de dados (HAN; KAMBER; PEI, 2011), (MITCHELL, 1997).

O primeiro passo foi a utilização do algoritmo *WrapperSubsetEval* para selecionar um subconjunto de características que melhor representa cada um dos cursos. Estas características são específicas para cada curso e serão apresentadas a seguir[2].

No Curso de Administração, os atributos mais relevantes foram os relativos ao fórum (acesso_forum, ad_post_forum, upd_post_forum e acesso_discussao_forum), além de acessos a apostilas e vídeos (acesso_recurso), envio de atividades (up_tarefas), e número de acessos ao curso (acesso_curso).

Para o Curso de Física, os atributos mais relevantes também foram os relativos ao fórum (acesso_forum, ad_post_forum, ad_discussao_forum), além de acesso as pastas de materiais (acesso_material), acessos a vídeos de sites externos (acesso_video), acessos a apostilas e vídeos (acesso_recurso), envio de atividades (up_tarefas), e número de acessos ao curso (acesso_curso).

Os atributos mais relevantes no Curso de Matemática são os relativos ao fórum (acesso _forum, ad_post_forum, ad_discussao_forum, upd_post_forum), acessos a página de informação (acesso_inf), acesso ao histórico de conversas do chat (rel_chat), acessos ao curso (acesso _curso), acesso as pastas de materiais (acesso_material), envio de atividades (up_tarefas), acessos a apostilas, e vídeos (acesso_recurso).

Finalmente, no Curso de Química, os atributos mais relevantes são os relativos ao fórum (acesso_forum, ad_post_forum), acessos ao curso (acesso_curso), visita ao histórico de conversas do chat (rel_chat), além de acessos a apostilas e vídeos (acesso_recurso).

Os resultados dos classificadores aplicados aos conjuntos de dados dos cursos, após a seleção de atributos, são apresentados na Tabela 8.

Com a seleção de atributos no conjunto de dados do Curso de Administração, o melhor desempenho foi do algoritmo *Multilayer Perceptron* que alcançou 78,6% de acurácia. No Curso de Física ocorreu um empate, novamente o *Multilayer Perceptron* juntamente com o *Random Forest* alcançaram 90,2% de acurácia.

Para o conjunto de dados do Curso de Química, os melhores resultados foram obtidos pelos algoritmos IBK e SVM com 87,1% de acurácia. Já no Curso de Matemática, que é desbalanceado, outro empate, dessa vez entre os algoritmos JRip e SVM com 87% de acurácia.

Ainda no Curso de Matemática, a *F-Measure* da classe minoritária (Reprovado) teve

[2] Veja a Tabela 3 para a identificação dos atributos.

um aumento em relação ao resultado obtido pelo *Random Forest* no Experimento 1, passando de 0,780 para 0,791 - valor obtido pelo SVM.

Tabela 8 – Resultados para a classificação com Seleção de Atributos

Algoritmo	Administração		Física		Matemática		Química	
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*
AB	67,3%	0,669 0,677 0,673	87,7%	0,906 0,824 0,874	78,3%	0,842 0,651 0,781	83,9%	0,839 0,839 0,839
BN	73,4%	0,734 0,734 0,734	87,7%	0,903 0,831 0,876	85,5%	0,894 0,773 0,855	83,9%	0,857 0,815 0,837
IBk	66,5%	0,658 0,672 0,665	88,5%	0,911 0,837 0,883	81,2%	0,851 0,745 0,817	**87,1%**	0,882 0,857 0,870
J48	69,8%	0,708 0,686 0,697	83,6%	0,872 0,773 0,834	76,8%	0,833 0,619 0,765	77,4%	0,800 0,741 0,771
JRip	73,4%	0,744 0,723 0,733	83,6%	0,873 0,767 0,833	**87,0%**	0,909 0,769 0,864	83,9%	0,848 0,828 0,838
MP	**78,6%**	0,782 0,791 0,786	**90,2%**	0,924 0,860 0,900	84,1%	0,882 0,756 0,841	83,6%	0,848 0,828 0,838
RF	69,4%	0,694 0,694 0,694	**90,2%**	0,922 0,867 0,901	82,6%	0,875 0,714 0,824	83,9%	0,857 0,815 0,837
SVM	73,0%	0,715 0,743 0,730	88,5%	0,908 0,848 0,885	**87,0%**	0,905 0,791 0,869	**87,1%**	0,875 0,867 0,871

Na Tabela 9 são apresentadas as comparações dos desempenhos obtidos através das duas abordagens empregadas nos experimentos realizados neste trabalho. A coluna Média representa o aumento médio da acurácia obtido pelos classificadores utilizando a técnica de Seleção de Atributos em relação aos conjuntos de dados originais.

No Curso de Administração, houve uma melhora significativa no desempenho dos algoritmos *Multilayer Perceptron*, JRip, J48, *BayesNet* e SVM. O algoritmo *Multilayer Perceptron* passou de 155 acertos para 195 de um total de 248 alunos, obtendo o melhor resultado dentre os algoritmos testados neste conjunto de dados.

No Curso de Física, todos os algoritmos tiveram um aumento no seu desempenho, com exceção do *Multilayer Perceptron* que manteve sua acurácia. Os melhores resultados foram obtidos pelos algoritmos *Random Forest* e *Multilayer Perceptron*, que classificaram 110 dos 122 estudantes corretamente.

No Curso de Matemática, com exceção do algoritmo *Random Forest*, todos os algoritmos tiveram um aumento do desempenho, sendo o JRip e SVM os algoritmos com melhores resultados (classificaram corretamente 60 estudantes do total de 69).

Tabela 9 – Comparação das acurácias obtidas pelos algoritmos de Classificação com Seleção de Atributos

Algoritmos	AB	BN	IBk	J48	JRip	MP	RF	SVM	Média
Administração	67,3%	71,8%	66,9%	63,3%	66,1%	62,5%	72,2%	67,3%	-
Seleção Adm.	67,3%	**73,4%**	66,5%	**69,8%**	**73,4%**	**78,6%**	69,4%	**73,0%**	**4,3%**
Física	82,0%	85,2%	78,7%	82,0%	81,1%	90,2%	86,9%	86,1%	-
Seleção Fís.	**87,7%**	**87,7%**	**88,5%**	**83,6%**	**83,6%**	90,2%	**90,2%**	**88,5%**	**3,5%**
Matemática	78,3%	82,6%	79,7%	73,9%	76,8%	78,3%	87,0%	84,1%	-
Seleção Mat.	78,3%	**85,5%**	**81,2%**	**76,8%**	**87,0%**	**84,1%**	82,6%	**87,0%**	**2,7%**
Química	80,6%	80,6%	74,2%	74,2%	74,2%	67,7%	80,6%	74,2%	-
Seleção Quí.	**83,9%**	**83,9%**	**87,1%**	**77,4%**	**83,9%**	**83,9%**	**83,9%**	**87,1%**	**8,1%**

No Curso de Química, todos os algoritmos tiveram um aumento do desempenho, com destaque para a melhora do algoritmo *Multilayer Perceptron*, que atingiu 83,9% de acurácia classificando 26 dos 31 estudantes corretamente (aumento de 5 (16,2%) diante dos 21 classificados anteriormente). Os melhores desempenhos foram dos algoritmos IBk e SVM, que classificaram corretamente 27 estudantes do total de 31. Ainda, no Curso de Química, o aumento médio da acurácia obtida pelos classificadores chegou a 8,1%.

Os resultados obtidos comprovam que a seleção de atributos pelo método Cápsula pode ser utilizada para melhorar o desempenho dos algoritmos classificadores na realização de inferências relativas ao desempenho acadêmico.

