UNIVERSIDADE FEDERAL DO PARANÁ

NATHALIA FERREIRA NETTO

MINERAÇÃO DE DADOS NA BASE *ABSENTEEISM AT WORK*

CURITIBA

2019

NATHALIA FERREIRA NETTO

MINERAÇÃO DE DADOS NA BASE *ABSENTEEISM AT WORK*

Trabalho desenvolvido como requisito para aprovação na disciplina SIN195 - Mineração de Dados, do curso de Gestão da Informação da Universidade Federal do Paraná.

Profª. Dra. Denise Tsunoda

CURITIBA

2019

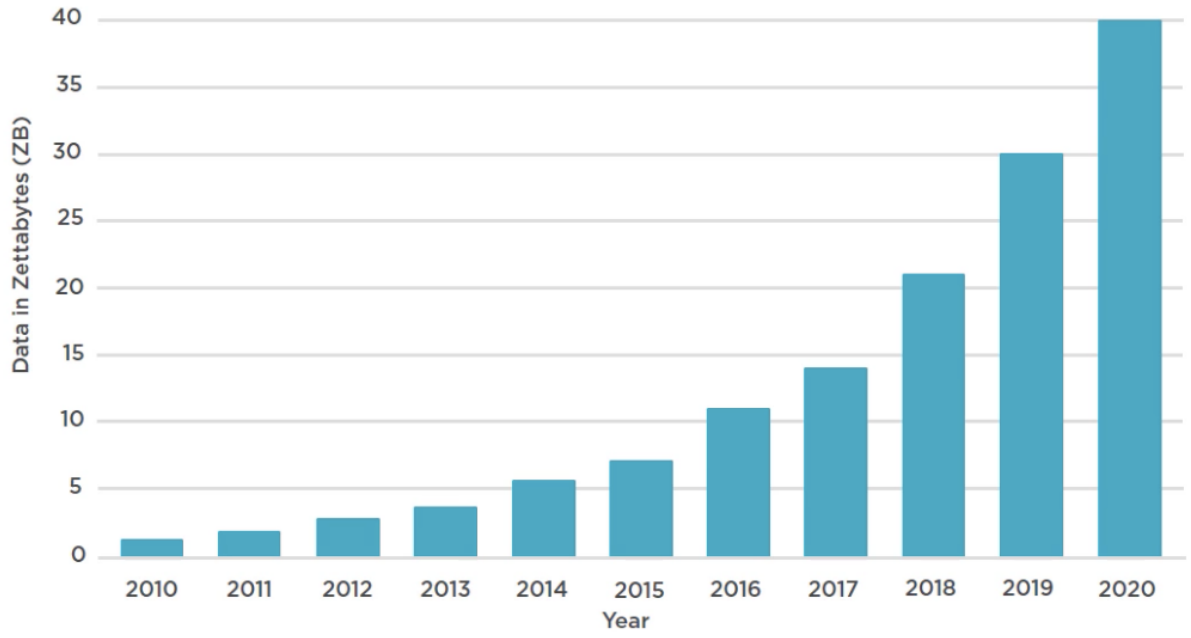
SUMÁRIO

1. INTRODUÇÃO

Desde o surgimento dos sistemas computacionais, um dos principais objetivos das organizações tem sido armazenar dados (CAMILO; SILVA, 2009). De acordo com Camilo e Silva (2009), “nas últimas décadas essa tendência ficou ainda mais evidente com a queda nos custos para a aquisição de hardware, tornando possível armazenar quantidades cada vez maiores de dados”.

Com isso, e num contexto de popularização de tecnologias e até aumento da população, de fato pode-se observar um crescimento exponencial no número de dados produzidos e armazenados, como mostra o relatório de 2013 da UNECE, que previa 40 zettabytes de dados produzidos pela humanidade até o fim da década (FELIX, 2018). A figura 1 mostra o crescimento global de dados ao longo da última década a partir da previsão da UNECE.

FIGURA 1 - CRESCIMENTO GLOBAL DE DADOS



Fonte: Felix (2018)

Com o volume de dados armazenados crescendo diariamente, tornou-se crucial saber o que fazer com os dados armazenados. Com a finalidade de atender

esta demanda, foi proposta, no final da década de 80, a Mineração de Dados, do inglês *Data Mining* (CAMILO; SILVA, 2009).

> Mineração de dados é uma das alternativas mais eficazes para extrair conhecimento a partir de grandes volume de dados, descobrindo relações ocultas, padrões e gerando regras para predizer e correlacionar dados, que podem ajudar as instituições nas tomadas de decisões mais rápidas ou, até mesmo, a atingir um maior grau de confiança (GALVÃO; MARIN, 2009).

Amo (2004) define Mineração de Dados como a aplicação de técnicas inteligentes a fim de se extrair os padrões de interesse em grandes volumes de dados. A autora destaca ainda que a mineração é uma etapa essencial do processo de descoberta de conhecimento em base de dados, do inglês *Knowledge Discovery in Databases* (KDD).

A mineração de dados conta com diferentes tarefas realizadas por algoritmos para extração de padrões, que podem ser divididas entre preditivas e descritivas (GALVÃO; MARIN, 2009). Entre as tarefas de predição se destacam a regressão e classificação e entre as tarefas de descrição estão o agrupamento, a associação e a sumarização (GALVÃO; MARIN, 2009).

