UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

CENTRO UNIVERSITÁRIO NORTE DO ESPÍRITO SANTO

DEPARTAMENTO DE COMPUTAÇÃO E ELETRÔNICA

BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

Jadsmila Ferreira Rocha

Aplicação do processo de descoberta de conhecimento na base de dados do ENADE 2017 utilizando linguagem R e o software Power BI

São Mateus, ES

2020

Jadsmila Ferreira Rocha

Aplicação do processo de descoberta de conhecimento na base de dados do ENADE 2017 utilizando linguagem R e o software Power BI

Monografia apresentada ao Colegiado do Curso de Engenharia de Computação do Departamento de Computação e Eletrônica da Universidade Federal do Espírito Santo, campus São Mateus, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Universidade Federal do Espírito Santo – UFES

Departamento de Computação e Eletrônica

Colegiado do Curso de Engenharia de Computação

Orientador: Prof. Dr. Silvia das Dores Rissino

São Mateus, ES

2020

Jadsmila Ferreira Rocha

Aplicação do processo de descoberta de conhecimento na base de dados do ENADE 2017 utilizando linguagem R e o software Power BI/ Jadsmila Ferreira Rocha. – São Mateus, ES, 2020-

40 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Silvia das Dores Rissino

Monografia (PG) – Universidade Federal do Espírito Santo – UFES
Departamento de Computação e Eletrônica
Colegiado do Curso de Engenharia de Computação, 2020.

1. Apriori. 2. KDD. 3. Enade 2017. 4. Mineração de Dados. I. Universidade Federal do Espírito Santo. IV. Aplicação do processo de descoberta de conhecimento na base de dados do ENADE 2017 utilizando linguagem R e o software Power BI

CDU 02:141:005.7

Jadsmila Ferreira Rocha

Aplicação do processo de descoberta de conhecimento na base de dados do ENADE 2017 utilizando linguagem R e o software Power BI

Monografia apresentada ao Colegiado do Curso de Engenharia de Computação do Departamento de Computação e Eletrônica da Universidade Federal do Espírito Santo, campus São Mateus, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Trabalho aprovado. São Mateus, ES, 11 de agosto de 2020:

Prof. Dr. Silvia das Dores Rissino
Orientador

Professor
Convidado 1

Professor
Convidado 2

São Mateus, ES
2020

Dedico este Trabalalho a todos que me apoiaram para que eu pudesse completar esta etapa da minha vida..

Agradecimentos

Agradeço primeiramente a Deus, a minha família, meus amigos, a professova Silvia que me incentivou a sempre continuar e ao Sr. R.

"Ser ruim em alguma coisa é o primeiro passo para se tornar bom em alguma coisa.
(Jake, Hora da Aventura)

Resumo

Grandes volumes de dados são gerados todos os dias na internet e, em sua maioria, são armazenados e subutilizados. Com a necessidade de transformar esses dados em informação útil, surge o processo de Descoberta de Conhecimento em Banco de Dados, o KDD (Knowledge Discovery in Database). Um ramo de estudos dentro deste processo é a Mineração de dados educacionais, MDE. O trabalho em questão consiste em aplicar as etapas do KDD: Pré-processamento, Mineração de Dados e Pós-processamento na base de dados do Enade 2017, desenvolvendo códigos em linguagem R e apresentando os resultados encontrados em *dashboards* criados no Power BI. A etapa de Pré-processamento foi em sua completude executada em linguagem R, onde foi realizado a carga, limpeza e transformação dos dados. Na etapa de mineração de dados, o algoritmo usado no conjunto de dados já tratado anteriormente foi o Apriori que resulta na geração de regras de associação, que mostram a correlação de elementos em comum dentro de um conjunto de dados. A fase seguinte foi a realização do Pós-processamento que consiste em apresentar de forma gráfica o conhecimento obtido no processo de mineração de dados. Com os resultados encontrados, nota-se que o ensino superior presencial foi a modalidade mais frequentada entre os candidatos que realizaram o Enade em 2017 dos quais candidatos solteiros e com renda familiar entre 3 a 4,5 salários mínimos possuem forte tendência a cursar esta categoria.

Palavras-chaves: KDD. Algoritmo Apriori. Enade 2017. Regras de Associação. Painel.

Lista de ilustrações

Lista de tabelas

Sumário

1 Introdução

1.1 Considerações Gerais

Novas aplicabilidades voltadas a tecnologia vêm surgindo e, com o acesso à internet cada vez mais fácil, uma grande quantidade de dados é gerada, desde quando usamos um aplicativo, preenchemos formulários online, efetuamos transações bancárias e até mesmo quando acessamos a internet em algum lugar. Assim como o surgimento emergente de tecnologias da informação e de comunicação, o crescimento exacerbado de dados gerados na internet vem sendo tema de estudos nos últimos anos, já que o armazenamento dessas grandes quantidades de dados proporciona estudos sobre como retirar informações relevantes, como modelos, padrões, relações, confiabilidade, entre outros, o que se enquadra no conceito de Big Data.

O termo Big Data, na realidade, se refere a um conjunto de informações muito amplo que, exatamente por isso, carece de meios para lidar com seu tamanho, de maneira que qualquer dado possa ser interpretado e analisado em tempo hábil [1]. Uma área onde o estudo do levantamento de dados é muito importante é a da educação, pois a fim de encontrar pontos críticos e sempre buscar melhorias é necessário ter métricas que avaliem a qualidade de ensino de determinada instituição em uma região específica ou como um todo.

No Brasil, o assunto educação sempre é pauta de discussões referentes à como realizar melhorias no sistema de ensino. Relativo ao ensino superior tem-se o ENADE, que é o exame nacional de desempenho dos estudantes.

Este exame faz parte do SINAES – Sistema Nacional de Avaliação da Educação Superior – e seu objetivo é avaliar a qualidade dos cursos de formação superior e o rendimento de seus alunos em relação aos conteúdos programáticos, suas habilidades e competências. Este é aplicado a uma amostra selecionada de estudantes do primeiro e do último ano dos cursos [2]. Sua primeira aplicação foi no ano de 2004 e desde então se tornou obrigatório para os estudantes concluintes.

A utilização de técnicas de mineração de dados educacionais permite a análise das informações obtidas através de exames como o ENADE a fim de fomentar estratégias para melhorias no sistema educacional.

1.2 Descrição do Problema

Dados são abertos quando qualquer pessoa pode livremente acessá-los, utilizá-los, modificá-los e compartilhá-los para quaisquer finalidades, estando sujeito a, no máximo, a exigências que visem preservar sua proveniência e sua abertura [3].

Neste trabalho, os estudos serão realizados sobre a base de dados livres do ENADE, que está disponível de forma pública [4]. Com intuito de obter informações através de padrões e/ou associações segundo as características socioeconômicas dos candidatos que realizaram o exame e tiveram suas provas validadas, utilizando técnicas de mineração de dados.

1.3 Objetivo Geral

Desenvolver um código em linguagem R para fazer o pré-processamento e a mineração de dados aplicando o algoritmo Apriori, utilizar o resultado da mineração para mostrar graficamente no serviço de análise de negócios, que fornece visualizações interativas e recursos de business intelligence, o Power BI.

1.4 Objetivo Específico

A seguir, estão descritos os objetivos específicos desse trabalho:

- Realizar a limpeza da base fazendo a seleção das variáveis necessárias para o estudo;
- Utilizar o pacote Arules para aplicar o algoritmo Apriori na base de dados já tratada a fim de gerar regras de associações para encontrar relacionamentos ou padrões frequentes para o conjunto de dados;
- Realizar uma poda nas regras geradas advindas da aplicação do Apriori para eliminar as regras ditas não interessantes e/ou redundantes para o estudo dos dados;
- O resultado da poda, será mostrado com as regras geradas, de forma visual através de um painel no Power BI, assim, apresentando o conhecimento obtido pelo algoritmo de mineração.

1.5 Organização do Trabalho

O trabalho está organizado em cinco seções. Na Seção 1, é apresentada a introdução ao tema do trabalho. Na Seção 2, é feito um levantamento bibliográfico sobre as etapas do KDD, sobre mineração de dados educacionais e apresenta alguns trabalhos relacionados

ao tema. A Seção 3 descreve a metodologia aplicada ao trabalho. Na Seção 4, são exibidos os resultados obtidos nas etapas do KDD utilizando o R e o Power BI. E na Seção 5, é descrita a conclusão e trabalhos futuros.

2 Levantamento Bibliográfico

2.1 Condiderações Iniciais

Com a internet possibilitando que vários serviços migrem da forma física para a digital, e com o aumento da utilização desses serviços gerando um volume absurdo de dados, tornou-se possível que estes sejam armazenados como registros de bancos de dados contendo informações diversas sobre os usuários dos serviços oferecidos.

O problema é que estes registros muitas vezes representam apenas dados e não conhecimento. Visando transformar estes dados em conhecimento, surge o processo chamado de Descoberta de Conhecimento em Bancos de Dados (Knowledge Discovery in Databases - KDD) [5].

O processo de KDD é constituído de várias etapas que por sua vez formam um ciclo para o processamento de dados, para que o dado passe de apenas uma informação armazenada para se tornar de fato conhecimento sobre aquilo que eles representam.

A Figura 1 mostra de forma ilustrativa como são divididas as etapas do processo de descoberta de conhecimento em banco de dados – KDD.

As cinco etapas do KDD são:

- Seleção: Nesta fase é escolhido o conjunto de dados, pertencente a um domínio, contendo todas as possíveis variáveis e registros que farão parte da análise [5].

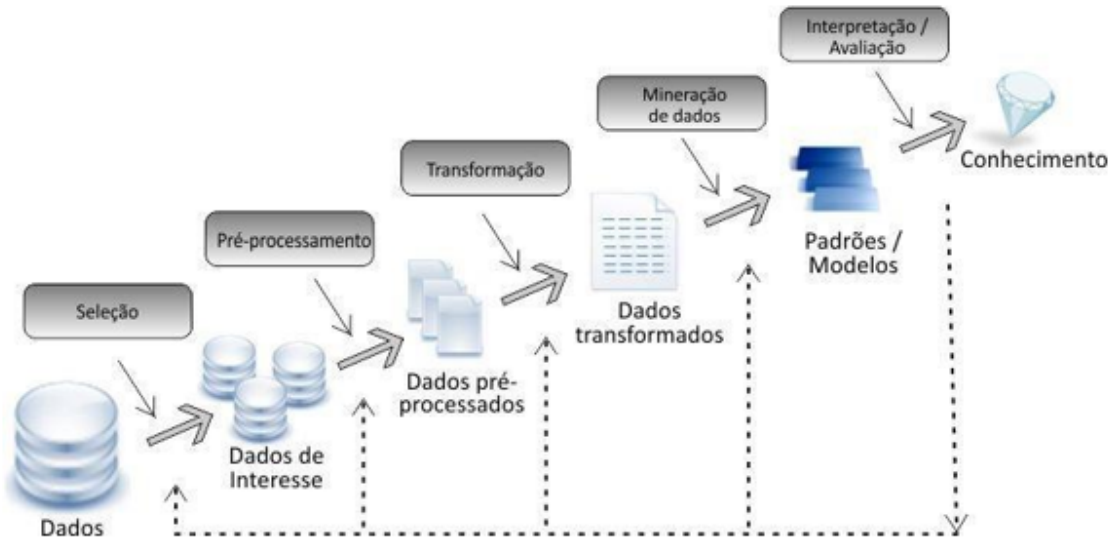

Figura 1 – Etapas do processo de KDD. Fonte: Adaptado de [6].