Com o objetivo de verificar o motivo da redução do desempenho do algoritmo *Random Forest*, nos conjuntos dos cursos de Administração e Matemática, optou-se por realizar a seleção de atributos pelo método filtro utilizando os algoritmos *Cfs Subset Eval*, *Chi Squared Attribute Eval*, *Info Gain Attribute Eval*, *Gain Ratio Attribute Eval*, *Relief FAttribute Eval*, utilizados em alguns trabalhos relacionados, tais como Ramaswami e Bhaskaran (2009) e Marquez-Vera, Romero e Ventura (2011). A utilização destes cinco algoritmos retornaram um total de 9 atributos para o conjunto de dados do Curso de Administração (insc_forum_all, acesso_forum, ad_discussao_forum, page_tarefas, acesso_video, acesso_recurso, acesso_discussao_forum, up_tarefas, acesso_curso) e 7 para o Curso de Matemática (acesso_material, page_tarefas, acesso_user_all, up_tarefas, acesso_curso, acesso_forum, ad_post_forum), que foram submetidos ao classificador *Random Forest*. Como podemos observar na Tabela 10, os resultados obtidos pelo método filtro ficaram abaixo dos obtidos tanto no conjunto original quanto no método cápsula. Isso permite concluir que o algoritmo *Radom Forest*, para as bases citadas, não tem aumento de desempenho utilizando a técnica de Seleção de Atributos

Através da aplicação da técnica de Seleção de Atributos pela abordagem Cápsula, foi possível extrair as características que possuem maior influência no aprendizado dos estudantes em cada um dos cursos analisados. Utilizando-se destas características, os resultados do Experimento mostraram um aumento significativo no desempenho dos classificadores na realização de inferências sobre o desempenho acadêmico dos estudantes. O algoritmo *Random Forest* obteve 90,2% de acurácia no Curso de Física, enquanto no Curso de Química, todos os algoritmos

Tabela 10 – Comparação dos resultados para a classificação dos conjuntos original e seleção de atributos pelos métodos cápsula e filtro

Curso	**Conjunto Original**		**Método Cáspula**		**Método Filtro**	
	Ac.	***F***	***Ac.***	***F***	***Ac.***	***F***
Administração	**72,2%**	**0,729** **0,714** **0,721**	69,4%	0,694 0,694 0,694	66,9%	0,667 0,672 0,669
Matemática	**87,0%**	**0,907** **0,780** **0,867**	82,6%	0,875 0,714 0,824	81,9%	0,869 0,707 0,817

tiveram um aumento no desempenho, demonstrando a viabilidade da aplicação da técnica de Seleção de Atributos pela abordagem Cápsula.

4.5.4 Experimento 4 - Séries Temporais

Neste experimento foi aplicado o conceito de Séries Temporais para propor um modelo de acompanhamento de estudantes durante a realização de um curso a distância. Na IFES a oferta de disciplinas nos cursos à distância é realizada de forma semestral. Durante o semestre são realizadas duas avaliações presenciais, sendo este o procedimento padrão para todos os cursos de graduação ofertados.

O período de oferta da disciplina foi dividido em 8 períodos. Como pode ser observado na Figura 13, os períodos possuem duração de aproximadamente 15 dias, com exceção do primeiro que possui 25 dias. Devido aos ajustes de matrículas e a ambientação do estudante na plataforma, optou-se pelo maior número de dias neste período.

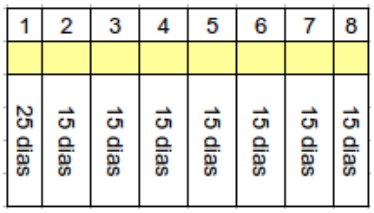

Figura 13 – Divisão dos períodos da série temporal

A primeira avaliação presencial acontece no final do período 3 e a partir do período 4 esta nota parcial é inserida como um atributo adicional ao conjunto de dados original, conforme a Figura 14.

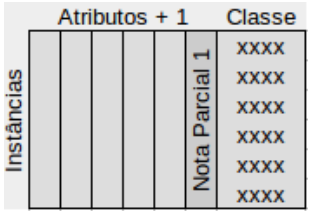

Figura 14 – Conjunto de dados original mais nota da primeira avaliação

A segunda avaliação presencial acontece ao final do sétimo período e no período 8. Esta segunda nota parcial é inserida como um segundo atributo adicional ao conjunto de dados, como pode ser observado na Figura 15.

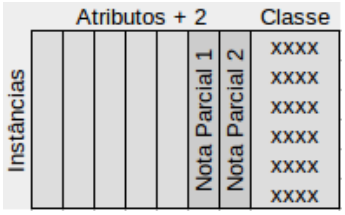

Figura 15 – Conjunto de dados original mais as notas da primeira e segunda avaliação

O conjunto de dados analisado foi o do Curso de Administração, por ser o curso com o maior número de estudantes atendidos. Como a oferta da disciplina foi dividido em 8 períodos de tempo, foram gerados 8 conjuntos de dados, onde o conjunto 1 contém as interações dos estudantes com a plataforma nos 25 primeiros dias da oferta. O conjunto de dados 2 contém as interações dos 25 dias do período 1 mais os 15 dias do período 2, e assim sucessivamente até o oitavo conjunto de dados.

O objetivo da divisão do conjunto de dados em séries temporais é que os responsáveis pela turma (professores, tutores) tenham o feedback do progresso do estudante e possam acompanhá-los durante a realização do curso. Nesta proposta de divisão, os responsáveis terão 3 feedbacks antes da primeira avaliação e mais 4 entre as duas avaliações presenciais.

Os objetivos deste estudo são:

- Analisar o desempenho do modelo de previsão em séries temporais no processo de acompanhamento do estudante durante a realização da disciplina;
- Analisar o desempenho do modelo de previsão em séries temporais ao inserir as notas parciais das avaliações no processo de acompanhamento do estudante durante a realização da disciplina;
- Analisar o desempenho do modelo de previsão em séries temporais ao inserir as notas parciais das avaliações e a seleção de atributos;

- Analisar a viabilidade de avaliar o desempenho de estudantes em estágios precoces da disciplina.

A Tabela 11 mostra os resultados dos algoritmos aplicados às séries temporais utilizando o conjunto de dados original. Temos o melhor desempenho ao longo da série com algoritmo *BayesNet* que obteve uma média de 68,1% de acurácia nos 8 conjuntos temporais.

Tabela 11 – Resultados para a classificação do conjunto original em séries temporais

Algoritmo	**1**		**2**		**3**		**4**		**5**		**6**		**7**		**8**		**Média**
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*
		0,537		0,520		0,584		0,626		**0,711**		0,646		0,650		0,687	
AB	64,5%	0,612	61,3%	0,676	62,1%	0,652	62,9%	0,632	**68,5%**	**0,655**	62,9%	0,610	65,7%	0,664	67,3%	0,658	64,4%
		0,627		0,600		0,619		0,629		**0,682**		0,628		0,657		0,672	
		0,593		0,604		**0,650**		**0,650**		0,672		**0,693**		0,691		0,722	
BN	67,3%	0,727	66,1%	0,704	**66,5%**	**0,680**	**66,5%**	**0,680**	67,7%	0,683	**69,0%**	**0,686**	69,8%	0,704	71,8%	0,713	**68,1%**
		0,662		0,656		**0,665**		**0,665**		0,678		**0,689**		0,698		0,718	
		0,564		0,524		0,603		0,606		0,598		0,612		0,650		0,634	
IBk	60,1%	0,632	55,2%	0,578	63,3%	0,659	61,7%	0,627	62,1%	0,641	63,7%	0,659	66,1%	0,672	66,9%	0,699	62,4%
		0,599		0,552		0,632		0,617		0,620		0,636		0,661		0,667	
		0,485		0,519		0,554		0,584		0,664		0,637		0,607		0,629	
J48	59,7%	0,669	64,1%	0,714	61,7%	0,664	59,3%	0,601	65,7%	0,650	63,7%	0,637	64,5%	0,676	63,3%	0,637	62,8%
		0,580		0,620		0,611		0,593		0,657		0,637		0,643		0,633	
		0,553		**0,607**		0,638		0,647		0,623		0,600		**0,684**		0,667	
JRip	64,1%	0,700	**66,5%**	**0,709**	65,7%	0,674	65,7%	0,667	62,9%	0,635	61,3%	0,625	**70,2%**	**0,718**	66,1%	0,656	65,3%
		0,629		**0,659**		0,657		0,657		0,629		0,613		**0,701**		0,661	
		0,457		0,467		0,529		0,531		0,583		0,586		0,613		0,614	
MP	59,7%	0,679	61,3%	0,696	60,5%	0,660	60,1%	0,653	62,5%	0,659	62,9%	0,664	65,3%	0,686	62,5%	0,635	61,9%
		0,572		0,585		0,596		0,594		0,622		0,626		0,651		0,625	
		0,675		0,594		0,626		0,594		0,598		0,613		0,664		**0,729**	
RF	**68,5%**	**0,695**	60,9%	0,623	63,3%	0,640	60,9%	0,623	60,9%	0,620	61,3%	0,613	66,5%	0,667	**72,2%**	**0,714**	64,3%
		0,685		0,609		0,633		0,609		0,609		0,613		0,665		**0,721**	
		0,522		0,527		0,528		0,557		0,545		0,529		0,591		0,633	
SVM	65,3%	0,728	64,5%	0,716	63,3%	0,700	65,3%	0,715	64,9%	0,715	64,1%	0,710	66,5%	0,717	67,3%	0,705	65,2%
		0,628		0,625		0,617		0,639		0,632		0,623		0,656		0,671	
Ac. Média	63,7%	—	62,5%	—	63,3%	—	62,8%	—	64,4%	—	63,6%	—	66,8%	—	67,2%	—	—

Pode-se observar um baixo desempenho dos algoritmos classificadores no início da série, e conforme progride-se ao oitavo período, tem-se uma melhora no desempenho dos classificadores. Este fato ocorre devido ao aumento do volume de dados gerado pelas interações dos estudantes com a plataforma no decorrer do tempo, melhorando o treinamento e consequente aprendizado dos classificadores.