A presente pesquisa tem então como objetivo utilizar ferramentas e técnicas de mineração de dados para descobrir padrões em uma base de dados, bem como descrever todo o processo de descoberta de conhecimento.

A base escolhida foi a *Absenteeism at Work*, disponível no repositório de bases de dados públicas UCI - *Machine Learning Repository*. Pretende-se com ela entender quais são os fatores que influenciam na ausência no trabalho e constatando preditivamente possíveis ausências dadas por esses fatores.

2. DESCRIÇÃO DA BASE DE DADOS

Absenteeism at work em tradução livre significa absenteísmo no trabalho. Como o próprio nome indica, na base contém registros de não comparecimento no trabalho de funcionários de uma empresa de correio (entregas) do Brasil, e datam de julho de 2007 a julho de 2010.

O quadro 1 apresenta as principais características da base.

QUADRO 1 - CARACTERÍSTICAS DA BASE

Características do conjunto de dados	Séries temporais multivariadas
Características dos atributos	Categóricos, Inteiros, Reais
Tarefas associadas	Classificação, Agrupamento
Número de instâncias	740
Número de atributos	21
Valores nulos	Nenhum
Área	Administração / Gestão de Pessoas
Data de doação para a UCI	05/04/2018
Tipo de formato	Matriz

Fonte: UCI - *Machine Learning Repository*

O site da UCI ainda informa que os proprietários e doadores da base são Andrea Martiniano, Ricardo Pinto Ferreira e Renato Jose Sassi, do Programa de Pós-Graduação em Informática e Gestão do Conhecimento da Universidade Nove de Julho de São Paulo - SP.

Os atributos são descritos a seguir:

1. Identificação individual (ID) (de 0 a 36);

2. Motivo da ausência (CID) (de 0 a 28);

3. Mês de ausência (número real);

4. Dia da semana (segunda-feira (2), terça-feira (3), quarta-feira (4), quinta-feira (5), sexta-feira (6));

5. Estações do ano (verão (1), outono (2)) , inverno (3), primavera (4));

6. Despesas de transporte (número real);

7. Distância entre residência e trabalho (quilômetros) (número real);

28. Tempo de serviço (número inteiro);

9. Idade (número inteiro);

10. Carga de trabalho Média / dia (número real);

11. Objetivo atingido;

12. Falha disciplinar (sim = 1; não = 0);

13. Educação (ensino médio (1), graduação (2), pós-graduação (3), mestrado e doutorado (4));

14. Filho (número de filhos) (número real);

15. Bebedor social (sim = 1; não = 0);

16. Fumante social (sim = 1; não = 0);

17. Animal de estimação (número de animais) (número real);

18. Peso (número real);

19. Altura (número real);

20. Índice de massa corporal (número real);

21. Tempo de absenteísmo em horas (número real).

Em relação ao segundo atributo (Motivo da ausência) são 21 categorias de ausências atestadas pelo Código Internacional de Doenças (CID), como segue:

01 Certas doenças infecciosas e parasitárias;

02 Neoplasias;

03 Doenças do sangue e órgãos formadores de sangue e certas desordens envolvendo o mecanismo imunológico;

04 Endócrino , doenças nutricionais e metabólicas;

05 Perturbações mentais e comportamentais;

06 Doenças do sistema nervoso;

07 Doenças do olho e anexos;

08 Doenças do ouvido e da mastóide;

09 Doenças do sistema circulatório;

10 Doenças do sistema respiratório X Doenças do sistema respiratório;

11 Doenças do sistema digestivo;

12 Doenças da pele e tecido subcutâneo;

13 Doenças do sistema músculo-esquelético e do tecido conjuntivo;

14 Doenças do aparelho geniturinário;

15 Gravidez, parto e puerpério;

16 Certas condições originadas no período perinatal;

17 Malformações congênitas, deformações e anormalidades cromossômicas;

18 Sintomas, sinais e achados clínicos e laboratoriais anormais, não classificados em outra parte;

19 Lesões, envenenamentos e algumas outras conseqüências de causas externas;

20 Causas externas de morbimortalidade;

21 Fatores que influenciam o estado de saúde e o contato com os serviços de saúde.

E 7 outras categorias: acompanhamento de paciente (22), consulta médica (23), doação de sangue (24), exame laboratorial (25), ausência injustificada (26), fisioterapia (27), consulta odontológica (28).

O cabeçalho do arquivo .arff e a primeira instância (a fim de demonstrar um exemplo) são apresentados na figura 2.