- Pré-processamento: Etapa onde deverão ser realizadas tarefas que eliminem dados redundantes e inconsistentes, recuperem dados incompletos e avaliem possíveis dados discrepantes ao conjunto. Nesta fase também são utilizados métodos de redução ou transformação para diminuir o número de variáveis envolvidas no processo, visando com isto melhorar o desempenho do algoritmo de análise [5].

- Transformação: Após serem selecionados, limpos e pré-processados os dados necessitam ser armazenados e formatados adequadamente para que os algoritmos de aprendizado possam ser aplicados [5].

- Mineração de dados: É considerada a parte mais importante, sendo o núcleo do processo. Nela, são definidas as tarefas e técnicas e aplicados os algoritmos escolhidos sobre o modelo obtido. Durante esse procedimento, pode ser preciso acessar dados adicionais e/ou executar outras mudanças nos dados originalmente selecionados [7].

- Interpretação: Consiste em validar o conhecimento extraído da base de dados, identificando e interpretando padrões em conhecimentos utilizáveis [7].

Essas etapas do KDD são agrupadas em três estágios principais: pré-processamento, aplicação do algoritmo para mineração de dados e pós-processamento, que são descritas a seguir:

O **Pré-processamento:** é a fase de seleção e preparação dos dados. Ela é iniciada, a partir da premissa de especificação dos objetivos a serem alcançados no final do processo de extração de conhecimento. Neste momento que são retirados os dados ruidosos (que contenham valores discrepantes do esperado), inconsistentes e incompletos [8].

A **Mineração de dados:** é o processo de busca de conhecimento através de algoritmos inteligentes. Nesta etapa, os dados são transformados em informações que posteriormente, após a análise e interpretação dessas informações, são transformadas em conhecimentos para tomadas de decisões [9]. Dentre as atividades que podem ser implementadas na Mineração de Dados, destacam-se a classificação, clusterização e sumarização [9].

O **Pós-processamento:** é a fase que envolve a interpretação, análise e apresentação do modelo de conhecimento gerado pela fase de Mineração de Dados. Nesta etapa é avaliado a utilidade do conhecimento extraído na etapa anterior, para que possa ser utilizado como um fator para a tomada de decisão de um especialista ou de um sistema especialista [10].

A Figura 2 mostra o agrupamento dessas etapas nos três estágios descritos.

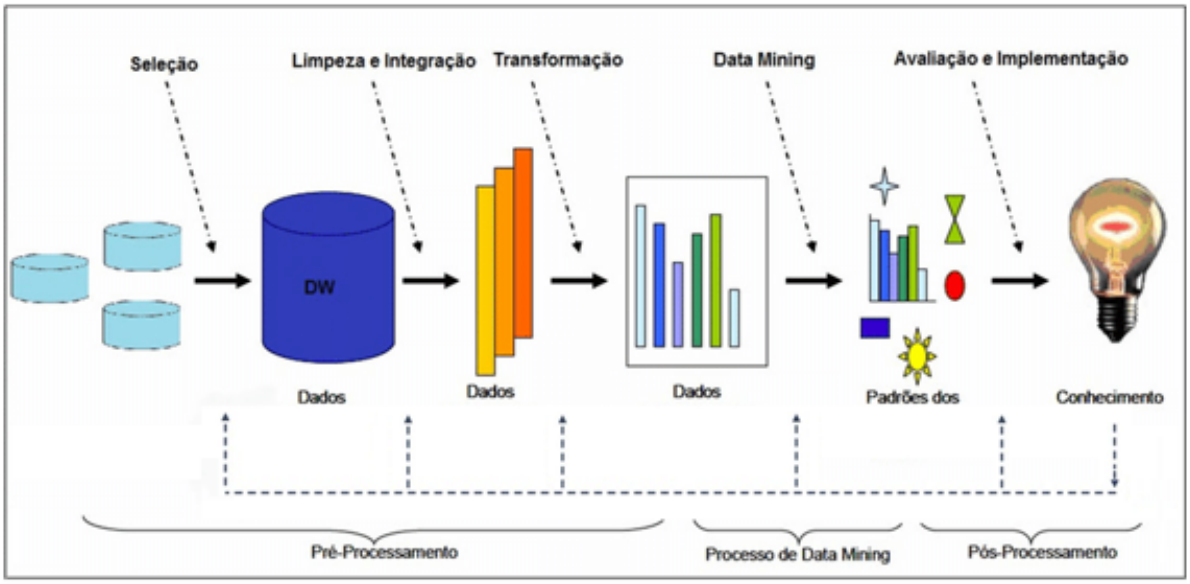

Figura 2 – Divisão das etapas do KDD em três estágios. Fonte: Adaptado de [11]

2.2 Mineração de Dados Educacionais

A mineração de dados tem sido aplicada em diversas áreas do conhecimento, como por exemplo, vendas, bioinformática, e ações contraterrorismo [12]. Assim, pesquisadores da área da educação, viram uma oportunidade de utilizar as técnicas de mineração de dados a fim de buscar respostas para algumas perguntas científicas relacionadas à educação, dando origem a uma nova área de pesquisa conhecida como Mineração de Dados Educacionais - MDE.

A MDE 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 [12]. Com isso, é possível obter diversas informações sobre alunos, instituições, métodos de ensino e personalizar o ambiente educacional para oferecer melhores condições de aprendizagem.

A comunidade de mineração de dados educacionais, desde seus primórdios vem crescendo consideravelmente e de maneira bem rápida, com isso, surgiram diversas conferências internacionais sobre o tema e também workshops bem sucedidos que são realizados anualmente. E com isso, criou-se a Revista de Mineração de Dados Educacionais (Journal of Educational Data Mining).

Existem muitos métodos para mineração de dados e estes são utilizados na MDE, porém, nem todos podem ser aplicados diretamente, antes eles precisam passar por modificações, incluindo algoritmos e ferramentas, principalmente por causa da hierarquia da informação e porque dados educacionais apresentam certa falta de dependência estatística.

As linhas de pesquisas na MDE são diversas e muitas delas derivam diretamente

Tabela 1 – Algoritmos aplicáveis a cada subárea da MDE. Fonte: Adaptado de [12].

Subárea	**Algoritmo**
Predição (prediction)	Redes Neurais Artificiais, C4.5, Classificadores Bayesianos e Algoritmos Genéticos
Agrupamento (clustering)	K-Means, Fuzzy K-Means, K-Modes e KMedoid
Mineração de relações (Relationship Mining)	Apriori, Partition, GSP, MSDD

da mineração de dados, e ainda existem dentre essas linhas de pesquisas, as principais subáreas que a MDE aborda que são: predição, agrupamento, mineração de relações, destilação de dados para facilitar decisões humanas, descobrimento com modelos.

As três primeiras categorias dessa taxonomia são de interesse tanto da área de MDE quanto da área de mineração de dados em geral [12]. A Tabela 1 apresenta a seguir um resumo dos algoritmos aplicados nas três primeiras subáreas que são de interesse tanto para MDE quanto para mineração de dados.

2.3 Trabalhos Relacionados

Os trabalhos apresentados seguir demonstram conceitos, técnicas utilizadas, aplicações e ferramentas para o processo e mineração de dados educacionais.

Em “Mineração de dados e Big Data na educação”, [13], é apresentado uma análise de conceitos e utilidades que perpassam a Mineração de Dados (Data Mining) e Big Data, com o objetivo de demonstrar as consequências das aplicações desses dois pilares tecnológicos da atualidade no campo da educação. Este trabalho mostra que tanto para escolas como para universidades de âmbito público ou privado, a utilização de Sistemas Tutores Inteligentes (STIs), Ambiente Virtual de Aprendizagem (AVA), entre outros, gera um grande volume de dados referentes aos alunos e às instituições. Para que seja possível a extração de informações relevantes que agreguem melhorias para os sistemas de ensino e aprendizagem, é necessário aplicar o processo de mineração de dados educacionais sobre tais dados. Os autores apresentam uma relação das seis melhores ferramentas de mineração de dados de código aberto que podem ser utilizadas, além e demonstrar um estudo feito em 2016, do pesquisador Amjad Abu Saa, do departamento de tecnologia da informação da universidade de ciência e tecnologia de Ajman, nos Estados Unidos, para descobrir as relações entre fatores pessoais e sociais dos alunos, e seu desempenho no semestre anterior utilizando técnicas de mineração de dados como a classificação [13].

Em “Mineração de Dados Educacionais: Conceitos, Técnicas, ferramentas e aplicações”, [14], também é explicada a importância de se analisar dados provenientes do

sistema educacional, e neste são apresentados tarefas e algoritmos de MDE. Uma tarefa apresentada é a predição, cujo objetivo é desenvolver modelos que façam inferência sobre aspectos específicos dos dados (variáveis preditivas) por meio da análise e associação dos diversos aspectos encontrados nos dados (variáveis preditoras). Em MDE, são utilizados mais frequentemente dois tipos de técnicas de predição: classificação e regressão. Na classificação a variável preditiva é binária ou categórica enquanto que na regressão a variável preditiva é contínua. Em ambos os casos, as variáveis preditoras podem ser categóricas ou contínuas. No trabalho em questão, também são apresentados os seguintes métodos de mineração de dados: árvore de decisão, máquinas de vetores de suporte, regressão linear, agrupamento, algoritmo K-Means, algoritmo genético, mineração de relações, regras de associação, destilação de dados para facilitar decisões humanas e descoberta com modelos. Para cada método é apresentado uma explicação de seu funcionamento. Também há uma abordagem de como preparar os dados para que seja possível a aplicação dessas técnicas e algoritmos.

Em "Mineração de dados educacionais guiado por mapa de conhecimento", [15], a autora apresenta o desenvolvimento de mapas de conhecimento criados com a colaboração de gestores da Universidade Estadual do Rio de Janeiro, gerando um estudo de mitigação do índice de candidatos desistentes das vagas oferecidas pelo Vestibular da Instituição, otimizando a ocupação de vagas ofertadas. A base de dados usada foi referente ao ano de 2010. O trabalho propõe uma camada de interação com os usuários do modelo de Gestão de Conhecimento, com a implementação de um portal onde são apresentados indicadores obtidos através de mapas de conhecimento, implementados nas etapas da metodologia `Domain-Drive Data Mining`. E para a etapa de mineração de dados foi utilizada a ferramenta WEKA.

3 Metodologia

3.1 Ambiente de Dados

O Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP) é uma autarquia federal vinculada ao Ministério da Educação (MEC). Sua missão é subsidiar a formulação de políticas educacionais dos diferentes níveis de governo com intuito de contribuir para o desenvolvimento econômico e social do país [16].

O INEP atua sobre as seguintes áreas: Avaliações, exames e indicadores da educação básica. E uma de suas finalidades é organizar e manter o sistema de informações e estatísticas educacionais, para isso, são gerados pelo Inep, dados referentes ao desempenho de estudantes e magistrados.