A Tabela 12 apresenta os resultados da classificação do conjunto original de dados mais as notas das duas avaliações presenciais. Pode-se observar um aumento considerável no desempenho de todos os algoritmos classificadores testados ao inserir a primeira nota parcial a partir do período 4. Ao inserir a primeira nota parcial, o algoritmo *AdaBoost* obteve o melhor desempenho dentre os classificadores utilizados, passando de 62,1% de acurácia no período 3 para 73,79% no período 4. O melhor desempenho médio para os oito períodos foi do algoritmo JRip com 71,7% de acurácia.

Tabela 12 – Resultados para a classificação do conjunto original mais notas parciais em séries temporais

Algoritmo	**1**		**2**		**3**		**4**		**5**		**6**		**7**		**8**		**Média**
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*
		0,537		0,520		0,584		**0,743**		0,753		0,763		0,752		0,785	
AB	64,5%	0,612	61,3%	0,676	62,1%	0,652	**73,8%**	**0,733**	74,6%	0,739	74,6%	0,761	75,8%	0,764	79,0%	0,795	70,7%
		0,627		0,600		0,619		**0,738**		0,746		0,762		0,758		0,790	
		0,593		0,604		**0,650**		0,686		0,686		0,702		0,718		0,786	
BN	67,3%	0,727	66,1%	0,704	**66,5%**	**0,680**	69,4%	0,701	69,4%	0,701	69,4%	0,717	72,2%	0,725	78,2%	0,779	69,8%
		0,662		0,656		**0,665**		0,694		0,694		0,710		0,722		0,782	
		0,564		0,524		0,603		0,691		0,700		0,669		0,617		0,743	
IBk	60,1%	0,632	55,2%	0,578	63,3%	0,659	69,8%	0,704	71,0%	0,719	68,5%	0,700	69,9%	0,641	75,4%	0,764	66,7%
		0,599		0,552		0,632		0,698		0,710		0,685		0,629		0,754	
		0,485		0,519		0,554		0,681		0,688		0,675		0,695		0,746	
J48	59,7%	0,669	64,1%	0,714	61,7%	0,664	69,4%	0,705	69,0%	0,691	69,0%	0,672	70,6%	0,716	75,0%	0,754	67,3%
		0,580		0,620		0,611		0,693		0,690		0,673		0,706		0,750	
		0,553		**0,607**		0,638		0,724		**0,744**		**0,724**		**0,767**		0,759	
JRip	64,1%	0,700	**66,5%**	**0,709**	65,7%	0,674	73,0%	0,735	**75,0%**	**0,756**	**75,0%**	**0,735**	**77,4%**	**0,781**	77,0%	0,780	**71,7%**
		0,629		**0,659**		0,657		0,730		**0,750**		**0,730**		**0,774**		0,770	
		0,457		0,467		0,529		0,708		0,702		0,700		0,686		0,795	
MP	59,7%	0,679	61,3%	0,696	60,5%	0,660	71,4%	0,719	70,2%	0,702	70,2%	0,711	70,2%	0,715	79,4%	0,794	67,9%
		0,572		0,585		0,596		0,714		0,702		0,706		0,701		0,794	
		0,675		0,594		0,626		0,689		0,693		0,711		0,720		**0,806**	
RF	**68,5%**	**0,695**	60,9%	0,623	63,3%	0,640	68,5%	0,698	71,0%	0,678	71,0%	0,700	71,0%	0,715	**80,2%**	**0,798**	69,3%
		0,685		0,609		0,633		0,694		0,695		0,705		0,718		**0,802**	
		0,522		0,527		0,528		0,758		0,765		0,756		0,759		**0,802**	
SVM	65,3%	0,728	64,5%	0,716	63,3%	0,700	72,2%	0,673	73,0%	0,782	73,0%	0,676	72,6%	0,682	**80,2%**	**0,803**	70,5%
		0,628		0,625		0,617		0,714		0,722		0,715		0,719		**0,802**	
Ac. Média	63,7%	—	62,5%	—	63,3%	—	70,9%	—	71,7%	—	71,3%	—	72,5%	—	78,1%	—	—

Na Tabela 13 pode-se observar os resultados para a classificação do conjunto de dados com seleção de atributos pelo método cápsula mais as notas das duas avaliações presenciais. Com a utilização da seleção de atributos o desempenho melhorou de forma significativa, e todos os algoritmos classificadores elevaram as suas taxas de acurácia em relação aos resultados para o conjunto de dados original. Destaca-se a melhoria de desempenho do *Multilayer Perceptron* que obteve a melhor média para os oito períodos 73,6% de acurácia. Na inserção da nota da primeira avaliação presencial, a acurácia obtida foi de 76,2%, aumento de 9,7% em relação aos testes realizados no conjunto de dados original.

Tabela 13 – Resultados para a classificação do conjunto usando a técnica de seleção de atributos pelo método cápsula mais notas parciais em séries temporais

Algoritmo	**1**		**2**		**3**		**4**		**5**		**6**		**7**		**8**		**Média**
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*ac.*	*F*	*Ac.*	*F*	*Ac.*
		0,539		0,577		0,667		0,706		0,723		0,739		0,744		0,777	
AB	64,1%	0,706	65,7%	0,712	67,7%	0,688	71,8%	0,729	71,0%	0,695	73,0%	0,720	75,0%	0,756	77,8%	0,779	70,8%
		0,625		0,647		0,677		0,718		0,709		0,729		0,750		0,778	
		0,589		0,614		0,647		0,661		0,686		0,719		0,718		0,802	
BN	65,7%	0,706	66,5%	0,705	66,5%	0,682	68,5%	0,707	70,2%	0,715	72,6%	0,732	73,4%	0,748	79,4%	0,787	70,4%
		0,650		0,661		0,665		0,685		0,701		0,726		0,734		0,794	
		0,509		0,426		0,498		0,650		0,648		0,664		0,667		0,791	
IBk	58,1%	0,634	56,5%	0,649	55,2%	0,596	65,3%	0,656	64,9%	0,651	66,1%	0,659	66,9%	0,672	78,6%	0,782	64,0%
		0,574		0,541		0,541		0,653		0,649		0,661		0,669		0,786	
		0,517		0,463		0,645		**0,740**		0,720		0,730		0,712		0,778	
J48	60,9%	0,671	58,9%	0,667	64,1%	0,637	**76,2%**	**0,781**	73,0%	0,739	75,0%	0,767	72,6%	0,738	78,6%	0,794	69,9%
		0,597		0,568		0,641		**0,761**		0,730		0,749		0,726		0,786	
		0,626		0,619		**0,705**		0,723		0,750		**0,766**		**0,754**		0,802	
JRip	**68,1%**	**0,723**	63,7%	0,654	**71,0%**	**0,714**	73,4%	0,744	74,2%	0,733	**75,8%**	**0,750**	**75,0%**	**0,466**	80,2%	0,803	72,7%
		0,676		0,637		**0,710**		0,734		0,741		**0,758**		**0,750**		0,802	
		0,604		0,611		0,700		0,742		**0,744**		0,736		0,741		**0,838**	
MP	67,7%	0,728	68,1%	0,730	70,6%	0,711	74,2%	0,742	**75,0%**	**0,756**	74,6%	0,755	73,8%	0,735	**84,7%**	**0,855**	**73,6%**
		0,668		0,673		0,706		0,742		**0,750**		0,746		0,738		**0,847**	
		0,584		0,583		0,621		0,750		0,712		0,734		**0,746**		0,805	
RF	63,3%	0,671	61,3%	0,639	64,5%	0,667	75,0%	0,750	71,0%	0,707	73,4%	0,734	**75,0%**	**0,754**	80,6%	0,808	70,5%
		0,629		0,612		0,644		0,750		0,710		0,734		**0,750**		0,806	
		0,543		**0,596**		0,600		0,774		0,767		0,769		0,769		0,805	
SVM	66,1%	0,731	**69,4%**	**0,753**	67,7%	0,730	73,8%	0,689	73,0%	0,679	73,4%	0,686	73,4%	0,686	79,8%	0,792	72,1%
		0,640		**0,677**		0,667		0,730		0,722		0,726		0,726		0,798	
Ac. Média	64,3%	—	63,8%	—	65,9%	—	72,3%	—	71,5%	—	73,0%	—	73,1%	—	80,0%	—	—

A Tabela 14 apresenta a comparação dos resultados médios em cada período de tempo obtidos pelas três abordagens apresentadas neste experimento. A coluna Média representa o aumento médio da acurácia obtido pelos classificadores utilizando o conjunto de dados original mais as notas parciais, e a técnica de seleção de atributos mais as notas parciais em relação ao conjunto de dados original.