FIGURA 2 - CABEÇALHO DA BASE DE DADOS NO ARQUIVO .ARFF

```
1  @relation Absenteeism_at_work_A
2
3  @attribute ID {31.0, 27.0, 19.0, 30.0, 7.0, 20.0, 24.0, 32.0, 3.0, 33.0, 26.0,
   29.0, 18.0, 25.0, 17.0, 14.0, 16.0, 23.0, 2.0, 21.0, 36.0, 15.0, 22.0, 5.0, 12.0,
   9.0, 6.0, 34.0, 10.0, 28.0, 13.0, 11.0, 1.0, 4.0, 8.0, 35.0}
4  @attribute Reason_for_absence {17.0, 3.0, 15.0, 4.0, 21.0, 2.0, 9.0, 24.0, 18.0,
   1.0, 12.0, 5.0, 16.0, 7.0, 27.0, 25.0, 8.0, 10.0, 26.0, 19.0, 28.0, 6.0, 23.0,
   22.0, 13.0, 14.0, 11.0, 0.0}
5  @attribute Month_of_absence REAL
6  @attribute Day_of_the_week {5.0, 2.0, 3.0, 4.0, 6.0}
7  @attribute Seasons {4.0, 1.0, 2.0, 3.0}
8  @attribute Transportation_expense REAL
9  @attribute Distance_from_Residence_to_Work REAL
10 @attribute Service_time INTEGER
11 @attribute Age INTEGER
12 @attribute Work_load_Average/day_ REAL
13 @attribute Hit_target REAL
14 @attribute Disciplinary_failure {1.0, 0.0}
15 @attribute Education REAL
16 @attribute Son REAL
17 @attribute Social_drinker {1.0, 0.0}
18 @attribute Social_smoker {1.0, 0.0}
19 @attribute Pet REAL
20 @attribute Weight REAL
21 @attribute Height REAL
22 @attribute Body_mass_index REAL
23 @attribute Absenteeism_time_in_hours REAL
24
25 @data
26 11.0, 26.0, 7.0, 3.0, 1.0, 289.0, 36.0, 13.0, 33.0, 239554.0, 97.0, 0.0, 1.0, 2.0,
   1.0, 0.0, 1.0, 90.0, 172.0, 30.0, 4.0
```

Fonte: arquivo .arff na interface do editor de texto Sublime Text 3

3. METODOLOGIA

3.1. SELEÇÃO DA BASE DE DADOS

A seleção da base de dados se deu por uma triagem das bases públicas contidas no repositório UCI - *Machine Learning Repository* de acordo com três critérios principais: área de interesse, tarefa de mineração indicada e número de instâncias (entre 500 e 1000).

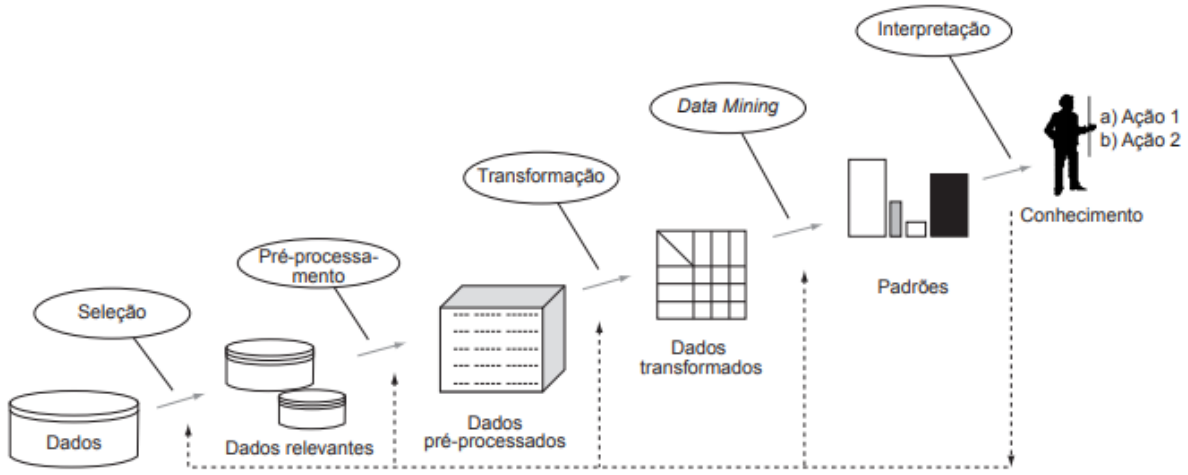
Após análise das bases que passaram pela triagem inicial, a *Absenteeism at Work* foi escolhida por, além de atender os critérios de forma satisfatória, ser uma base de aproximadamente apenas 1 ano de disponibilidade, tendo origem no Brasil e, por fim, pela disponibilidade de informações sobre os dados.

3.2. PROCESSO DE KDD

Após a seleção dos dados, dá-se continuidade ao trabalho de acordo com as etapas do processo de descoberta de conhecimento na base de dados (KDD).

Essas etapas são apresentadas por Steiner et al. (2006) na figura 3, e sua aplicação será descrita detalhadamente no tópico 5 do presente documento, com exceção da primeira etapa (seleção) já descrita.

FIGURA 3 - PROCESSO DE KDD



Fonte: Steiner et al. (2006)

3.3. SELEÇÃO DA TAREFA DE MINERAÇÃO

As tarefas indicadas pela UCI para esta base são a classificação e o agrupamento.

Por ser, como supracitado, uma tarefa preditiva, a tarefa indicada foi a classificação, descrita por Amo (2004) como:

> o processo de encontrar um conjunto de modelos (funções) que descrevem e distinguem classes ou conceitos, com o propósito de utilizar o modelo para predizer a classe de objetos que ainda não foram classificados. O modelo construído baseia-se na análise prévia de um conjunto de dados de amostragem ou dados de treinamento, contendo objetos corretamente classificados (AMO, 2004).