Os dados utilizados neste trabalho foram extraídos do sítio eletrônico do INEP (http://portal.inep.gov.br/web/guest/início) onde foi acessado o ícone dado e posteriormente microdados. A base selecionada foi microdados ENADE 2017.

Os dados se encontram em formato txt, no pacote microdados_Enade_2017, que contém também a documentação da base e arquivos de entrada para as ferramentas R, SAS e SPSS.

Segundo o manual do usuário [17], presente no pacote microdados_Enade_2017, os dados coletados são advindos do exame realizado no dia 26 de novembro quando foram avaliados os estudantes dos cursos que conferem diploma de bacharel em: Arquitetura e Urbanismo, Engenharia Ambiental, Engenharia Civil, Engenharia de Alimentos, Engenharia de Computação, Engenharia de Controle e Automação, Engenharia de Produção, Engenharia Elétrica, Engenharia Florestal, Engenharia Mecânica, Engenharia Química, Engenharia e Sistema de Informação; dos cursos que conferem diploma de bacharel ou licenciatura em: Ciência da Computação, Ciências Biológicas, Ciências Sociais, Filosofia, Física, Geografia, História, Letras – Português, Matemática e Química, dos cursos que conferem diploma de licenciatura em: Artes Visuais, Educação Física, Letras - Português e Espanhol, Letras - Português e Inglês, Letras – Inglês, Música e Pedagogia; e dos cursos que conferem diploma de tecnólogo em: Análise e Desenvolvimento de Sistemas, Gestão da Produção Industrial, Gestão da Tecnologia da Informação e Redes de Computadores.

3.2 Etapas do KDD

Este capítulo contém a apresentação da metodologia utilizada no processo de execução deste trabalho. Englobando os processos aplicados a massa de dados na fase de

Pré-processamento, sendo eles: seleção, limpeza, transformação, na fase de Mineração dos Dados e no Pós-processamento.

3.2.1 Pré-Processamento

A base de dados original, inicialmente tratada possuía 537.436 registros e 150 variáveis. Para a fase de seleção, foi necessário fazer uma varredura nestes dados para que permanecessem na base, apenas as variáveis que seriam utilizadas para estudo, neste caso, aquelas que possuíam informações dos candidatos, instituição de ensino e curso, restando assim 19 variáveis. A Tabela 2 apresenta o conjunto que permaneceu na base depois da etapa de seleção.

Logo após a seleção, foi realizada a limpeza da base, onde foram eliminados todos os registros que possuíam a informação de candidato ausente e candidatos com provas inválidas. Para isto, foram utilizadas as variáveis que continham os registros de presença no Enade e das provas de cada candidato que realizou o exame no referido ano sendo elas TP_PRES e TP_PR_GER, respectivamente. Após esta triagem, estes dois atributos, também foram retiradas da base.

Ainda nesta etapa, foram removidos todos os registros que tinham campos com dados nulos e/ou vazios para não afetar a qualidade dos modelos de conhecimento que serão extraídos no final do processo de KDD. Ao final desta etapa, a base continha 441.939 registros e 17 variáveis. Na Listagem 3.1, é apresentado o código utilizado para a seleção e limpeza dos dados feito em linguagem R.

Na etapa de transformação dos dados, foram realizados dois procedimentos: no primeiro realizou-se a classificação das variáveis com informações de notas dos candidatos, estas passaram a ser rotuladas como Bom, Médio e Ruim, dependendo da pontuação ao qual ela representava. Classificadas como Bom, notas acima de 50 pontos, Médio, notas entre 30 e 50 pontos e ruim, abaixo de 30 pontos. O segundo procedimento de transformação dos dados foi o de converter as variáveis do tipo numéricas para caractere que posteriormente serão utilizadas como entrada para o algoritmo de mineração de dados, Apriori. A Listagem 3.2 apresenta o código em R da Transformação de dados.

3.2.2 Mineração de Dados

Para esta etapa do trabalho, o algoritmo utilizado foi o Apriori que, em R, encontra-se no pacote Arules, disponível para todas as versões.

APRIORI

O algoritmo Apriori é um dos mais utilizados para mineração de regras de associação. Segundo [18], regras de associação tem formato A => B, na qual A é chamado de

Tabela 2 – Variáveis selecionadas. Fonte: Próprio autor.

Variável	**Tipo**	**Descrição**
CO_CATEGAD	Numérica	Código da categoria administrativa da IES
CO_GRUPO	Numérica	Código da área de enquadramento do curso no Enade
CO_MODALIDADE	Numérica	Código da modalidade de Ensino
CO_UF_CURSO	Numérica	Código da UF de funcionamento do curso
CO_REGIAO_CURSO	Numérica	Código da região de funcionamento do curso
NU_IDADE	Numérica	Idade do inscrito em 26/11/2017
TP_SEXO	Caractere	Tipo de sexo
ANO_FIM_EM	Numérica	Ano de conclusão do Ensino Médio
ANO_IN_GRAD	Numérica	Ano de início da graduação
CO_TURNO_GRADUACAO	Numérica	Código do turno de graduação
NT_GER	Numérica	Nota bruta da prova - Média ponderada da formação geral (25%) e componente específico (75%). (valor de 0 a 100)
NT_FG	Numérica	Nota bruta na formação geral - Média ponderada da parte objetiva (60%) e discursiva (40%) na formação geral.(valor de 0 a 100)
NT_OBJ_FG	Numérica	Nota bruta na parte objetiva da formação geral. (valor de 0 a 100)
NT_DIS_FG	Numérica	Nota bruta na parte discursiva da formação geral. (valor de 0 a 100)
QE_I01	Caractere	Qual o seu estado civil?
QE_I02	Caractere	Qual é a sua cor ou raça?
QE_I08	Caractere	Qual a renda total de sua família, incluindo seus rendimentos?
TP_PRES	Numérica	Tipo de presença no Enade
TP_PR_GER	Numérica	Tipo de presença na prova

Listagem 3.1 – Código em R usado para realizar a seleção das variáveis e a limpeza da base de dados. Fonte: Próprio autor.

```
1
2
3 #SELECIONANDO AS VARIVEIS NECESSARIAS
4
5 Enade2017_Selecao <- DadosEnade2017 %>% select(-(1:2),-4, -6, -8, -(16:33),
      -(36:44), -(49:69), -(72:76), -(78:150))
6 view(Enade2017_Selecao)
7
8 #LIMPEZA DA BASE
9
10 # ---------------- eliminando candidatos ausentes:
11 Enade2017_Limpo <- Enade2017_Selecao %>% filter(TP_PRES != 222 & TP_PR_GER !=
      222)
12 #view(Enade2017_Limpo)
13
14 # ---------------- mantendo apenas candidatos presentes com provas  v lidas :
15 Enade2017_Limpo2 <- Enade2017_Limpo %>% filter(TP_PRES == 555 & TP_PR_GER == 555)
16 #view(Enade2017_Limpo2)
17
18 # ---------------- eliminhando as colunas de presenca:
19 Enade2017_Oficial <- Enade2017_Limpo2 %>% select(-11, -12)
20 #view(Enade2017_Oficial)
21
22 # ---------------- eliminhando campos com NA:
23 Enade2017_Oficial <- Enade2017_Oficial[!(Enade2017_Oficial$QE_I02 == ""), ]
24
25 ApagarLinhas <- c()
26
27 for (i in 1: length(Enade2017_Oficial[,1])) {
28   for (j in 1: length(Enade2017_Oficial[1,])){
29     if((Enade2017_Oficial[i,j] == "" | is.na(Enade2017_Oficial[i,j]))){
30       ApagarLinhas <- c(ApagarLinhas,i)
31       break
32     }
33   }
34 }
35
36 ApagarLinhas
37 length(ApagarLinhas)
38 ApagarLinhas2 <- sort(ApagarLinhas, decreasing = TRUE)
39 ApagarLinhas2
40
41 Enade2017_Oficial_2 <- Enade2017_Oficial
42
43 for (k in ApagarLinhas2) {
44   #Enade2017_Oficial_2 <- removeRows(i, Enade2017_Oficial)
45   Enade2017_Oficial_2 <- Enade2017_Oficial_2[-k, , drop = FALSE]
46 }
```

Listagem 3.2 – Algoritmo em R da transformação de dados. Fonte: Próprio autor.

```
1
2 # ---------------- classificando as notas Gerais do ENADE:
3 j <- 12
4 for (i in 1: length(BaseApriori_2[,1])) {
5   for (j in (12:15)) {
6     if(BaseApriori_2[i,j] <= 30){
7       BaseApriori_2[i,j] <- "Ruim"
8     } else if (BaseApriori_2[i,j] <= 50){
9       BaseApriori_2[i,j] <- "Mdio"
10    }else{
11      BaseApriori_2[i,j] <- "Bom"
12    }
13  }
14 }
15
16 view(BaseApriori_2)
17 write.csv(BaseApriori_2, "BC_Comnotas.csv")
18
19 # ---------------- transformando variveis numericas em character:
20 BaseApriori_2 <- read.csv("BC_Comnotas.csv")
21 BaseApriori_3 <- BaseApriori_2
22 BaseApriori_3 <- BaseApriori_3 %>% select(-1,-2,-6, -(13:15))
23 #
24 view(BaseApriori_3)
25
26 BaseApriori_3$CO_CATEGAD <- as.character(BaseApriori_3$CO_CATEGAD)
27 BaseApriori_3$CO_GRUPO <- as.character(BaseApriori_3$CO_GRUPO)
28 BaseApriori_3$CO_MODALIDADE <- as.character(BaseApriori_3$CO_MODALIDADE)
29 BaseApriori_3$CO_UF_CURSO <- as.character(BaseApriori_3$CO_UF_CURSO)
30 BaseApriori_3$CO_REGIAO_CURSO <- as.character(BaseApriori_3$CO_REGIAO_CURSO)
31 BaseApriori_3$NU_IDADE <- as.character(BaseApriori_3$NU_IDADE)
32 BaseApriori_3$TP_SEXO <- as.character(BaseApriori_3$TP_SEXO)
33 BaseApriori_3$ANO_FIM_EM <- as.character(BaseApriori_3$ANO_FIM_EM)
34 BaseApriori_3$ANO_IN_GRAD <- as.character(BaseApriori_3$ANO_IN_GRAD)
35 BaseApriori_3$CO_TURNO_GRADUACAO <- as.character(BaseApriori_3$CO_TURNO_GRADUACAO
       )
```

antecedente, e B de consequente. A e B são conjuntos de itens ou transações. Uma regra pode ser lida como: o atributo A frequentemente implica no atributo B.

De forma geral, o algoritmo recebe como parâmetro de entrada um conjunto de transações da base de dados e dois valores percentuais que servem como quesito avaliativo para geração das regras, o suporte e a confiança. Gera como saída, um conjunto de regras que possuem formato A => B, sendo A o conjunto conhecido como antecedente e B como consequente. As regras geradas possuem suporte e confiança maior ou igual aos passados como parâmetro no algoritmo.