Tabela 14 – Comparação dos resultados das três abordagens propostas no experimento com séries temporais

Bases	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Média**
Conjunto Original	63,7%	62,5%	63,3%	62,8%	64,4%	63,6%	66,8%	67,2%	—
Conjunto Original mais Notas parciais	63,7%	62,5%	63,3%	70,9%	**71,7%**	71,3%	72,5%	78,1%	7,9%
Seleção de Atributos mais Notas Parciais	**64,3%**	**63,8%**	**65,9%**	**72,3%**	71,5%	**73,0%**	**73,1%**	**80,0%**	9,0%

O aumento médio informado na Tabela 14 é referente ao intervalo entre os períodos 4 e 8, pois é a partir do quarto período que são inseridas as notas parciais. É possível observar que no conjunto original e no conjunto original mais notas parciais, os três primeiros resultados são iguais, isso porque os dados são os mesmos. Para o conjunto que utiliza a seleção de atributos mais as notas parciais, houve aumento da acurácia nos três primeiros períodos.

Utilizando o conceito de séries temporais, as inferências relativas ao desempenho dos estudantes obtiveram 68,1% de acurácia logo no primeiro corte, e após a inserção da nota da primeira avaliação presencial chegou a 76,2% de acurácia, demonstrando a sua viabilidade já em períodos iniciais do curso. Essas inferências, disponibilizadas desde o início do curso, poderiam auxiliar gestores, professores e tutores no acompanhamento individual de estudantes e no desenvolvimento de estratégias pedagógicas que possam melhorar o desempenho.

4.5.5 Experimento 5 - Utilizando Base Antiga para Classificar Novos Conjuntos

Nos experimentos anteriores foi utilizada a técnica *k-fold cross validation* para divisão dos conjuntos de treinamento e teste. Esta seção apresenta experimentos que utilizam os dados das mesmas disciplinas dos cursos de Administração, Matemática e Física ofertadas no semestre anterior como conjunto de dados para treinamento dos algoritmos classificadores e construção do modelo de predição. E para a realização dos testes são utilizados os conjuntos de dados das disciplinas ofertadas no semestre atual. Neste experimento o curso de Química não foi contemplado devido ao fato de não haver oferta anterior da disciplina utilizada neste estudo. Nenhum dos trabalhos pesquisados fez uso deste método para treinamento de algoritmos classificadores utilizando conjuntos de dados retirados de cursos de graduação à distância ofertados em AVA's.

O objetivo deste experimento é analisar o desempenho do modelo de previsão construído com dados de semestres anteriores para predizer o desempenho de estudantes em semestres atuais. A composição dos conjuntos de dados utilizados para treinamento dos classificadores neste experimento é apresentada na Tabela 15.

Tabela 15 – Relação dos conjuntos de dados obtidos de semestres anteriores e suas respectivas classes

Conjunto	**Número total de estudantes**	**Número de estudantes na Classe Aprovado**	**Número de estudantes na Classe Reprovado**
Administração	182 (100%)	86 (47,3%)	96 (52,7%)
Física	87 (100%)	47 (54,0%)	40 (46,0%)
Matemática	54 (100%)	33 (61,1%)	21 (38,9%)

Podemos observar na Tabela 16 que os algoritmos treinados com os dados do semestre anterior tiveram um desempenho significativo quando aplicados aos cursos de Matemática e Física atingindo 88,4% e 79,5% de acurácia respectivamente. Mas no curso de Administração esta abordagem não se mostrou eficiente, onde o melhor resultado dos classificadores utilizados atingiu apenas 55,6% de acurácia. Analisando os valores da F-Measure para a classe Aprovado no curso de Administração, podemos concluir que os algoritmos tiveram dificuldades em classificar corretamente os exemplos desta classe.

Tabela 16 – Resultados para a classificação dos conjuntos de dados originais utilizando modelo construído com dados do semestre anterior

Algoritmo	**Administração**		**Física**		**Matemática**	
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*
AB	54,0%	0,136 0,687 0,420	70,5%	0,700 0,710 0,704	**88,4%**	**0,915** **0,818** **0,884**
BN	53,3%	0,109 0,687 0,407	69,7%	0,694 0,699 0,696	85,5%	0,889 0,792 0,858
IBk	**55,6%**	**0,295** **0,676** **0,492**	64,8%	0,645 0,650 0,647	72,5%	0,787 0,612 0,731
J48	54,0%	0,095 0,692 0,403	68,0%	0,678 0,683 0,680	79,7%	0,841 0,720 0,802
JRip	52,8%	0,093 0,681 0,397	**79,5%**	**0,847** **0,691** **0,787**	82,6%	0,867 0,750 0,829
MP	51,6%	0,000 0,681 0,351	77,0%	0,816 0,696 0,770	73,9%	0,780 0,679 0,748
RF	52,4%	0,078 0,679 0,388	76,2%	0,775 0,748 0,765	87,0%	0,901 0,809 0,872
SVM	55,2%	0,213 0,687 0,458	54,9%	0,433 0,626 0,507	84,1%	0,882 0,756 0,841

Com o objetivo de verificar os motivos da baixa acurácia obtida pelos algoritmos classificadores no conjunto de dados do curso de Administração, foi aplicado o algoritmo *InfoGainAttributeEval*[3] nas bases de dados de treinamento (conjunto de dados anterior) e teste (conjunto de dados atual) dos três cursos estudados neste experimento. Na Tabela 17 estão relacionados os 8 atributos que possuem maior influência na classificação em cada um dos conjuntos de dados. Estes atributos foram ranqueados e são apresentados em ordem decrescente de influência. De acordo com os resultados apresentados, pode-se concluir que quanto maior a semelhança na relação, presença e posição dos atributos, melhores serão os resultados da classificação. Para facilitar a visualização, os atributos que se repetem foram destacados na coluna Conjunto de Dados Atual.

Tabela 17 – Atributos retornados pelo algoritmo InfoGainAttributeEval

	Conjuntos de Dados Anterior	**Conjuntos de Dados Atual**
Atributos Administração	up_tarefas	**acesso_curso**
	acesso_curso	acesso_recurso
	ad_post_forum	**acesso_forum**
	acesso_forum	**up_tarefas**
	acesso_discussao_forum	**page_tarefas**
	page_tarefas	acesso_video
	acesso_inf	**acesso_discussao_forum**
	acesso_material	acesso_user
Atributos Física	up_tarefas	**up_tarefas**
	acesso_forum	**page_tarefas**
	page_tarefas	**acesso_curso**
	acesso_material	**acesso_recurso**
	acesso_curso	acesso_video
	ad_post_forum	**acesso_forum**
	acesso_recurso	acesso_discussao_forum
	acesso_user	**ad_post_forum**
Atributos Matemática	up_tarefas	**up_tarefas**
	acesso_material	**acesso_material**
	upd_post_forum	**page_tarefas**
	acesso_discussao_forum	**acesso_curso**
	acesso_curso	**acesso_forum**
	page_tarefas	**ad_post_forum**
	ad_post_forum	**acesso_discussao_forum**
	acesso_forum	acesso_user

Pode-se observar na Tabela 17 que no conjunto de dados do curso de Administração, apenas 5 dos 8 atributos mais influentes se repetem, e não necessariamente nas mesmas posições. No curso de Matemática, que obteve melhores valores de acurácia, 7 dos 8 atributos se repetem e 2 dos atributos estão nas mesmas posições de influência (up_tarefas e acesso_material).

[3] Utilizado para avaliar o quanto um atributo influência o critério de classificação do algoritmo.

4.5.6 Experimento 6 - Classificador em Cascata

Este experimento utiliza a abordagem Classificador em Cascata, abordagem que, dentre os trabalhos pesquisados na área de educacional, ainda não havia sido empregada. Este método consiste em inserir a saída do algoritmo classificador como um atributo adicional no conjunto de dados original, como se pode observar na Figura 16.

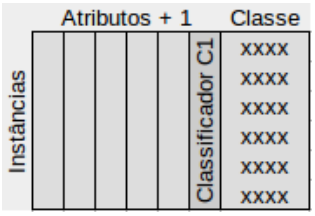

Figura 16 – Abordagem Classificador em Cascata

O experimento foi realizado da seguinte forma, o conjunto de dados original de cada curso foi submetido aos oito algoritmos classificadores utilizados nesta pesquisa. Para cada conjunto de dados são construídos oito modelos de classificação, cada exemplo presente nos conjuntos de dados gera uma saída, que é o rótulo da classe predita (Aprovado ou Reprovado). Estes rótulos são as saídas '*Classificador C1*', ou seja, o atributo adicional que foi inserido ao conjunto original de dados.

Como são oito classificadores foram gerados oito novos conjuntos de dados para cada curso (representados nas Tabelas 18, 20, 22 e 24 como '*C_NomeDoAlgoritmo*').

O objetivo deste experimento é verificar o desempenho dos modelos utilizando a abordagem de Classificador em Cascata.