3.4. SELEÇÃO DOS ALGORITMOS

Já a seleção dos algoritmos ocorreu a partir de uma triagem daqueles disponíveis na ferramenta Weka 3.8 e de acordo com a taxa de acerto gerada, ou seja, o número de acertos que o algoritmo alcançava após ser treinado com a base pré-processada e transformada. Dessa maneira, a taxa de acerto mínima para a seleção foi de 70%.

A ferramenta Weka 3.8 foi desenvolvida pela Universidade de Waikato e já estava pré-definida para ser usada neste trabalho por ser uma das principais ferramentas para mineração de dados.

Assim, foram definidos os algoritmos PART e *Random Subspace*, detalhados no tópico 4 do presente documento, que aborda as heurísticas.

4. HEURÍSTICAS

4.1. PART

4.1.1. Histórico

O algoritmo PART foi introduzido por Frank e Witten (1998) no final da década de 90 como uma proposta de combinar os algoritmos de aprendizagem de regras (*Rules*) C4.5 e RIPPER, eliminando seus respectivos problemas e combinando as vantagens de seus paradigmas para criar regras que evitassem a otimização global mas que fossem precisas e compactas.

As críticas dos autores em relação a estes dois métodos focam no excesso de complexidade, consumindo muito tempo, no caso do C4.5, e o que os autores chamam de “generalização apressada” para o RIPPER, já que as generalizações são feitas antes de se saber das implicações.

4.1.2. Algoritmo

O nome PART é derivado de *Partial Decision Trees*, ou árvores de decisão parciais, em português. O nome se dá pelo fato do PART gerar regras a partir de árvores de decisão parciais que são repetidamente criadas e descartadas. Uma árvore de decisão parcial é uma árvore de decisão comum que contém ramificações para subárvores indefinidas (FRANK; WITTEN, 1998).

O PART se utiliza de dois métodos herdados dos paradigmas C4.5 e RIPPER: a geração da árvore de decisão e posteriormente o mapeamento da árvore de decisão em regras aplicando processos de refinamento e a utilização do paradigma dividir-para-conquistar. O algoritmo trabalha construindo a regra e estimando sua cobertura como no processo de dividir-para-conquistar repetidamente até que todas as instâncias estejam cobertas. Constrói uma árvore de decisão parcial em cada interação e converte os ramos com a mais alta cobertura em regras (PIRES, 2011).

Usar uma árvore podada para obter uma regra em vez de construí-la de forma incremental adicionando conjunções de uma vez evita o problema de poda excessiva do paradigma dividir-para-conquistar. E usar esse paradigma em conjunto com a árvore de decisão adiciona flexibilidade e velocidade ao algoritmo (FRANK; WITTEN, 1998).

Abaixo, pode-se observar um resumo do algoritmo que constrói e poda a árvore parcial (FRANK; WITTEN, 1998):

Procedimento Expandir subconjunto
escolha a divisão de um determinado conjunto de exemplos em subconjuntos;
enquanto houver subconjuntos que não foram expandidos **e**
todos os subconjuntos expandidos até agora são folhas
escolha o próximo subconjunto a ser expandido e expanda;
se todos os subconjuntos expandidos forem folhas **e**
erro estimado para subárvore >= erro estimado para nó
desfazer a expansão em subconjuntos e tornar o nó uma folha.

Dessa maneira, ocorre o procedimento padrão do C4.5 para se obter uma árvore parcial a partir de cada interação e torna as melhores folhas em regras.

Como exemplo prático de aplicação do algoritmo, pode-se citar o trabalho “A mineração de dados e a qualidade de conhecimentos extraídos dos boletins de ocorrência das rodovias federais brasileiras” de Costa, Bernardini e Filho (2014), em que foi utilizado o algoritmo PART para se obter regras em relação às causas de acidentes em rodovias brasileiras. As regras obtidas estão apresentadas na figura 4, a seguir.

FIGURA 4 - EXEMPLO DE REGRAS OBTIDAS DA APLICAÇÃO DO PART

- SE Tipo de Acidente = Atropelamento de Animal E Modelo da Pista = Reta E Tipo de Veículo = Automóvel E Período do Dia = Noite ENTÃO Causa do Acidente = Animais na Pista. (#Cob = 1036; #Incorr = 17)
- SE Tipo de Acidente = Incêndio E Estado Físico da Pessoa = Ileso E Modelo da Pista = Reta ENTÃO Causa do Acidente = Defeito Mecânico no Veículo. (#Cob = 628; #Incorr = 7)
- SE Tipo de Acidente = Incêndio E Dia da Semana = Quarta-feira E Período do Dia = Manhã ENTÃO Causa do Acidente = Motorista Dormindo. (#Cob = 16)
- SE Tipo de Acidente = Atropelamento de Pessoa E Modelo da Pista = Reta E Período do Dia = Tarde e Tipo de Veículo = Automóvel ENTÃO Causa do Acidente = Falta de Atenção. (#Cob = 343; #Incorr = 85)

Fonte: Costa, Bernardini e Filho (2014)

4.2. RANDOM SUBSPACE

4.2.1. Histórico

Também no final da década de 90, a cientista da computação Tin Kam Ho (1998) apresentou o algoritmo *Random Subspace Method* (RSM). Seu objetivo foi propor um mecanismo para a construção das chamadas florestas de decisão, combinando diversas árvores de decisão. A autora se baseou em algo que ela mesma já havia proposto em 1995: combinar várias árvores construídas em subespaços selecionados aleatoriamente para obter um aumento quase monotônico na precisão da generalização, preservando a precisão perfeita nos dados de treinamento, desde que os recursos sejam suficientes para distinguir todas as amostras pertencentes a diferentes classes ou que não exista ambiguidade intrínseca ao conjunto de dados (HO, 1998).