De [19] temos que, suporte é o percentual de vezes que um dado conjunto A aparece no conjunto de transações. Valores de suporte muito baixos indicam regras pouco relevantes que, portanto, podem ser descartadas ou preteridas. Confiança é a significância estatística do suporte, dado uma regra A => B, representa dentre as transações que possuem os itens de A, a porcentagem de transações que possuem também os itens de B, ou seja, indica o percentual de ocorrência da regra.

O objetivo principal do Apriori é gerar regras de associação se baseando na quanti-

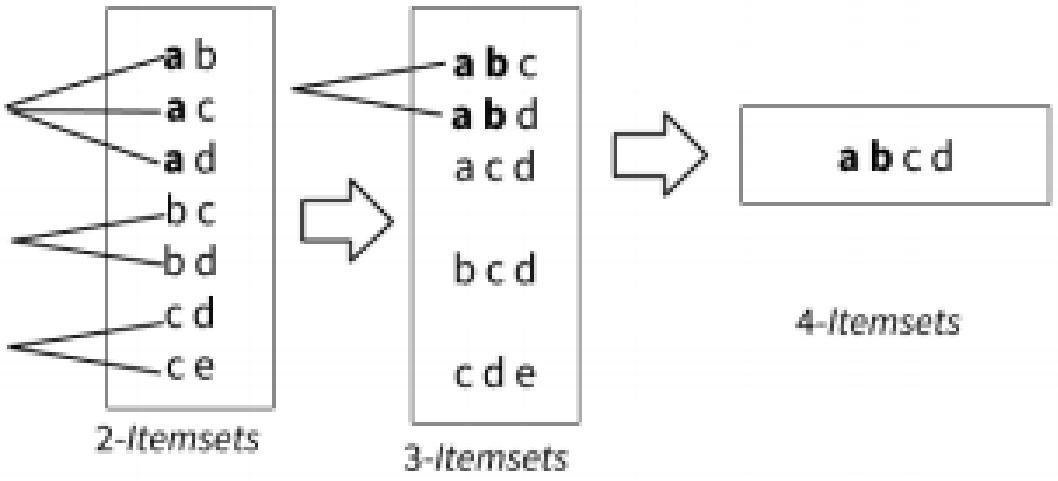

Figura 3 – Exemplo de geração de itemsets. Fonte: Adaptado de [22].

dade de vezes que itens ocorrem juntos no conjunto de transação. Realizando inúmeras combinações entre eles.

No algoritmo, item é definido como uma variável da base e um *itemset* é um conjunto de itens. O Apriori trabalha sobre o princípio de que se um *itemset* é frequente, então, todos os seus subconjuntos devem ser frequentes. De acordo com [20], a primeira etapa do algoritmo realiza o cálculo do suporte de todas as combinações de itens e faz a contagem de ocorrências dos mesmos, e assim determinar os *itemsets* frequentes de tamanho unitário. Estes, chamados de 1-*itensets* frequentes, serão usados na geração dos próximos conjuntos de itens de tamanho k (k-itemsets), onde k>1.

Para gerar os *itemsets* de tamanho k, segundo [21], é feito um processo de junção entre itens de conjuntos de itemsets de tamanho k-1 pelo critério de prefixos iguais. Assim, as combinações só serão efetivas entre os itens que possuem o mesmo prefixo. A Figura 3 apresenta um exemplo de geração de Itemsets.

Segundo [22], *"Cada novo conjunto de itens gerados é denominado conjunto de itens candidatos e o suporte é calculado consultando novamente a base de dados de transações. Em seguida os itemsets que não atingirem o valor do suporte mínimo são podados do conjunto candidatos, gerando um novo conjunto de itens denominado conjunto de itens frequentes. O processo continua até que nenhum novo conjunto de itens candidatos seja criado"*. A seguir, é apresentado um exemplo ilustrativo de como ocorre o processo de seleção dos itemsets frequentes. Para este, foi usado um valor inteiro de suporte igual a 2, ou seja, o número de exemplos cobertos pelo *itemset*. A Figura 4 apresenta uma Ilustração do processo de seleção de *itemssets* frequentes.

A ultima etapa realizada pelo algoritmo é a de geração das regras, onde, para cada conjunto de itens frequentes obtidos nas etapas anteriores, gera-se as regras de acordo com

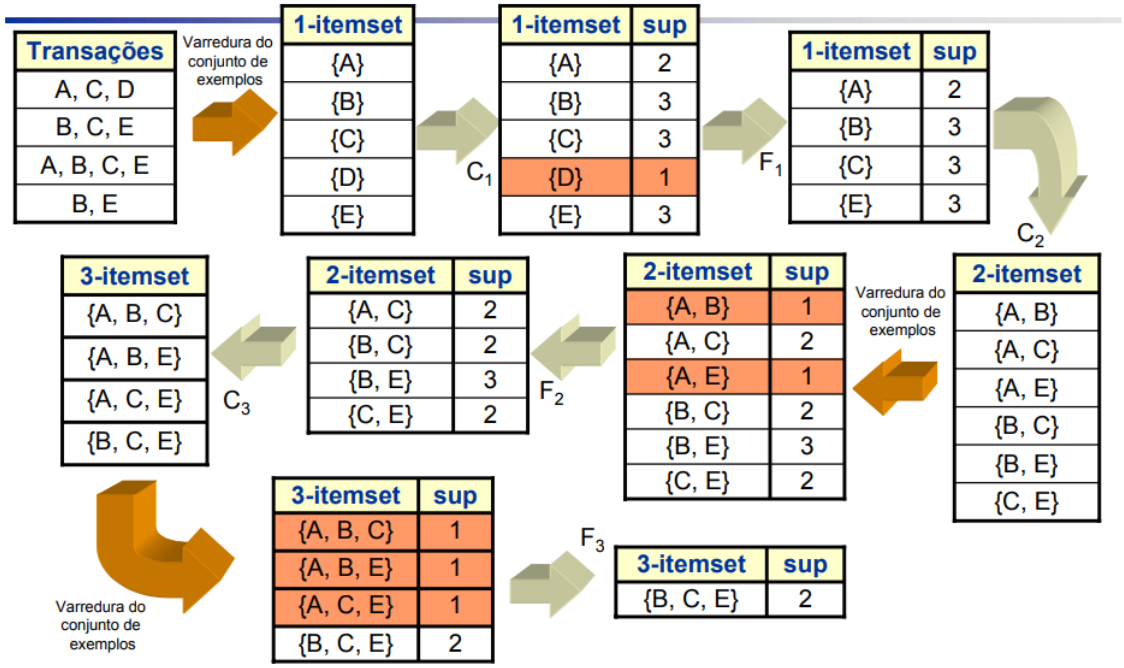

Figura 4 – Ilustração do processo de seleção de itemssets frequentes. Fonte: Adaptado de [23].

o valor de confiança definido como parâmetro de entrada.

A Listagem 3.3 apresenta o código em linguagem R da aplicação do algoritmo Apriori na base de estudos.

Listagem 3.3 – Aplicação do algoritmo Apriori em linguagem R. Fonte: Próprio autor.

```
1
2
3 # ---------------- Aplicando o algoritmo:
4 rules <- apriori(BaseApriori_3, parameter = list(supp = 0.1, conf = 0.9))
5 #inspect(rules)
6
7 BaseRules <- as(rules, "data.frame")
```

Para finalizar a etapa de Mineração de Dados, foi implementado um algoritmo de poda e seleção das melhores regras geradas pelo Apriori, baseado na proposta apresentada em [24]. A Figura 5 apresenta o princípio de funcionamento do algoritmo de poda e seleção das melhores regras, isto é, uma representação do funcionamento do algoritmo ao qual utiliza como princípios a poda por cobertura e eliminação de contradições.

A primeira etapa executada pelo algoritmo é a de poda por cobertura, que é realizada tendo em vista os fatores confiança e suporte na geração dos itens frequentes, pois quando uma regra é gerada, ocorre a verificação da equivalência da confiança de seus consequentes. Assim, quando estes são iguais, ocorre o cruzamento e gera-se uma nova regra que pode estar no mesmo subconjunto de regras de acordo com a confiança definida.

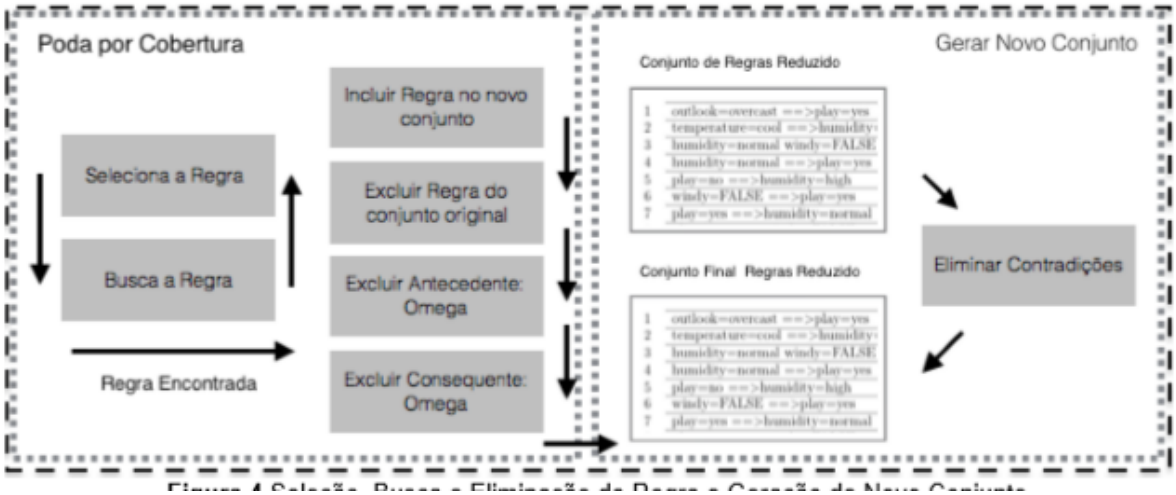

Figura 5 – Princípio de funcionamento do algoritmo de poda e seleção das melhores regras. Fonte: Adaptado de [24].

Portanto, é possível que regras sejam geradas e já estejam cobertas por outras regras e assim, estas podem ser eliminadas.

Para o processo de eliminação de uma regra é necessário verificar os antecedentes e consequentes de cada regra de um conjunto de regras. Dada uma regra do conjunto de dados, chamaremos ela de R1, é realizada uma busca verificando se outras regras possuem itens de antecedentes e consequentes iguais a R1, ou seja, regras que são cobertas por ela. Caso seja encontrada uma regra que é coberta por R1, esta será eliminada do conjunto de dados e R1 será salva em uma base auxiliar. Este processo é repetido até que todas as regras do conjunto sejam testadas.

A segunda etapa executada pelo algoritmo é a de eliminar contradições, esta é regida pelo Paradoxo de Simpson, segundo [25] *"O paradoxo de Simpson é um paradoxo da estatística no qual um conjunto de dados completo aponta em uma direção, mas uma análise de subconjuntos aponta na direção contrária".*

Nesta parte, a base auxiliar criada na etapa anterior é avaliada. Dado uma regra nela presente, será verificado se no restante da base, existem regras iguais a ela, porém, com antecedentes e consequentes invertidos como mostra o exemplo ilustrativo do processo de eliminar contradições, o qual é exibido na Figura 6. Caso exista uma regra igual, porém com antecedentes e consequentes invertidos, está regra invertida será eliminada da base auxiliar. Este teste é realizado em todas as regras da base auxiliar, restando assim, um conjunto com a seleção das melhores regras geradas pelo algoritmo Apriori.