A Tabela 18 apresenta os resultados dos algoritmos aplicados às oito bases de dados geradas com a saída dos classificadores como atributo adicional para o curso de Administração. O melhor resultado obtido foi de 75,0% de acurácia pelo algoritmo *Multilayer Perceptron* utilizando a base de dados com a adição da saída gerada pelo próprio *Multilayer Perceptron*. Resultado que superou o melhor desempenho obtido pelos classificadores no conjunto de dados original.

Tabela 18 – Resultados para a classificação no conjunto de dados utilizando classificador em cascata do Curso de Administração

Algoritmos	C1_AB		C1_BN		C1_IBk		C1_J48		C1_JRip		C1_MP		C1_RF		C1_SVM	
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*
		0,669		0,667		**0,664**		**0,664**		**0,664**		**0,664**		**0,664**		**0,664**
AB	**67,7%**	**0,685**	67,3%	0,680	**67,7%**	**0,690**	**67,7%**	**0,690**	**67,7%**	**0,690**	**67,7%**	**0,690**	**67,7%**	**0,690**	**67,7%**	**0,690**
		0,677		0,673		**0,677**		**0,677**		**0,677**		**0,677**		**0,677**		**0,677**
		0,706		**0,725**		0,722		0,727		0,719		0,700		0,700		0,714
BN	70,2%	0,697	**72,2%**	**0,718**	71,4%	0,705	71,8%	0,708	71,0%	0,700	70,6%	0,711	70,6%	0,711	71,0%	0,705
		0,701		**0,722**		0,713		0,717		0,709		0,705		0,705		0,709
		0,649		0,634		**0,730**		0,655		0,616		0,655		0,647		0,653
IBk	67,7%	0,701	66,5%	0,691	**74,6%**	**0,760**	66,9%	0,682	65,3%	0,684	67,7%	0,697	66,5%	0,682	66,9%	0,685
		0,676		0,664		**0,746**		0,669		0,651		0,677		0,665		0,669
		0,650		0,681		0,653		0,683		**0,700**		0,696		0,641		0,641
J48	66,5%	0,680	69,4%	0,705	66,1%	0,669	68,1%	0,680	**70,6%**	**0,711**	69,8%	0,699	65,7%	0,672	65,7%	0,672
		0,665		0,693		0,661		0,681		**0,706**		0,698		0,657		0,657
		0,745		0,738		0,724		0,727		0,725		0,714		0,725		0,719
JRip	**73,8%**	**0,730**	73,4%	0,730	71,8%	0,711	72,2%	0,716	72,2%	0,718	70,6%	0,697	72,2%	0,718	71,4%	0,708
		0,737		0,734		0,717		0,721		0,722		0,705		0,722		0,713
		0,706		0,700		0,725		0,661		0,707		**0,739**		0,706		0,721
MP	72,2%	0,736	71,4%	0,726	73,4%	0,742	68,1%	0,700	70,6%	0,704	**75,0%**	**0,760**	72,2%	0,736	72,6%	0,730
		0,721		0,724		0,734		0,681		0,706		**0,750**		0,721		0,726
		0,712		0,717		**0,744**		0,719		0,706		0,688		0,728		0,701
RF	71,0%	0,707	71,0%	0,702	**73,4%**	**0,723**	71,4%	0,708	70,2%	0,697	68,1%	0,675	71,4%	0,698	69,4%	0,686
		0,710		0,709		**0,733**		0,713		0,701		0,681		0,712		0,693
		0,702		**0,741**		0,703		0,725		0,724		0,707		0,720		0,678
SVM	70,6%	0,709	**73,8%**	**0,735**	71,8%	0,731	72,2%	0,718	71,8%	0,711	70,6%	0,704	71,8%	0,715	70,2%	0,722
		0,706		**0,738**		0,718		0,722		0,717		0,706		0,718		0,701

A Tabela 19 mostra a comparação dos resultados obtidos pelos classificadores utilizando a base de dados original e a base que possui a saída do classificador como atributo adicional. Como é possível observar, todos os algoritmos utilizados tiveram aumento no desempenho, ficando com o *Multilayer Perceptron* o maior crescimento percentual, chegando a 12,5%.

Tabela 19 – Comparação dos resultados da classificação no conjunto de dados do curso de Administração utilizando o conjunto de dados original e com classificador em cascata

Bases	**AB**	**BN**	**IBk**	**J48**	**JRip**	**MP**	**RF**	**SVM**
Conjunto de Dados Original	67,3%	71,8%	66,9%	63,3%	66,1%	62,5%	72,2%	67,3%
Conjunto de Dados com Classificador em Cascata	**67,7%**	**72,2%**	**74,6%**	**70,6%**	**73,8%**	**75,0%**	**73,4%**	**73,8%**

Na Tabela 20 são apresentados os resultados dos algoritmos aplicados às oito bases de dados geradas com a saída dos classificadores como atributo adicional para o curso de Física. O melhor resultado obtido foi de 90,2% de acurácia pelo algoritmo SVM utilizando a base de dados com a adição da saída gerada pelo classificador *Multilayer Perceptron*. Este valor foi semelhante ao obtido pelos classificadores utilizando o conjunto de dados original.

Tabela 20 – Resultados para a classificação no conjunto de dados utilizando classificador em cascata do Curso de Física

Algoritmos	C1_AB		C1_BN		C1_IBk		C1_J48		C1_JRip		C1_MP		C1_RF		C1_SVM	
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*
		0,831		0,839		0,831		0,831		0,831		**0,889**		0,841		0,831
AB	78,7%	0,711	79,5%	0,719	78,7%	0,711	78,7%	0,711	78,7%	0,711	**86,1%**	**0,809**	79,5%	0,713	78,7%	0,711
		0,785		0,793		0,785		0,785		0,785		**0,860**		0,791		0,785
		0,883		0,876		0,876		0,876		0,876		**0,889**		**0,890**		0,870
BN	82,5%	0,800	84,4%	0,791	84,4%	0,791	84,4%	0,791	84,4%	0,791	**86,1%**	**0,813**	**86,1%**	**0,809**	83,6%	0,778
		0,851		0,843		0,843		0,843		0,843		**0,860**		**0,859**		0,835
		0,828		0,822		**0,896**		0,831		0,828		0,877		0,819		0,819
IBk	79,5%	0,747	78,7%	0,735	**86,9%**	**0,822**	80,3%	0,765	79,5%	0,747	85,2%	0,816	78,7%	0,740	78,7%	0,740
		0,797		0,788		**0,868**		0,805		0,797		0,853		0,789		0,789
		0,808		0,805		0,834		0,829		0,813		**0,882**		0,810		0,824
J48	76,2%	0,688	75,4%	0,667	79,5%	0,731	78,7%	0,717	77,0%	0,702	**85,2%**	**0,804**	76,2%	0,681	77,9%	0,703
		0,762		0,752		0,795		0,786		0,770		**0,852**		0,761		0,777
		0,855		0,850		0,864		0,855		0,855		**0,908**		0,850		0,855
JRip	81,1%	0,729	80,3%	0,714	82,0%	0,732	81,1%	0,729	81,1%	0,729	**88,5%**	**0,848**	80,3%	0,714	81,1%	0,729
		0,807		0,798		0,813		0,807		0,807		**0,885**		0,798		0,807
		0,851		0,889		0,882		**0,901**		0,867		0,877		0,896		0,846
MP	82,0%	0,771	86,1%	0,813	85,2%	0,804	**87,7%**	**0,839**	83,6%	0,787	85,2%	0,816	86,9%	0,822	81,1%	0,758
		0,820		0,860		0,852		**0,877**		0,836		0,853		0,868		0,812
		0,873		0,872		**0,893**		0,868		0,881		0,883		0,886		0,873
RF	83,6%	0,767	83,6%	0,773	**86,1%**	**0,800**	82,8%	0,753	84,4%	0,776	85,2%	0,800	85,2%	0,791	83,6%	0,767
		0,833		0,834		**0,857**		0,824		0,840		0,851		0,849		0,833
		0,861		0,876		0,838		0,859		0,846		**0,921**		0,886		0,885
SVM	82,8%	0,774	84,4%	0,791	80,3%	0,750	82,0%	0,750	80,3%	0,727	**90,2%**	**0,870**	85,2%	0,791	85,2%	0,795
		0,828		0,843		0,804		0,817		0,80		**0,901**		0,849		0,850

Mas como pode ser observado na Tabela 21, seis dos oito algoritmos classificadores obtiveram aumento de desempenho na predição utilizando o conjunto de dados com o classificador em cascata. O algoritmo IBk por exemplo, passou de 78,7% de acurácia para 86,9%.