4.2.2. Algoritmo

Para construir a floresta de decisão, o algoritmo consiste em:

Selecionar um subconjunto de recursos em cada divisão durante o crescimento das árvores;

Todos os outros recursos não selecionados recebem um valor constante (0);

As instâncias são projetadas no referido espaço de recurso;

As árvores são treinadas no referido subespaço;

A probabilidade condicionada de pertencer a cada classe é armazenada em cada folha;

$$P\left(c\middle|v_j(x)\right) = \frac{P\left(c\middle|v_j(x)\right)}{\sum_{k=1}^{n_c} P\left(c_k, v_j(x)\right)}$$

A combinação de classificadores (função de decisão) é feita pela maioria;

$$g_c(x) = \frac{1}{n_t}\sum_{j=1}^{n_t} \hat{P}\left(c\middle|v_j(x)\right)$$

A similaridade entre os membros é calculada, avaliando um conjunto fixo de n exemplos de teste, assumindo pesos iguais, como:

$$\hat{s}_{i,j} = \frac{1}{n}\sum_{k=1}^{n} f\left(x_k\right),$$

Onde:

$$f\left(\boldsymbol{x}_k\right) = \begin{cases} 1 & \text{if } c_i\left(\boldsymbol{x}_k\right) = c_j\left(\boldsymbol{x}_k\right) \\ 0 & \text{otherwise} \end{cases}.$$

De forma simplificada um conjunto de modelos que emprega esse método pode ser construído da seguinte maneira:

A. Defina o número de pontos de treinamento *N* e o número de recursos nos dados de treinamento *D*;
B. Escolha *L* para ser o número de modelos individuais no conjunto;

C. Para cada modelo individual *l*, escolha *nl (nl <N)* para ser o número de pontos de entrada para *l*. É comum ter apenas um valor de *nl* para todos os modelos individuais;
D. Para cada modelo individual de *l*, crie um conjunto de treinamento, escolhendo características *dl* de *D* com substituição e treine o modelo.

Para aplicar o modelo de conjunto a um ponto invisível, combine as saídas dos modelos individuais *L* por votação majoritária ou combinando as probabilidades posteriores.

Um exemplo de aplicação prática pode ser observado no trabalho "*Random Subspace Method in Text Categorization*" de Kamel e Duin (2010), em que o método de floresta de decisão se apresenta vantajoso para classificar um espaço de alto recurso dimensional e a alta proporção de recurso para instância, tal como a categorização de textos.

5. DESCRIÇÃO DO PROCESSO DE DESCOBERTA DE CONHECIMENTO

5.1. ETAPAS DO KDD

5.1.1. Seleção

A seleção da base já foi descrita no tópico 3 deste documento. Cabe ressaltar que para que se concretizasse a escolha da base, essa foi dada pela área de interesse, e a partir disso também, houve um estudo sobre o domínio da base, para que assim fossem selecionados apenas os atributos de maior interesse para atender o objetivo definido, supracitado, de entender os fatores que influenciam a ausência no trabalho para prever futuras ausências.

Foram então desconsiderados do conjunto de dados os atributos (18) Peso e (19) Altura, uma vez que a base também conta com o atributo (20) Índice de massa corporal, que é dado em função da altura e peso do funcionário. Da mesma forma foi excluído o atributo (5) Estações do ano por ser dado em função do (3) Mês de ausência. Não foram considerados também o (1) ID por não ser de fato um fator que possa ser considerado para fins de previsão, e sim apenas a identificação do funcionário.

5.1.2. Pré-processamento

A base já contava com atributos normalizados, não continha nenhum valor nulo e nenhuma inconsistência. Apenas contava com atributos contínuos que foram discretizados na etapa de transformação.

5.1.3. Transformação

A etapa de transformação foi composta pela discretização dos seguintes atributos contínuos:

6. Despesas de transporte;
7. Distância entre residência e trabalho;
8. Tempo de serviço;
9. Idade;
10. Carga de trabalho Média / dia;
11. Objetivo atingido;
20. Índice de massa corporal;
21. Tempo de absenteísmo em horas.

Além disso, cada mês, número de filhos e animais passaram a ser categorias, e o resultado final do conjunto de dados aceito em cada atributo é apresentado na figura 5, abaixo.