A Listagem 3.4 apresenta o código em linguagem R da poda que foi implementado para este trabalho com as regras geradas pelo Apriori em linguagem R.

Listagem 3.4 – Algoritmo de poda das regras geradas pelo Apriori em linguagem R. Fonte: Próprio autor.

```
1
2 # ---------------- algoritmo de poda:
3 Poda_Regras <- function(arq_regras){
4 #separando a coluna de regras em duas
5   Aux <- arq_regras
6   Entrada_Aux <- Aux %>% separate(rules, into = c('antecedente', 'consequente'),
        sep = ' => ')
7 #Poda por cobertura
8   Saida <- Aux[0,]
9   i <- 1
10    while (i <= length(Entrada_Aux[,1])){
11      elem <- gsub(".*[{]([^.]+)[}].*", "\\1", Entrada_Aux[i,2])
12      elem2 <- gsub(".*[{]([^.]+)[}].*", "\\1", Entrada_Aux[i,3])
13      j <- 1
14      while (j <= length(Entrada_Aux[,1])) {
15        if(is.na(elem2) || elem2 == "NA"){
16          break
17        }
18        if((i != j) && (grepl(elem, Entrada_Aux[j,2]) == TRUE)){
19          if((grepl(elem2, Entrada_Aux[j,3]) == TRUE)){
20              Entrada_Aux <- Entrada_Aux[-c(j),]
21              j <- j - 1
22          }
23        }
24        j <- j + 1
25      }
26       Saida <- rbind(Saida, Entrada_Aux[i,])
27       i <- i + 1
28    }
29 #eliminando os paradoxo de Simpson
30   i <- 1
31    while (i<= length(Saida[,1])) {
32     flag <- 0
33     if(i > length(Saida[,1])){
34       break
35     }
36     j <- 1
37     while (j <= length(Saida[,1])) {
38       if(j > length(Saida[,1])){
39         break
40       }
41       if((all(Saida[i,2] == Saida[j,3])) && (all(Saida[i,3] == Saida[j,2]))){
42         Saida <- Saida[-c(j),]
43         flag <- 1
44         j <- j - 1
45       }
46       j <- j + 1
47     }
48    if (flag == 1){
49      Saida <- Saida[-c(i),]
50    }
51     i <- i + 1
52    }
53   return(Saida)
54 }
```

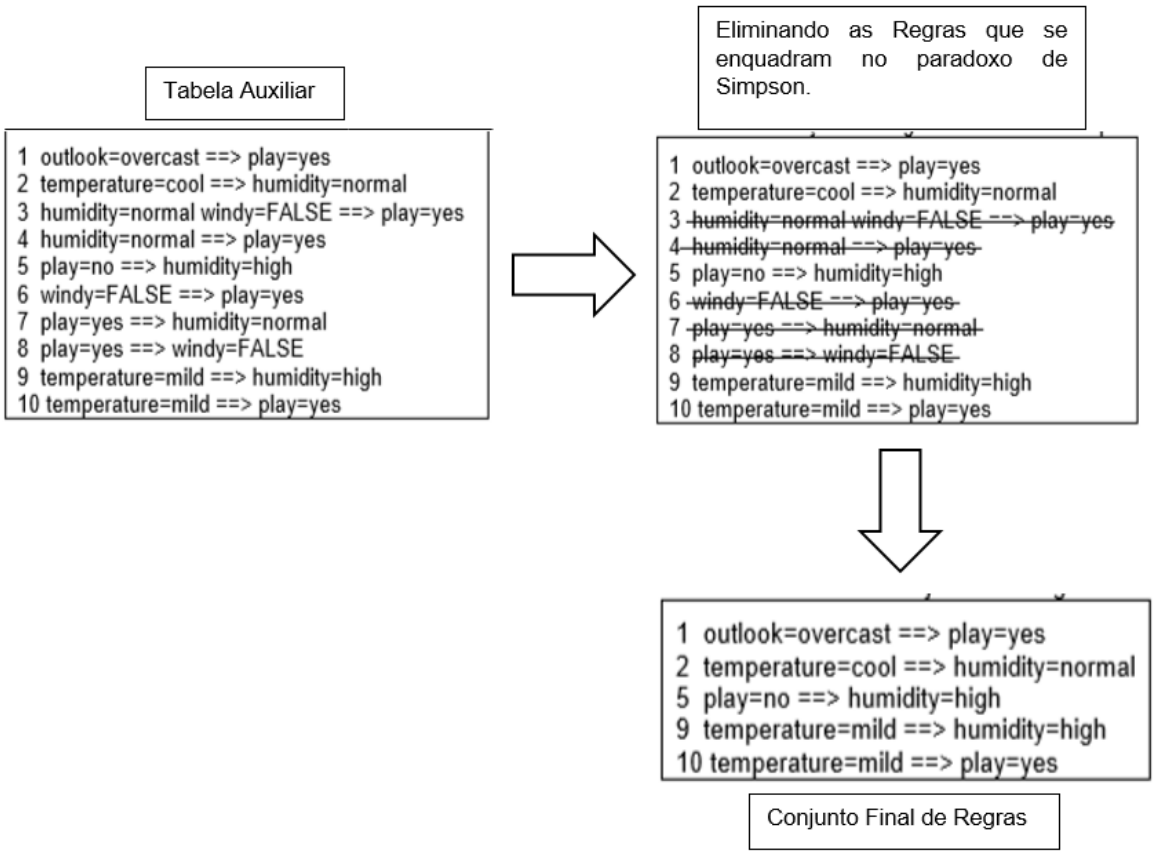

Figura 6 – Exemplo ilustrativo do processo de Eliminar Contradições. Fonte: Adaptado de [24].

3.2.3 Pós-Processamento

O Pós-Processamento, neste trabalho, foi realizado sobre a base de dados resultantes do processo de seleção das melhores regras geradas pelo algoritmo Apriori. A base de dados resultante foi carregada no Power BI Desktop com a finalidade de fornecer visualizações interativas a fim de avaliar o conhecimento extraído na etapa anterior.

Como exemplo de visualização gerada em forma de dashboard foi a de Ranking das regras referentes ao suporte e a confiança das mesmas, como mostra a Figura 7.

Assim como esta, foram geradas outras visualizações diferentes para mostrar o conhecimento obtido de diversas maneiras. E assim, poder identificar o que for relevante para uma possível tomada de decisão.

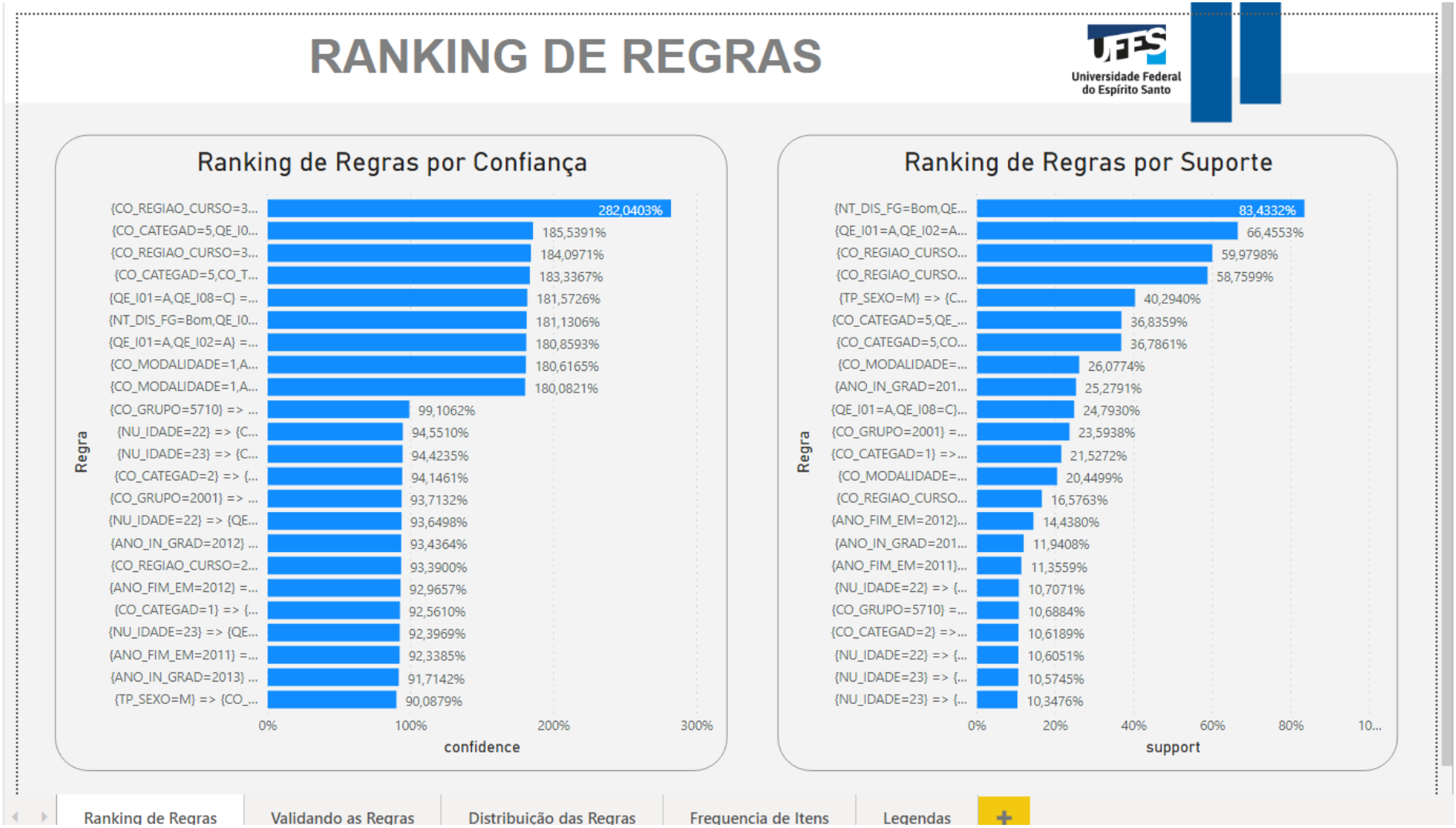

Figura 7 – Painel de visualização de Ranking das regras de acordo com suporte e confiança. Fonte: Próprio autor.

4 Resultados

Todas as etapas deste trabalho foram realizadas sobre a base de estudos completa, assim, as regras geradas são referentes a porção dos dados como um todo. Após a etapa de Pré-processamento (seção 3.2.1), o algoritmo Apriori foi aplicado na base já tratada que continha 441.939 registros e 17 variáveis. Foi usado suporte 0.1 e confiança 0.9, que são valores padrões no algoritmo, e obteve-se como resultado a formação de 93 regras de associação. A Tabela 3 a seguir apresenta uma amostra com as 10 primeiras regras geradas pelo algoritmo.