Tabela 21 – Comparação dos resultados da classificação no conjunto de dados do curso de Física utilizando o conjunto de dados original e com classificador em cascata

Bases	**AB**	**BN**	**IBk**	**J48**	**JRip**	**MP**	**RF**	**SVM**
Conjunto de Dados Original	82,0%	85,2%	78,7%	82,0%	81,1%	**90,2%**	**86,9%**	86,1%
Conjunto de Dados com Classificador em cascata	**86,1%**	**86,1%**	**86,9%**	**85,2%**	**88,5%**	87,7%	86,1%	**90,2%**

Na Tabela 22 pode-se observar os resultados dos algoritmos aplicados às oito bases de dados geradas com a saída dos classificadores como atributo adicional para o curso de Matemática. O melhor resultado obtido foi de 88,4% de acurácia pelo algoritmo *Random Forest* utilizando as bases de dados com a adição da saída gerada pelos demais classificadores com a exceção do próprio *Random Forest*. Este valor foi superior ao obtido pelo próprio *Random Forest* utilizando o conjunto de dados original, que foi de 87,0% de acurácia.

Tabela 22 – Resultados para a classificação no conjunto de dados utilizando classificador em cascata do Curso de Matemática

Algoritmos	C1_AB		C1_BN		C1_IBk		C1_J48		C1_JRip		C1_MP		C1_RF		C1_SVM	
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*
AB	**81,2%**	**0,863** **0,698** **0,810**	**81,2%**	**0,863** **0,698** **0,810**	**81,2%**	**0,863** **0,698** **0,810**	**81,2%**	**0,863** **0,698** **0,810**	**81,2%**	**0,863** **0,698** **0,810**	**81,2%**	**0,863** **0,698** **0,810**	**81,2%**	**0,863** **0,698** **0,810**	**81,2%**	**0,863** **0,698** **0,810**
BN	82,6%	0,872 0,727 0,826	**84,1%**	**0,884** **0,744** **0,840**	82,6%	0,872 0,727 0,826	82,6%	0,872 0,727 0,826	82,6%	0,872 0,727 0,826	82,6%	0,872 0,727 0,826	**84,1%**	**0,884** **0,744** **0,840**	82,6%	0,872 0,727 0,826
IBk	78,3%	0,831 0,694 0,788	76,8%	0,822 0,667 0,773	81,2%	0,860 0,711 0,713	81,2%	0,857 0,723 0,815	**82,6%**	**0,870** **0,739** **0,828**	78,3%	0,835 0,681 0,786	78,3%	0,835 0,681 0,786	78,3%	0,831 0,694 0,788
J48	75,4%	0,817 0,622 0,755	75,4%	0,817 0,622 0,755	75,4%	0,817 0,622 0,755	79,7%	0,851 0,682 0,797	75,4%	0,817 0,622 0,755	75,4%	0,817 0,622 0,755	**81,2%**	**0,860** **0,711** **0,813**	75,4%	0,817 0,622 0,755
JRip	69,9%	0,764 0,571 0,703	69,9%	0,764 0,571 0,703	69,9%	0,764 0,571 0,703	**72,5%**	**0,791** **0,596** **0,729**	66,7%	0,742 0,531 0,674	69,9%	0,764 0,571 0,703	69,9%	0,764 0,571 0,703	69,9%	0,764 0,571 0,703
MP	75,4%	0,809 0,653 0,759	72,5%	0,787 0,612 0,731	69,9%	0,759 0,588 0,704	**78,3%**	**0,835** **0,681** **0,786**	73,9%	0,795 0,640 0,746	72,5%	0,787 0,612 0,731	75,4%	0,809 0,653 0,759	71,0%	0,773 0,600 0,718
RF	**88,4%**	**0,918** **0,800** **0,881**	**88,4%**	**0,918** **0,800** **0,881**	**88,4%**	**0,918** **0,800** **0,881**	**88,4%**	**0,918** **0,800** **0,881**	**88,4%**	**0,918** **0,800** **0,881**	**88,4%**	**0,918** **0,800** **0,881**	87,0%	0,907 0,780 0,867	**88,4%**	**0,918** **0,800** **0,881**
SVM	81,2%	0,863 0,698 0,810	82,6%	0,872 0,727 0,826	81,2%	0,860 0,711 0,813	82,4%	0,870 0,739 0,828	82,6%	0,870 0,739 0,828	81,2%	0,860 0,711 0,813	**85,5%**	**0,896** **0,762** **0,853**	84,1%	0,882 0,756 0,841

A Tabela 23 mostra a comparação dos resultados obtidos pelos classificadores utilizando a base de dados original e a base que possui a saída do classificador como atributo adicional para o curso de Matemática. Como é possível observar 6 dos algoritmos classificadores utilizados obtiveram aumento das acurácias obtidas.

Tabela 23 – Comparação dos resultados da classificação no conjunto de dados do curso de Matemática utilizando o conjunto de dados original e com classificador em cascata

Bases	**AB**	**BN**	**IBk**	**J48**	**JRip**	**MP**	**RF**	**SVM**
Conjunto de Dados Original	78,3%	82,6%	79,7%	73,9%	**76,8%**	78,3%	87,0%	84,1%
Conjunto de Dados com Classificador em Cascata	**81,2%**	**84,1%**	**82,6%**	**81,2%**	72,5%	78,3%	**88,4%**	**85,5%**

A Tabela 24 apresenta os resultados dos algoritmos aplicados às oito bases de dados geradas com a saída dos classificadores como atributo adicional para o curso de Química. O melhor resultado obtido foi 87,01% de acurácia pelos algoritmos *Random Forest* e JRip. Este desempenho superou o melhor resultado obtido pelos algoritmos *AdaBoost*, *Bayes Net* e *Random Forest* no conjunto de dados original, que foi de 80,6% de acurácia.

Tabela 24 – Resultados para a classificação no conjunto de dados utilizando classificador em cascata do Curso de Química

Algoritmos	**C1_AB**		**C1_BN**		**C1_IBk**		**C1_J48**		**C1_JRip**		**C1_MP**		**C1_RF**		**C1_SVM**	
	Ac.	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*	*Ac.*	*F*
AB	71,0%	0,727 0,690 0,709	71,0%	0,727 0,690 0,709	71,0%	0,727 0,690 0,709	71,0%	0,727 0,690 0,709	71,0%	0,727 0,690 0,709	**74,2%**	**0,750** **0,733** **0,742**	71,0%	0,727 0,690 0,709	71,0%	0,727 0,690 0,709
BN	77,4%	0,788 0,759 0,774	77,4%	0,788 0,759 0,774	77,4%	0,788 0,759 0,774	**80,6%**	**0,824** **0,786** **0,805**	**80,6%**	**0,824** **0,786** **0,805**	77,4%	0,788 0,759 0,774	**80,6%**	**0,824** **0,786** **0,805**	74,2%	0,750 0,733 0,742
IBk	58,1%	0,581 0,581 0,581	61,3%	0,625 0,600 0,613	**74,2%**	**0,750** **0,733** **0,742**	71,0%	0,710 0,710 0,710	64,5%	0,667 0,621 0,644	64,5%	0,645 0,645 0,645	61,3%	0,625 0,600 0,613	67,7%	0,688 0,667 0,677
J48	**80,6%**	**0,824** **0,786** **0,805**	**80,6%**	**0,824** **0,786** **0,805**	**80,6%**	**0,824** **0,786** **0,805**	**80,6%**	**0,824** **0,786** **0,805**	**80,6%**	**0,824** **0,786** **0,805**	**80,6%**	**0,824** **0,786** **0,805**	**80,6%**	**0,824** **0,786** **0,805**	**80,6%**	**0,824** **0,786** **0,805**
JRip	83,9%	0,848 0,828 0,838	83,9%	0,848 0,828 0,838	83,9%	0,848 0,828 0,838	**87,1%**	**0,875** **0,867** **0,871**	87,1%	0,875 0,867 0,871	80,6%	0,800 0,813 0,806	83,9%	0,848 0,828 0,838	80,6%	0,800 0,813 0,806
MP	71,0%	0,727 0,690 0,709	67,7%	0,688 0,667 0,677	71,0%	0,727 0,690 0,709	**77,4%**	**0,788** **0,759** **0,774**	71,0%	0,727 0,690 0,709	67,7%	0,688 0,667 0,677	71,0%	0,727 0,690 0,709	71,0%	0,727 0,690 0,709
RF	**87,1%**	**0,882** **0,857** **0,870**	83,9%	0,848 0,828 0,838	83,9%	0,848 0,828 0,838	**87,1%**	**0,882** **0,857** **0,870**	83,9%	0,857 0,815 0,837	**87,1%**	**0,882** **0,857** **0,870**	83,9%	0,857 0,815 0,837	**87,1%**	**0,882** **0,857** **0,870**
SVM	**83,9%**	**0,848** **0,828** **0,838**	77,4%	0,788 0,759 0,774	74,2%	0,733 0,750 0,741	74,2%	0,750 0,733 0,742	74,2%	0,765 0,714 0,740	74,2%	0,750 0,733 0,742	**80,6%**	**0,824** **0,786** **0,805**	77,4%	0,774 0,774 0,774

A Tabela 25 mostra a comparação dos resultados obtidos pelos classificadores utilizando a base de dados original e a base que possui a saída do classificador como atributo adicional para o curso de Química. Dos oito algoritmos classificadores utilizados cinco tiveram aumento de desempenho, dois mantiveram, e apenas o *AdaBoost* obteve desempenho inferior utilizando a abordagem de classificador em cascata.