FIGURA 5 - CABEÇALHO DO ARQUIVO .ARFF APÓS ETAPA DE TRANSFORMAÇÃO

```
@attribute Reason_for_absencee {0, 17, 3, 15, 4, 21, 2, 9, 24, 18, 1,
12, 5, 16, 7, 27, 25, 8, 10, 26, 19, 28, 6, 23, 22, 13, 14, 11}
@attribute Month_of_absence {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}
@attribute Day_week {5, 2, 3, 4, 6}
@attribute Trans_Exp_Discretizado {1, 2, 3, 4}
@attribute distance_discretizado {1, 2, 3, 4}
@attribute serv-time_discretizado {1, 2, 3}
@attribute age_discretizado {1, 2, 3}
@attribute work_load_discretizado {1, 2, 3}
@attribute hit_target_discretizado {1, 2, 3}
@attribute Disciplinary_failure {1, 0}
@attribute Education {1, 2, 3, 4}
@attribute Son {0, 1, 2, 3, 4}
@attribute Social_drinker {1, 0}
@attribute Social_smoker {1, 0}
@attribute Pet {0, 1, 2, 4, 5, 8}
@attribute bmi_discretizado {1, 2, 3, 4}
@attribute absenteeism_discretizado {0, 1, 2, 3, 4, 5}
```

Fonte: arquivo .arff na interface do editor de texto Sublime Text 3

Para detalhar a discretização dos atributos, essa se deu pelo método de corte, gerando os seguintes parâmetros para definir as faixas categóricas:

6. Despesas de transporte foi dividido em 4 faixas (f1, f2, f3, f4):
    118 <= f1 < 185,5 <= f2 < 253 <= f3 < 320,5 <= f4 <= 388

7. Distância entre residência e trabalho foi dividida em 4 faixas (f1, f2, f3, f4):
    5 <= f1 < 16,75 <= f2 < 28,5 <= f3 < 40,25 <= f4 <= 52

8. Tempo de serviço 3 foi dividido em 3 faixas (f1, f2, f3):
    1 <= f1 < 10,3 <= f2 < 19,7 <= f3 <= 29

9. Idade foi dividida em 3 faixas (f1, f2, f3):
    27 <= f1 < 37,3 <= f2 < 47,7 <= f3 <= 58

10. Carga de trabalho Média / dia foi dividida em 3 faixas (f1, f2, f3):
    205917 <= f1 < 263572,7 <= f2 < 321228,3 <= f3 <= 378884

11. Objetivo atingido foi dividido em 3 faixas (f1, f2, f3):
    81 <= f1 < 87,3 <= f2 < 93,7 <= f3 <= 100

Dos que não foram discretizados pelo método de corte:

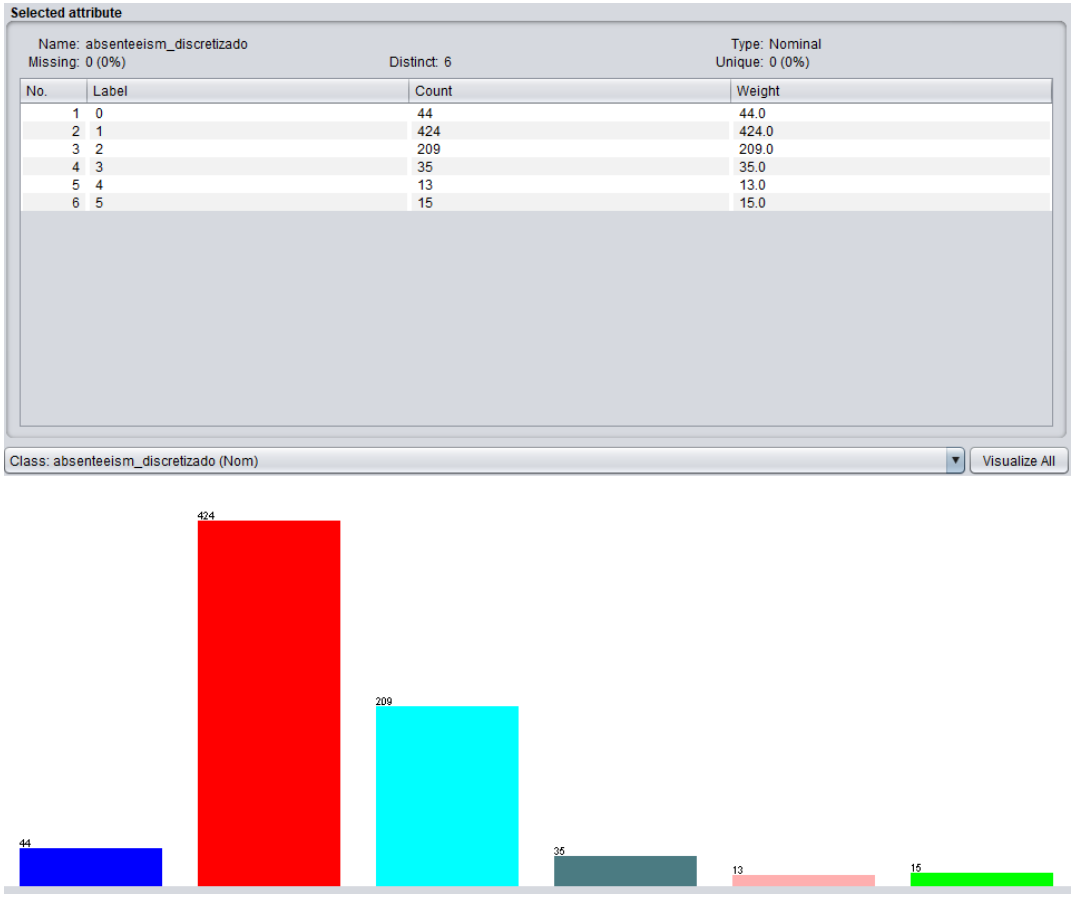
20. Índice de massa corporal foi dividido pela tabela de IMC que define as seguintes faixas:

    19 <= Saudável (1) < 25 <= Sobrepeso (2) < 30 <= Obesidade (3) < 35 <= Obesidade severa (4) <= 38

21. Tempo de absenteísmo em horas foi categorizado de forma a ser possível diferenciar a não ausência (0), o atraso (1), a falta de 1 dia (2), falta de 2 a 3 dias (3), faltas de 4 a 5 dias (4) e, finalmente, falta de mais de uma semana (em dias úteis) (5). Este foi definido como atributo meta e apesar de ter sido mantido na base seus dados originais (contando ao todo duas colunas), para análise no Weka foi considerado apenas o atributo categorizado.