Como resultado da aplicação do algoritmo de poda e seleção das melhores regras (seção 3.2.2) foram obtidas 23 regras de associação dividindo-as em antecedente e consequente. A Tabela 4 apresenta a saída do algoritmo.

As regras podadas e selecionadas foram aplicadas ao Power BI e os seguintes painéis foram gerados para visualiza-las.

Painel 1: Ranking de Regras

Este painel apresenta a classificação das regras de acordo com seu suporte e com a confiança das mesmas. Nele é apresentado dois gráficos de barras empilhadas um para ranking relativo ao suporte e outro a confiança e assim é possível classificar as regras de acordo com sua posição no ranking. O painel e representado na Figura 8.

Exemplo de funcionamento: É possível clicar em qualquer barra de ambos os gráficos que irá ocorrer a relação da regra de acordo com seu suporte e confiança, assim, é possível avaliar as regras por sua representação na base e o quanto ela é relevante. A Figura 9 mostra como ocorre esta relação.

Painel 2: Validando as Regras

Tabela 3 – Amostra das 10 primeiras regras geradas no algoritmo Apriori. Fonte: Próprio autor.

Regras	**Suporte**	**Confiança**	**Lift**
{CO_GRUPO=5710}=>{CO_MODALIDADE=1}	0,1068	0,9910	1,2399
{NU_IDADE=23}=>{QE_I01=A}	0,1034	0,9239	1,3507
{NU_IDADE=23}=>{CO_MODALIDADE=1}	0,1057	0,9442	1,1813
{CO_CATEGAD=2}=>{CO_MODALIDADE=1}	0,1061	0,9414	1,1779
{NU_IDADE=22}=>{QE_I01=A}	0.1060	0,9364	1,3690
{NU_IDADE=22}=>{CO_MODALIDADE=1}	0,1070	0,9455	1,1829
{ANO_FIM_EM=2011}=>{CO_MODALIDADE=1}	0,1135	0,9233	1,1553
{ANO_IN_GRAD=2012}=>{CO_MODALIDADE=1}	0,1194	0,9343	1,1690
{ANO_FIM_EM=2012}=>{CO_MODALIDADE=1}	0,1443	0.9296	1,1631
{CO_REGIAO_CURSO=2}=>{CO_MODALIDADE=1}	0,1657	0,9339	1,1684

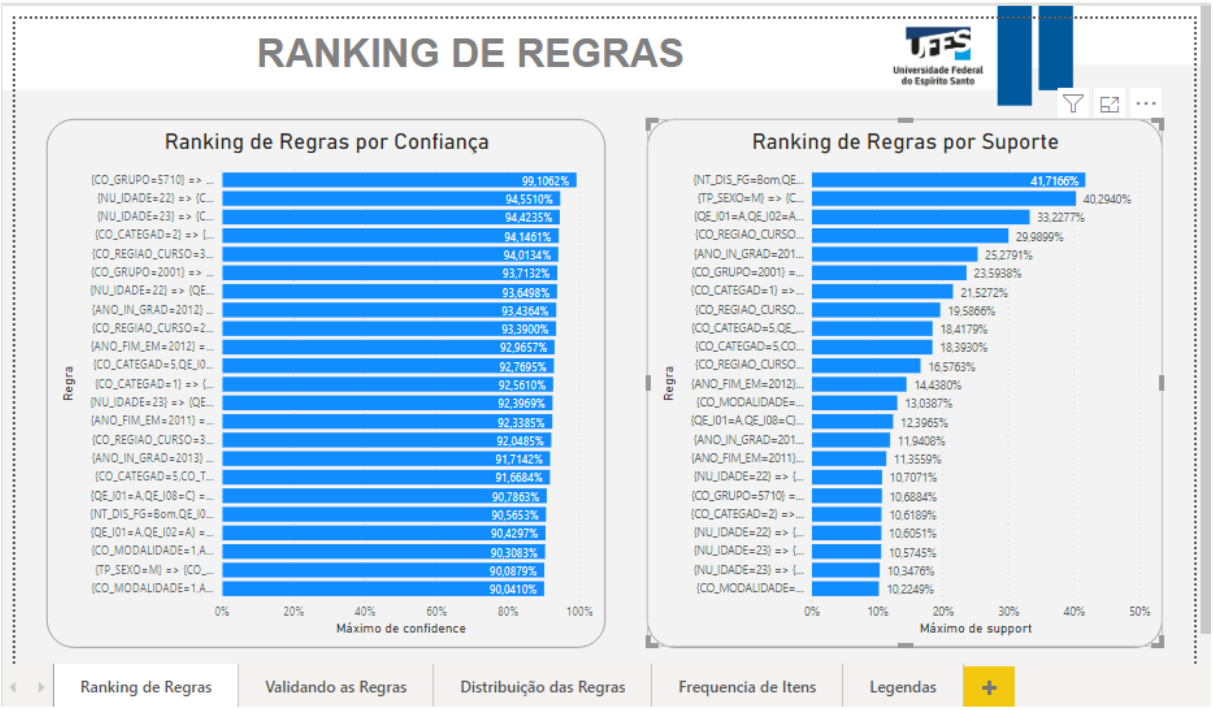

Figura 8 – Painel Ranking das Regras. Fonte: Próprio autor.

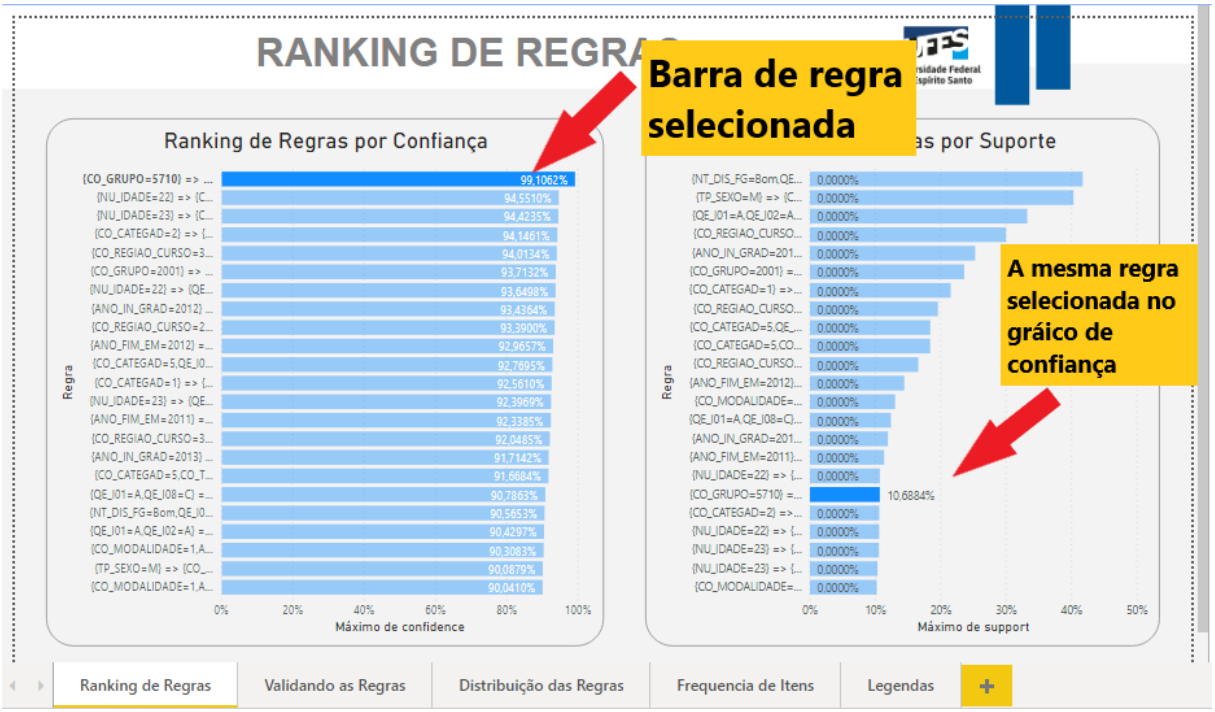

Figura 9 – Exemplo de funcionamento do painel de Ranking de Regras. Fonte: Próprio autor.

Tabela 4 – Regras resultantes do algoritmo de poda e seleção das melhores regras. Fonte: Próprio autor.

Antecedente	**Consequente**	**Suporte**	**Confiança**	**Lift**
{CO_GRUPO=5710}	{CO_MODALIDADE}=1}	0,1069	0,9911	1,2400
{NU_IDADE=23}	{QE_I01=A}	0,1035	0,9240	1,3507
{NU_IDADE=23}	{CO_MODALIDADE=1}	0,1057	0,9442	1,1814
{CO_CATEGAD=2}	{CO_MODALIDADE=1}	0,1062	0,9415	1,1779
{NU_IDADE=22}	{QE_I01=A}	0,1061	0,9365	1,3690
{NU_IDADE=22}	{CO_MODALIDADE=1}	0,1071	0,9455	1,1830
{ANO_FIM_EM=2011}	{CO_MODALIDADE=1}	0,1136	0,9234	1,1553
{ANO_IN_GRAD=2012}	{CO_MODALIDADE=1}	0,1194	0,9344	1,1690
{ANO_FIM_EM=2012}	{CO_MODALIDADE=1}	0,1444	0,9297	1,1632
{CO_REGIAO_CURSO=2}	{CO_MODALIDADE=1}	0,1658	0,9339	1,1685
{CO_CATEGAD=1}	{CO_MODALIDADE=1}	0,2153	0,9256	1,1581
{CO_GRUPO=2001}	{TP_SEXO=F}	0,2359	0,9371	1,6955
{ANO_IN_GRAD=2013}	{CO_MODALIDADE=1}	0,2528	0,9171	1,1475
{TP_SEXO=M}	{CO_MODALIDADE=1}	0,4029	0,9009	1,1272
{CO_MODALIDADE=1, ANO_FIM_EM=2011}	{QE_I01=A}	0,1022	0,9004	1,3163
{CO_MODALIDADE=1, ANO_FIM_EM=2012}	{QE_I01=A}	0,1304	0,9031	1,3202
{QE_I01=A, QE_I08=C}	{CO_MODALIDADE=1}	0,1240	0,9079	1,1359
{CO_CATEGAD=5, CO_TURNO_GRADUACAO=4}	{CO_MODALIDADE=1}	0,1839	0,9167	1,1469
{CO_CATEGAD=5, QE_I01=A}	{CO_MODALIDADE=1}	0,1842	0,9277	1,1607
{CO_REGIAO_CURSO=3, QE_I01=A}	{CO_MODALIDADE=1}	0,2999	0,9205	1,1517
{QE_I01=A, QE_I02=A}	{CO_MODALIDADE=1}	0,3323	0,9043	1,1314
{NT_DIS_FG=Bom, QE_I01=A}	{CO_MODALIDADE=1}	0,4172	0,9057	1,1331
{CO_REGIAO_CURSO=3, CO_TURNO_GRADUACAO=4, QE_I01=A}	{CO_MODALIDADE=1}	0,1959	0,9401	1,1763

Este painel é baseado na premissa do *Lift*, mostrando que se o mesmo possui valor maior que 1, significa que a ocorrência do antecedente provavelmente leva a ocorrência do consequente, ou seja, antecedentes e consequentes estão positivamente correlacionados de acordo com [26]. Nele é mostrado uma tabela com as regras e é possível selecionar qualquer uma delas e então é mostrado o valor do *Lift* da mesma. A Figura 10 apresenta o visual do painel.