Tabela 25 – Comparação dos resultados da classificação no conjunto de dados do curso de Química utilizando o conjunto de dados original e com classificador em cascata

Bases	**AB**	**BN**	**IBk**	**J48**	**JRip**	**MP**	**RF**	**SVM**
Conjunto de Dados Original	**80,6%**	80,6%	74,2%	74,2%	74,2%	67,7%	80,6%	74,2%
Conjunto de Dados com Classificador em Cascata	74,2%	80,6%	74,2%	**80,6%**	**87,1%**	**77,4%**	**87,1%**	**83,9%**

5 Conclusões e Trabalhos Futuros

Este trabalho analisou e extraiu conhecimento a partir dos bancos de dados do AVA *Moodle* e do SGA de uma IFES. A possibilidade de prever antecipadamente o desempenho de um estudante em cursos EAD é de grande importância para professores, tutores e gestores, que podem ajustar suas ações pedagógicas para evitar a reprovação destes estudantes. Através dos experimentos realizados, buscou-se obter um modelo que fosse capaz de predizer o desempenho acadêmico a partir dos dados de interações dos estudantes com o AVA.

No experimento 1 buscou-se identificar quais seriam os algoritmos mais adequados ao problema de identificação de estudantes candidatos a reprovação ou a um desempenho abaixo do desejado. Os melhores modelos foram gerados pelo algoritmo *Random Forest* baseado na técnica de Árvore de Decisão, alcançando os melhores resultados, na medida *F-Measure* para a classe Reprovado, em 3 dos 4 cursos analisados.

O experimento 2 abordou o problema de balanceamento de classes identificado em um dos conjuntos de dados utilizados nos experimentos desta pesquisa. Através da utilização da técnica de balanceamento de classes *SMOTE*, foram obtidos melhores resultados em relação aos dados originais. Outra vantagem obtida com a aplicação da técnica de balanceamento de classes foi o aumento de desempenho na predição da classe minoritária Reprovado, que resulta na identificação dos estudantes com maior probabilidade de terem um rendimento abaixo do esperado, e provável reprovação. Ao utilizar o balanceamento de classes, o algoritmo J48, que utiliza a técnica de Árvore de Decisão, obteve o maior aumento de acurácia.

No experimento 3, a identificação das características que possuem maior influência sobre o aprendizado dos estudantes, em cada um dos cursos analisados, e os resultados dos experimentos, comprovaram a eficácia da aplicação da técnica de Seleção de Atributos pela abordagem Cápsula nas bases de dados analisadas. A identificação das características com maior influência no desempenho estudantil mostrou a importância do fórum de discussão em cursos ofertados na modalidade à distância, devido ao maior número de atributos, referente ao fórum, selecionados. Tal resultado evidência a importância do contato aluno/tutor, aluno/professor na construção do conhecimento. Durante o processo de classificação utilizando a técnica de Seleção de Atributos, o algoritmo baseado em Redes Neurais *Multilayer Perceptron* obteve os melhores resultados em relação ao aumento de acurácia quando comparado aos resultados obtidos nos conjuntos de dados originais.

No experimento 4, com a utilização do conceito de séries temporais, buscou-se a identificação precoce de estudantes com maior probabilidade de baixo desempenho. Foram propostas três abordagens, sendo a abordagem que utiliza a Seleção de Atributos pelo método Cápsula com a inserção das notas das avaliações presenciais, a mais recomendada para o problema.

Utilizando o conceito de séries temporais, e realizando os cortes temporais conforme a Figura 5.5, os gestores podem acompanhar o progresso dos estudantes durante a realização do curso. São três *feedbacks* antes da primeira avaliação presencial e mais 4 até a segunda avaliação. Através destes *feedbacks*, os gestores das turmas podem realizar intervenções pedagógicas com o objetivo de melhor o desempenho dos estudantes.

Com o experimento 5 tentou-se avaliar a possibilidade da utilização de dados anteriores para realizar o treinamento dos algoritmos classificadores para a predição de desempenho de estudantes de cursos em andamento. Esta abordagem foi bem-sucedida em dois dos três testes realizados, e na busca por verificar os motivos da baixa acurácia obtida em um dos testes, foi apontado o possível problema. Ao aplicar o algoritmo *InfoGainAttributeEval*, que é utilizado para avaliar o quanto um atributo influência o critério de classificação do algoritmo, os resultados retornados tiveram pouca semelhança no ranqueamento dos atributos, com maior influencia para os dois conjuntos de dados, o de treinamento e teste de curso de Administração. Os resultados no curso de Administração vão na contramão dos resultados retornados para os conjuntos de treinamento e teste dos cursos de Matemática e Física, que tiveram maior semelhança no ranqueamento dos atributos, e consequentemente melhores taxas de acurácia.

O experimento 6, teve como objetivo aplicar uma técnica diferente das que se utilizam habitualmente. A técnica de classificador em cascata se mostrou promissora, alcançando ótimas elevações no desempenho dos algoritmos classificadores. As técnicas que melhor se adaptaram a abordagem foram Árvore de Decisão com o algoritmo J48 e Máquinas de Vetores com o algoritmo SVM. Ambos tiveram aumento no desempenho nos 4 conjuntos de dados.

Após a apresentação e discussão dos resultados obtidos pode-se concluir que as principais contribuições deste trabalho foram à utilização da técnica de seleção de atributos pelo método cápsula que, dentre os trabalhos pesquisados, não havia sido ainda empregada e alcançou a notável marca de 90,2% de acurácia. A utilização do conceito de séries temporais onde, os cortes temporais indicados mais a inserção das notas das avaliações presenciais no conjunto de atributos, permitem aos gestores das turmas obterem feedbacks quinzenais, facilitando o acompanhamento e intervenção. A utilização de dados de semestres anteriores para realizar o treinamento dos algoritmos classificadores para a predição de desempenho acadêmico de estudantes de cursos em andamento que, dentre os trabalhos pesquisados, não havia sido empregada. A apresentação da técnica de classificador em cascata, ainda não utilizada em nenhum trabalho pesquisado na área educacional, chegando a taxas de acurácia acima de 90%.

Como trabalhos futuros pretende-se:

- Aplicar as abordagens utilizadas neste trabalho em conjuntos de dados de fontes diversas com objetivo de prever o desempenho de estudantes;
- Automatizar a abordagem de classificador em cascata, de forma que novos conjuntos de dados sejam criados já com a saída do classificador;

- Testar a abordagem de classificador em cascata com um número maior de saídas;
- Realizar experimentos utilizando o conceito de séries temporais aplicando a abordagem de classificador em cascata;
- Realizar toda a sequência de experimentos desta dissertação utilizando conjuntos de dados com número maior de classes, como por exemplo, três classes.

Referências

ACHARYA, A.; SINHA, D. Application of feature selection methods in educational data mining. *International Journal of Computer Applications*, Foundation of Computer Science, v. 103, n. 2, 2014.

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

ALPAYDIN, E. *Introduction to machine learning.* MIT Press, 2010.

BAKER, R. et al. Mineração de dados educacionais: Conceitos, técnicas, ferramentas e aplicações. *Jornada de Atualização em Informática na Educação*, v. 1, p. 1–29, 2012.

BAKER, R. S. J. de; ISOTANI, S.; CARVALHO, A. M. J. B. de. Mineração de dados educacionais: oportunidades para o brasil. *Revista Brasileira de Informática na Educação*, v. 19, n. 2, 2011.

BOUCKAERT, R. R. et al. Weka manual for version 3-7-13. *Hamilton, New Zealand*, 2015.

BREIMAN, L. Random forests. *Machine learning*, Springer, v. 45, n. 1, p. 5–32, 2001.

CAMBRUZZI, W. L. *GVwise: uma aplicação de learning analytics para a redução da evasão na educação a distância*. Dissertação (Mestrado), 2014.

CARDOSO, O. N. P.; MACHADO, R. T. M. Gestão do conhecimento usando data mining estudo de caso na universidade federal de lavras. 2008.

CHAROUX, O. M. G. Metodologia: processo de produção, registro e relato do conhecimento. *São Paulo: DVS*, 2006.

CHAWLA, N. V. et al. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, p. 321–357, 2002.

CHEESEMAN, P.; STUTZ, J. Bayesian classification (autoclass): Theory and results. *In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy (Eds.), Advances in Knowledge Discovery and Data Mining*, Menlo Park, CA: American Association for Artificial Intelligence, p. 153–180, 1996.