O resultado da categorização proposta pode ser visualizado na ferramenta Weka, como mostra a figura 6.

FIGURA 6 - INTERFACE DO WEKA REPRESENTANDO A DISTRIBUIÇÃO DAS CATEGORIAS DO ATRIBUTO META



Fonte: Interface da ferramenta Weka 3.8

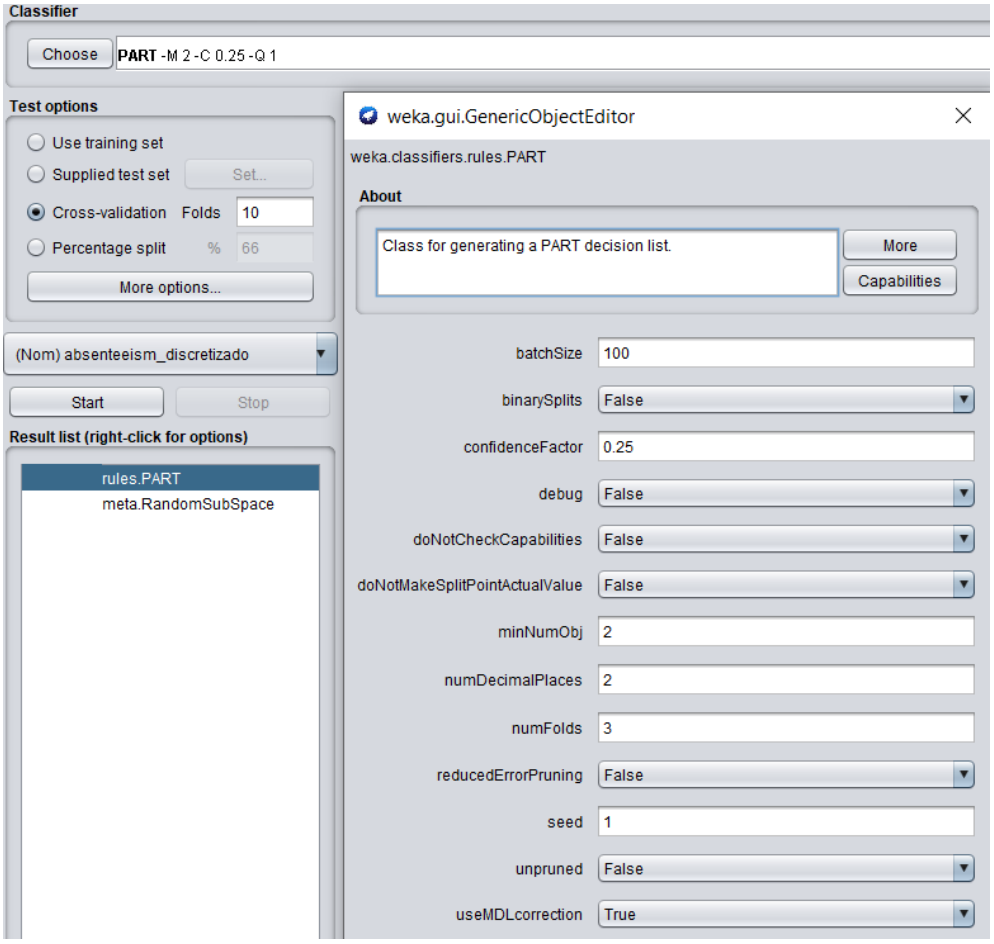
5.1.4. Mineração

Para esta etapa foi utilizada a ferramenta Weka 3.8, já apresentada no tópico 3 deste trabalho.

Já tendo sido definida a tarefa de mineração (classificação) e os algoritmos (PART e RSM), bastou carregar o arquivo .arff para o Weka e rodar os algoritmos nos parâmetros padrões predefinidos da ferramenta para se obter os primeiros resultados e depois com o estudo e teste de diferentes parâmetros, chegar a um resultado com maior taxa de acerto.