Exemplo de funcionamento: Neste painel é possível que selecione qualquer umas das regras presentes na tabela de regras e então é mostrado no cartão de *Lift* da Regra, o valor do *Lift* referente a esta regra, além de mostrar no cartão de Regra, mostra a regra selecionada. A Figura 11 exemplifica o funcionamento do painel.

Painel 3: Distribuição das Regras

Figura 10 – Painel Validando as Regras. Fonte: Próprio autor.

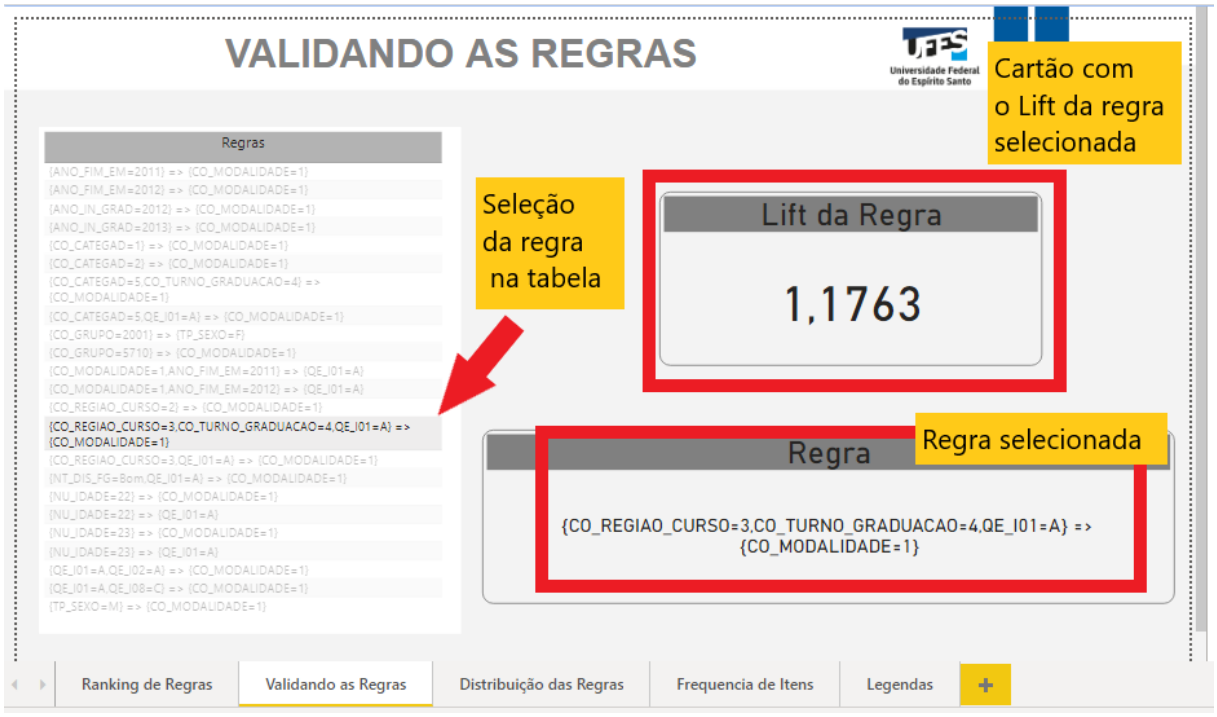

Figura 11 – Exemplo de funcionamento do painel Validando Regras. Fonte: Próprio autor.

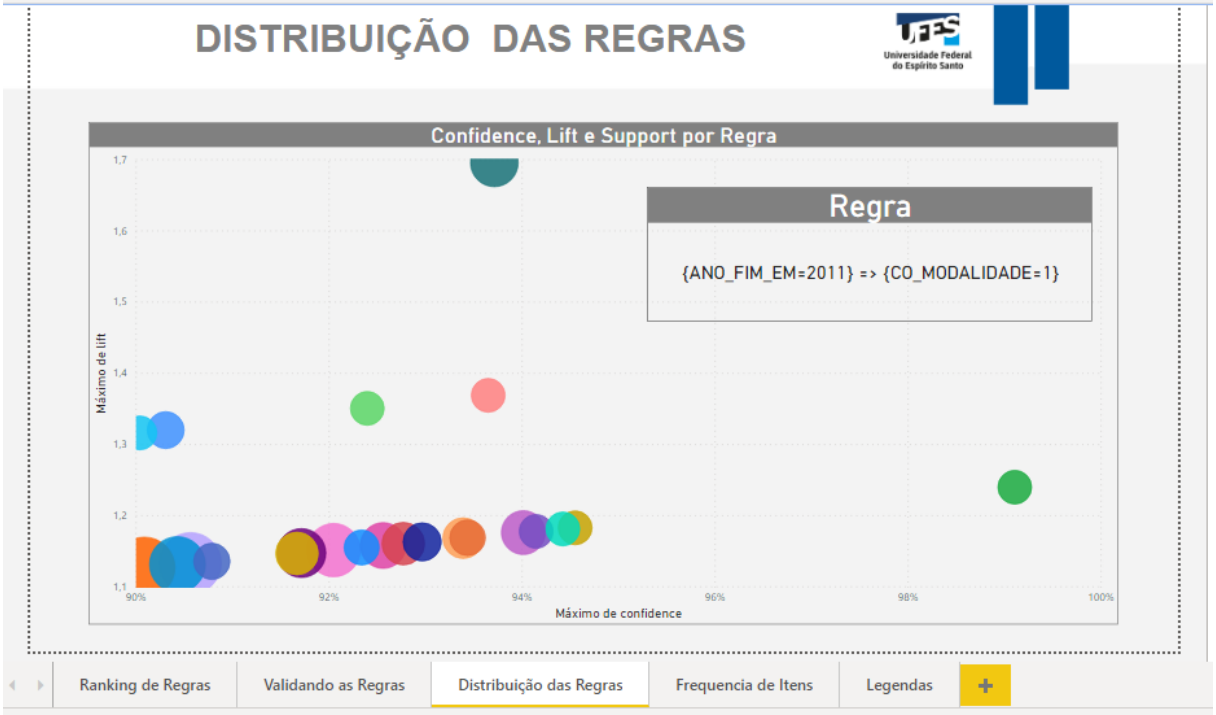

Figura 12 – Painel Distribuição das Regras. Fonte: Próprio autor.

Para este painel, foi usado um gráfico de dispersão onde o eixo Y representa o *Lift*, o eixo X representa a confiança, os pontos representam as regras e o tamanho de cada ponto é de acordo com o suporte de cada regra. Além disso, existe um cartão que informa qual é a regra selecionada. Nele é possível notar o destaque das regras quanto a confiança, suporte e *Lift*, que são as variáveis de saída do algoritmo Apriori. O visual do painel é apresentado na Figura 12.

Exemplo de funcionamento: Ao clicar em qualquer um dos pontos que representam as regras, no cartão de Regras aparecerá qual foi a regra selecionada, além de aparecer uma dica de ferramenta com todas as informações referentes a esta regra. O funcionamento do painel é mostrado na Figura 13.

Painel 4: Frequência de Itens

Possui duas tabelas, uma com a listagem de todos antecedentes das regras geradas na coluna antecedente e na coluna %GT (contagem de antecedente) a listagem dos valores em percentual de quanto aquele antecedente está representado na base utilizada para gerar as regras. A outra tabela possui a listagem de todos os consequentes gerados na coluna consequente e na coluna %GT (contagem de consequente), possui a listagem dos valores em percentual do consequente em relação a base usada para gerar as regras.

O painel também possui um gráfico *Treemap*, onde são listadas todas as 17 variáveis advindas da base resultante da etapa de Pré-processamento, e representa cada item das variáveis em proporção de quantidade referente á base. A Figura 14 mostra o visual do painel.

Figura 13 – Exemplo de funcionamento do painel Distribuição das Regra. Fonte: Próprio autor.

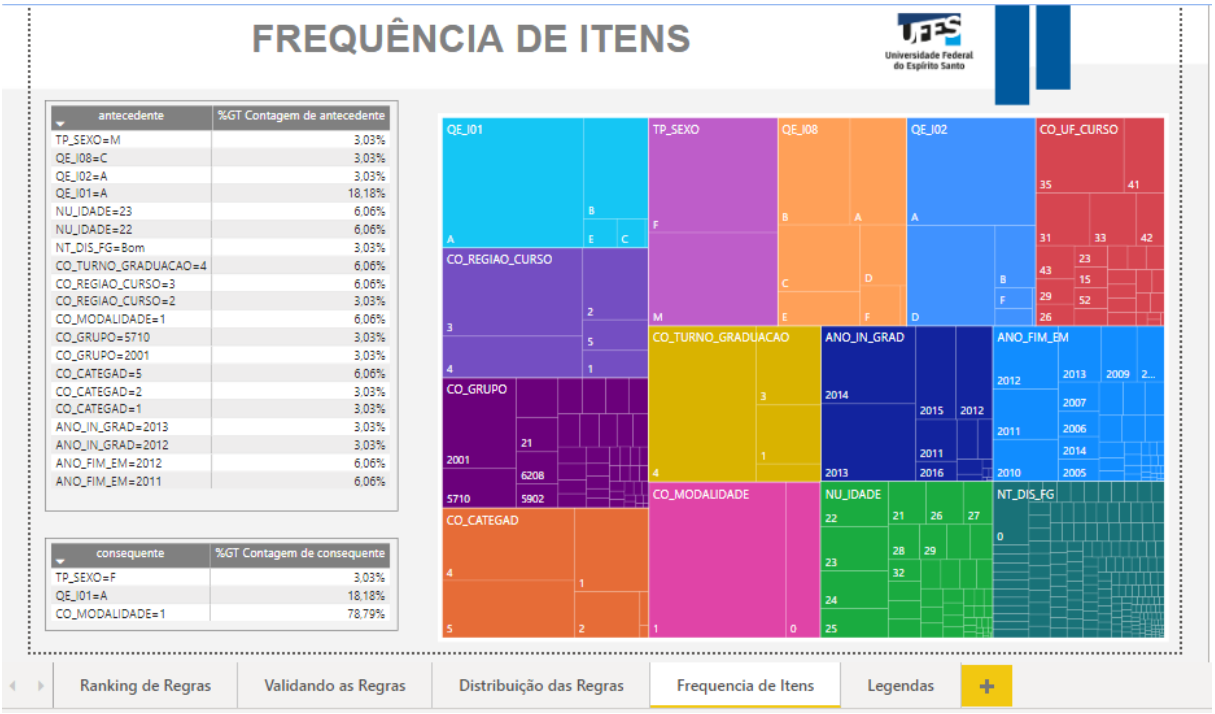

Figura 14 – Painel Frequência de Itens. Fonte: Próprio autor.