COHEN, W. W. Fast effective rule induction. p. 115–123, 1995.

DEKKER, G. W.; PECHENIZKIY, M.; VLEESHOUWERS, J. M. Predicting students drop out: A case study. *International Working Group on Educational Data Mining*, ERIC, 2009.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. *AI magazine*, v. 17, n. 3, p. 37, 1996.

FERREIRA, G. Investigação acerca dos fatores determinantes para a conclusão do ensino fundamental utilizando mineração de dados educacionais no censo escolar da educação básica do inep 2014. v. 4, n. 1, p. 1034, 2015.

FONSECA, S. O. da; NAMEN, A. A. Mineração em bases de dados do inep: uma análise exploratória para nortear melhorias no sistema educacional brasileiro. *Educação em Revista*, SciELO Brasil, v. 32, n. 1, 2016.

FRANÇA, G. Curso de preparação de monitores para a educação a distância. *São Paulo: Rede Brasileira de EAD LTDA*, 2000.

FREITAS, A. A. *Data mining and knowledge discovery with evolutionary algorithms*. Springer Science, 1998.

FREUND, Y.; SCHAPIRE, R. E. et al. Experiments with a new boosting algorithm. v. 96, p. 148–156, 1996.

GARCÍA, E. et al. A collaborative educational association rule mining tool. *The Internet and Higher Education*, Elsevier, v. 14, n. 2, p. 77–88, 2011.

GOLDSCHMIDT, R.; PASSOS, E. *Data Mining: um guia prático*. Gulf Professional Publishing, 2005.

GOTARDO, R.; CEREDA, P. R. M.; JUNIOR, E. R. H. Predição do desempenho do aluno usando sistemas de recomendação e acoplamento de classificadores. v. 24, n. 1, p. 657, 2013.

GOTTARDO, E. *Estimativa de desempenho acadêmico de estudantes em um AVA utilizando técnicas de mineração de dados*. Dissertação (Mestrado), 2012.

GUÉRCIO, H. et al. Análise do desempenho estudantil na educação a distância aplicando técnicas de mineração de dados. v. 3, n. 1, p. 641, 2014.

HÄMÄLÄINEN, W.; VINNI, M. Classifiers for educational data mining. *Handbook of Educational Data Mining, Chapman & Hall/CRC Data Mining and Knowledge Discovery Series*, p. 57–71, 2010.

HAN, J.; KAMBER, M.; PEI, J. *Data mining: concepts and techniques*. Elsevier, 2011.

HAND, D. J.; MANNILA, H.; SMYTH, P. *Principles of data mining*. MIT press, 2001.

HECKERMAN, D. A tutorial on learning with bayesian networks. *Microsoft Research*, p. 57, 1995.

HUANG, S. *Predictive modeling and analysis of student academic performance in an engineering dynamics course*. Tese (Doutorado), 2011.

KAMPFF, A. J. C. *Mineração de dados educacionais para geração de alertas em ambientes virtuais de aprendizagem como apoio à prática docente*. Tese (Doutorado), 2009.

KAMPFF, A. J. C. et al. Identificação de perfis de evasão e mau desempenho para geração de alertas num contexto de educação a distância. *RELATEC*, Universidad de Extremadura, 2014.

KOHAVI, R.; SOMMERFIELD, D.; DOUGHERTY, J. Data mining using mlc++, a machine learning library in c++. In: *Proceedings of the 8th International Conference on Tools with Artificial Intelligence*. Washington, DC, USA: IEEE Computer Society, 1996. (ICTAI '96), p. 234–. ISBN 0-8186-7686-8. Disponível em: <http://dl.acm.org/citation.cfm?id=850949.853584>.

KONAR, A. *Artificial intelligence and soft computing: behavioral and cognitive modeling of the human brain*. CRC press, 1999.

KOTSIANTIS, S. B.; PIERRAKEAS, C.; PINTELAS, P. E. Preventing student dropout in distance learning using machine learning techniques. p. 267–274, 2003.

LUO, J. et al. Predicting student grade based on free-style comments using word2vec and ann by considering prediction results obtained in consecutive lessons. *In Proc. 8th Intl. Conf. on Educational Data Mining*, p. 396–399, 2015.

MAES, P. Artificial life meets entertainment: lifelike autonomous agents. *Communications of the ACM*, ACM, v. 38, n. 11, p. 108–114, 1995.

MAIMON, O.; ROKACH, L. *Data mining and knowledge discovery handbook*. Springer, 2005. v. 2.

MANHÃES, L. M. B. *PREDIÇÃO DO DESEMPENHO ACADÊMICO DE GRADUANDOS UTILIZANDO MINERAÇÃO DE DADOS EDUCACIONAIS*. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, 2015.

MARQUEZ-VERA, C.; ROMERO, C.; VENTURA, S. Predicting school failure using data mining. p. 271–276, 2011.

MITCHELL, T. M. *Machine learning*. McGraw-Hill, Inc., New York, NY, 1997.

MOODLE. *Moodle.net Courses and Content*. 2016. <https://moodle.net/stats/?lang=pt_br>. [Acesso em: 16-Jun-2016].

NANDESHWAR, A.; MENZIES, T.; NELSON, A. Learning patterns of university student retention. *Expert Systems with Applications*, Elsevier, v. 38, n. 12, p. 14984–14996, 2011.

PEÑA-AYALA, A. Educational data mining: A survey and a data mining-based analysis of recent works. *Expert systems with applications*, Elsevier, v. 41, n. 4, p. 1432–1462, 2014.

PLATT, J. C. Fast training of support vector machines using sequential minimal optimization. MIT Press, p. 185–209, January 1998. Disponível em: <http://research.microsoft.com/apps/pubs/default.aspx?id=68391>.

PORTAL, C. *Estratégias para minimizar a evasão e potencializar a permanência em EAD a partir de sistema que utiliza mineração de dados educacionais e learning analytics*. Dissertação (Mestrado), 2016.

PRATI, R. C. et al. Uma experiência no balanceamento artificial de conjuntos de dados para aprendizado com classes desbalanceadas utilizando análise roc. v. 1, p. 28–33, 2003.

QUINLAN, J. R. *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993. ISBN 1-55860-238-0.

RAMASWAMI, M.; BHASKARAN, R. A study on feature selection techniques in educational data mining. *arXiv preprint arXiv:0912.3924*, 2009.

ROMERO, C.; VENTURA, S. Educational data mining: a review of the state of the art. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, Ieee, v. 40, n. 6, p. 601–618, 2010.

ROMERO, C.; VENTURA, S. Data mining in education. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Wiley Online Library, v. 3, n. 1, p. 12–27, 2013.

ROMERO, C.; VENTURA, S.; BRA, P. D. Knowledge discovery with genetic programming for providing feedback to courseware authors. *User Modeling and User-Adapted Interaction*, Springer, v. 14, n. 5, p. 425–464, 2004.

ROMERO-ZALDIVAR, V.-A. et al. Monitoring student progress using virtual appliances: A case study. *Computers & Education*, Elsevier, v. 58, n. 4, p. 1058–1067, 2012.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *Nature*, v. 323, p. 533–536, 1986.

RUSSELL, S.; NORVIG, P. *Inteligência Artificial*. Editora Campus, 2004.

SANTOS, R. N. dos; SIEBRA, C. de A.; OLIVEIRA, E. S. Uma abordagem temporal para identificação precoce de estudantes de graduação a distância com risco de evasão em um ava utilizando árvores de decisão. v. 3, n. 1, p. 262, 2014.

TALEBI, S.; SAYFICAR, A. A. Using educational data mining (edm) to prediction and classify students. *International Journal of Engineering and Computer Science*, v. 3, n. 12, 2014.

WAIKATO. *Weka 3: Data Mining Software in Java*. 2016. <http://www.cs.waikato.ac.nz/ml/weka/>. [Acesso em: 16-Jun-2016].

WENGER, E. *Artificial intelligence and tutoring systems: computational and cognitive approaches to the communication of knowledge*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1987.

WITTEN, I. H.; FRANK, E.; HALL, M. A. *Data Mining: Practical Machine Learning Tools and Techniques*. Third edition. Morgan Kaufmann Publishers Inc., 2011. (The Morgan Kaufmann Series in Data Management Systems).

WIVES, L. K. Tecnologias de descoberta de conhecimento em textos aplicadas à inteligência competitiva. 2002.

WU, X. et al. Top 10 algorithms in data mining. *Knowledge and information systems*, Springer, v. 14, n. 1, p. 1–37, 2008.

ZAFRA, A.; ROMERO, C.; VENTURA, S. Multiple instance learning for classifying students in learning management systems. *Expert Systems with Applications*, Elsevier, v. 38, n. 12, p. 15020–15031, 2011.