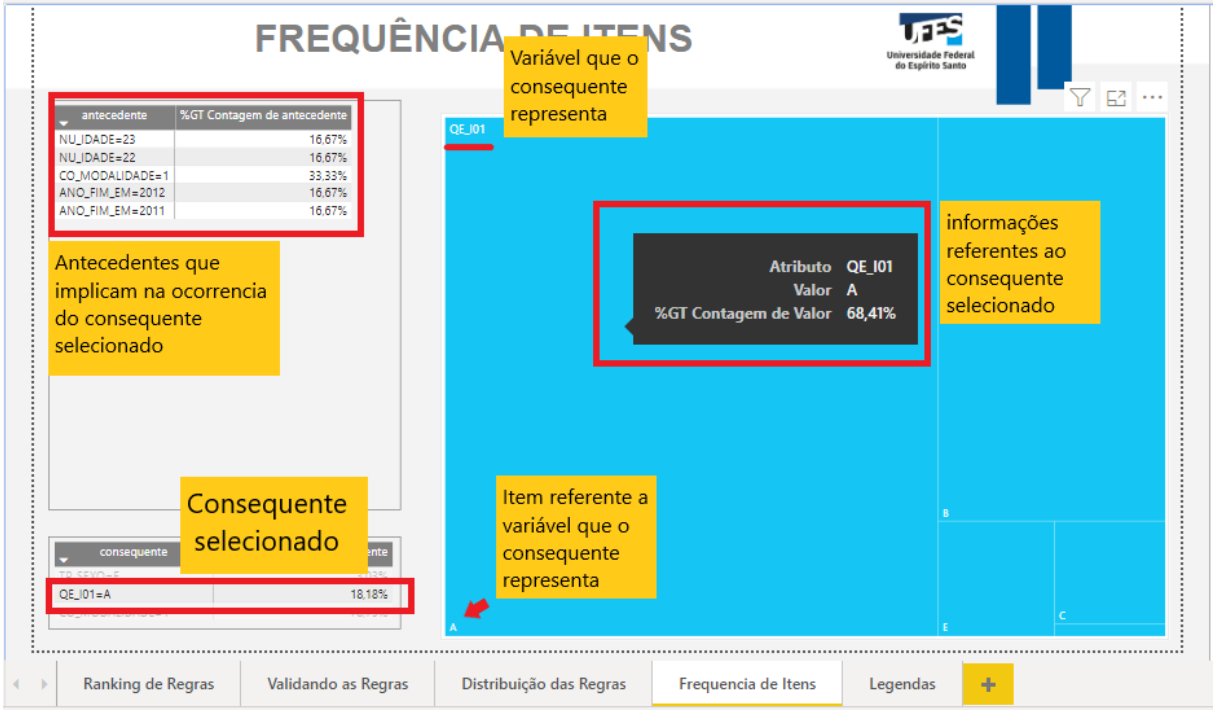

Figura 15 – Exemplo de funcionamento do painel Frequência de Itens. Fonte: Próprio autor.

Exemplo de funcionamento: Ao selecionar na tabela de consequentes, um dos consequentes gerados, na tabela de antecedentes aparecerá todos os antecedentes que implicam na ocorrência do consequente selecionado, e no gráfico Treemap aparecerá o quanto aquele consequente representa na base como um todo. Na Figura 15 é mostrado o funcionamento do painel.

5 Conclusão e Trabalhos Futuros

Ao final da aplicação das etapas de KDD, foram obtidas 23 regras de associação que são classificadas como relevantes e não redundantes para o estudo dos dados de acordo com a poda e a seleção nelas aplicadas.

Através das visualizações geradas no Power BI, notou-se que todas as 23 regras geradas possuíam lift superior a 1, ou seja, seus antecedentes e consequentes estão correlacionados positivamente. A maior parte das regras encontra-se num espaço em que seus lift variam de 1,1 a 1,2 e sua confiança entre 90 e 96%. Também foi possível notar que apenas 3 variáveis apareceram como consequente, dentre elas, CO_MODALIDADE=1, onde a modalidade de ensino é presencial, é o consequente que representa 78,79% da base e sua ocorrência como consequente aparece em 18 das 23 regras geradas, isso implica que 78,26% das regras possui consequente igual a CO_MODALIDADE=1. O que aponta, que mesmo com o crescimento da modalidade de ensino a distância, EaD, o ensino de forma presencial ainda é a modalidade de ensino superior mais frequentadas pelos brasileiros.

As regras geradas indicam relação forte entre a renda familiar e o ensino superior presencial. Uma delas diz que candidatos solteiros e com renda entre 3 a 4,5 salários mínimos frequentam o ensino superior presencial (QE_I01=A,QE_I08=C=>CO_MODALIDADE=1) e algumas regras indicam que candidatos com idade entre 22 e 23 anos também frequentam essa modalidade de ensino, isso implica que o ensino superior presencial é comum entre jovens de classe média alta.

Como proposta para trabalhos futuros, propõe-se realizar a aplicação do Apriori e da função de poda e seleção das melhores regras em partes específicas da base de estudo. Por exemplo, fazer este estudo por regiões do país e comparar as regras geradas na análise como um todo, a fim de descobrir situações específicas de cada região. Outra proposta é a otimização da função de poda e seleção das melhores regras para que o custo computacional seja menor. Por fim, propõe-se utilizar da vasta galeria de gráficos, filtros e aplicabilidades do Power BI, para mostrar novas maneiras de visualizar as regras geradas.

Referências

[1] AIRES, A. C.; ESCOVAR, J. V.; CARDOSO, M. L. Em um mundo conectado, dados armazenados tornam-se protagonistas. 2017. Disponível em: <http://paineira.usp.br/aun/index.php/2017/08/21/em-um-mundo-conectado-dados-armazenados-tornam-se-protagonistas/>. Acesso em: 15 mai. 2019.

[2] GUIA DO ESTUDANTE. O que é e para o que serve o Enade. Disponível em: <https://guiadoestudante.abril.com.br/enade/>. Acesso em: 15 mai. 2019.

[3] PORTAL BRASILEIRO DE DADOS ABERTOS. O que são dados abertos?. Disponível em: <http://dados.gov.br/pagina/dados-abertos>. Acesso em: 29 mai. 2019.

[4] INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA. Microdados. Disponível em: <http://inep.gov.br/web/guest/microdados>. Acesso em: 30 mai. 2019.

[5] PRASS, F. S. KDD-uma visão geral do processo. 2017. Disponível em: <http://fp2.com.br/blog/wp-content/uploads/2012/07/kdd_uma_visao_geral_do_processo.pdf>. Acesso em: 15 mai. 2019.

[6] RAMOS, J. L. C. Processo de KDD. 2016. Disponível em: <https://www.researchgate.net/figure/Figura-7-Processo-de-KDD_fig29_312605379>. Acesso em: 16 mai. 2019.

[7] SALES, A. S.; GOMES, R. R. Aplicação do processo de descoberta de conhecimento em base de dados num processo seletivo de educação a distância. In: CONGRESSO BRASILEIRO DE EDUCAÇÃO EM ENGENHARIA, 2014, Juiz de Fora. Anais... Juiz de Fora: Associação Brasileira de Educação em Engenharia, 2014.

[8] HAN, J.; KAMBER, M.; PEI, J. Data Mining Concepts and Techniques. Elsevier Editora Ltda. 2012. USA.

[9] GOLDSCHIMIDT, R.; PASSOS, E. Data Mining um guia prático. Elsevier Editora Ltda. 2005. 256p. Rio de Janeiro.

[10] FRANÇA, R. S. de; AMARAL, H. J. C. do. Aplicação de Técnicas de Mineração de Dados para o Mapeamento do Conhecimento na Aprendizagem de Programação: Uma Estratégia Baseada na Taxonomia de Bloom. Anais dos Workshops do Congresso Brasileiro de Informática na Educação. 2013.

[11] HORA, H. R. M. Etapas do processo de descoberta de conhecimento em bancos de

dados. 2014. Disponível em: <https://www.researchgate.net/figure/Figura-5-Etapas-do-Processo-de-Descoberta-de-Conhecimento-em-Banco-de-Dados_fig2_263336858>. Acesso em: 16 mai. 2019.

[12] BAKER, R. S. J. D.; DE CARVALHO, A. M. J. B. Mineração de dados educacionais: Oportunidades para o Brasil. Disponível em: <http://www.upenn.edu/learninganalytics/ryanbaker/BD-RBIE-pt-v22.pdf>. Acesso em: 30 mai. 2019.

[13] PATRICIO, T. S.; MAGNONI, M. G. M. Mineração de dados e Big Data na educação. Revista Geminis, v. 9, n. 1. p. 57-75, 2018.

[14] COSTA, E. et. al. Mineração de Dados Educacionais: Conceitos, Técnicas,Ferramentas e Aplicações. Jornada de Atualização em Informática na Educação– JAIE. 2012.

[15] SILVA, E. R. DE A. Mineração de dados educacionais guiado por mapas de conhecimento. 2016. 108 f. (Mestrado em computação aplicada) – Universidade Estadual do Ceará, Fortaleza, 2016.

[16] INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA. Conheça o Inep. Disponível em: <http://portal.inep.gov.br/conheca-o-inep >. Acesso em: 30 mai. 2019.

[17] ENADE. Microdados do ENADE 2017. Manual do Usuário. Disponível no pacote microdados_Enade_2017. Novembro de 2018.

[18] AGRAWAL, R.; IMIELINSKI T. and SWAMI A., Database mining: a performance perspective, in IEEE Transactions on Knowledge and Data Engineering, vol. 5, no. 6, pp. 914-925, Dec. 1993.

[19] GENG, L.; HAMILTON, H. J. Interestingness measures for data mining: A survey. ACM Computing Surveys (CSUR), ACM, v. 38, n. 3, p. 9, 2006.

[20] ROMÃO, W.et al. Extração de Regras de Associação em CT: Algoritmo Apriori. Programa de pós graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Centro Tecnológico, Florianópolis, 2020.

[21] AGRAWAL, R.; SRIKANT, R. Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th International Conference on Very Large Data Bases. USA: Morgan Kaufmann Publishers Inc., 1994. v. 1215, p. 487–499.

[22] CASTRO, Eduardo Petrini Silva. Uma Proposta de Implementação do algoritmo Apriori Distribuído Utilizando o Arcabouço Hadoop Mapreduce. 2014. 76 f. Monografia (Graduação em Ciência da Computação)-Universidade Federal de Lavras, Minas Gerais, 2014.

[23] REGRAS DE ASSOCIAÇÃO. Disponível em: < http://dcm.ffclrp.usp.br/ augusto/teaching/ami/AM-I-Regras-Associacao.pdf >. Acesso em: 27 mai. 2020.

[24] RODRIGUES, D. C.et al. Proposta de Método para Redução do Conjunto de Regras de Associação Resultantes do Algoritmo Apriori. Revista Cereus, Tocantins, Vol. 11, n 3 p.158–177. 2019.

[25] O PARADOXO DE SIMPSON TE MOSTRA QUE NEM TUDO É O QUE PARECE. Disponível em: < http://proec.ufabc.edu.br/gec/o-que-que-a-ciencia-tem/paradoxo-de-simpson/>. Acesso em: 28 mai. 2020.

[26] Han, J. and Kamber, M. (2005). Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, Inc. ix, 3, 5, 6, 9, 10, 11, 12.