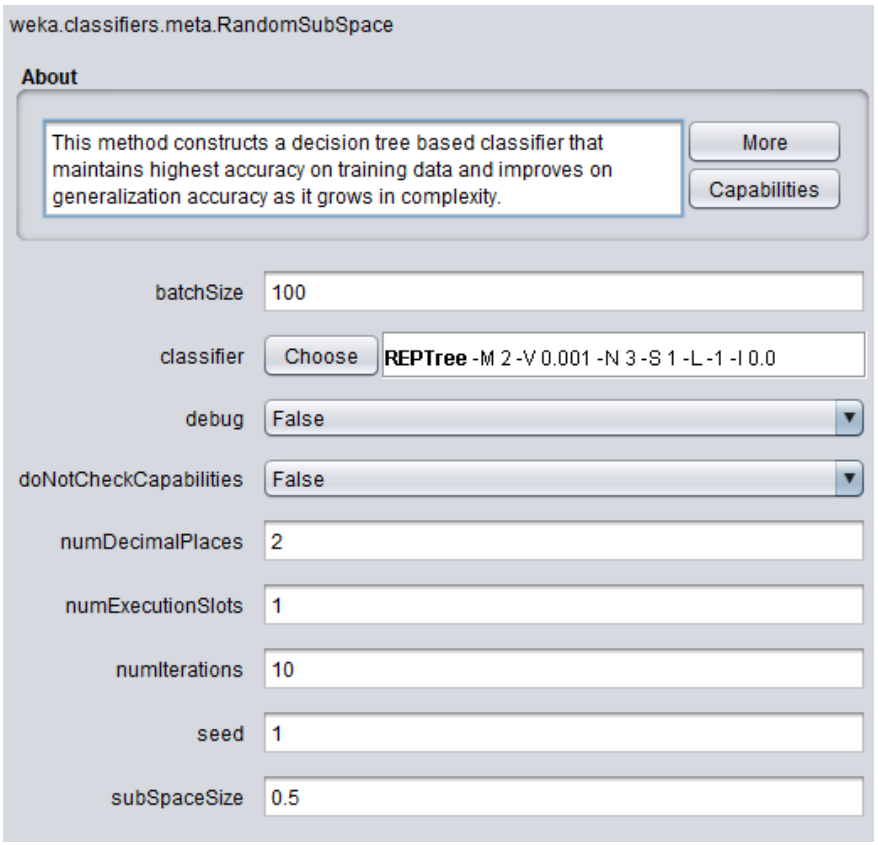
Os parâmetros padrões da ferramenta podem ser observados nas figuras 7 e 8.

FIGURA 7 - PARÂMETROS PADRÃO PARA O PART



Fonte: Interface da ferramenta Weka 3.8

FIGURA 8 - PARÂMETROS PADRÃO PARA O RSM



Fonte: Interface da ferramenta Weka 3.8

5.1.5. Interpretação

A fase de interpretação é essencial para a construção do conhecimento a partir dos dados. Os resultados apresentados pelos algoritmos são em linguagem técnica e demanda do usuário ou analista entender e interpretá-los para a descoberta de conhecimento relevante.

Essa fase será detalhada na parte final do presente documento, a partir do próximo tópico.

6. RESULTADOS OBTIDOS

6.1. RESULTADOS OBTIDOS PARA A BASE ORIGINAL

Não foi possível processar a base original nos algoritmos escolhidos por conta dos atributos contínuos.

6.2. RESULTADOS OBTIDOS PARA A BASE DISCRETIZADA

6.2.1. PART

Para este algoritmo, foram mantidos os parâmetros padrões do Weka. Com isso, os resultados finais obtidos são apresentados na figura 9.

FIGURA 9 - RESULTADOS APRESENTADOS PELO ALGORITMO PART

```
Number of Rules  :      45


Time taken to build model: 0.1 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         547               73.9189 %
Incorrectly Classified Instances       193               26.0811 %
Kappa statistic                          0.5426
Mean absolute error                      0.1056
Root mean squared error                  0.2556
Relative absolute error                 53.8863 %
Root relative squared error             81.8259 %
Total Number of Instances              740

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,977    0,000    1,000      0,977    0,989      0,988    0,988     0,979     0
                 0,854    0,228    0,834      0,854    0,844      0,629    0,874     0,878     1
                 0,675    0,173    0,605      0,675    0,638      0,486    0,809     0,588     2
                 0,029    0,020    0,067      0,029    0,040      0,013    0,651     0,101     3
                 0,000    0,015    0,000      0,000    0,000      -0,016   0,608     0,050     4
                 0,000    0,006    0,000      0,000    0,000      -0,011   0,521     0,024     5
Weighted Avg.    0,739    0,181    0,711      0,739    0,724      0,556    0,840     0,733

=== Confusion Matrix ===

   a   b   c   d   e   f   <-- classified as
  43   1   0   0   0   0 |   a = 0
   0 362  53   4   3   2 |   b = 1
   0  52 141   8   7   1 |   c = 2
   0  13  20   1   1   0 |   d = 3
   0   4   8   0   0   1 |   e = 4
   0   2  11   2   0   0 |   f = 5
```

Fonte: Interface do Weka 3.8

Destaca-se a geração de 45 regras e a taxa de acerto de 73,9%.

As regras obtidas foram:

Disciplinary_failure = 0 AND

Reason_for_absencee = 28 AND
Pet = 0: 1 (79.0/5.0)

Disciplinary_failure = 0 AND
Reason_for_absencee = 27: 1 (69.0/1.0)

Disciplinary_failure = 1: 0 (40.0)

Reason_for_absencee = 28 AND
Social_smoker = 0: 1 (31.0/1.0)

Reason_for_absencee = 23 AND
Education = 1: 1 (129.0/9.0)

Reason_for_absencee = 22: 2 (38.0/5.0)

Reason_for_absencee = 25: 1 (31.0/5.0)

Reason_for_absencee = 23: 1 (20.0/4.0)

Reason_for_absencee = 18 AND
Day_week = 3: 2 (8.0/1.0)

Reason_for_absencee = 18 AND
Day_week = 2: 2 (5.0)

Reason_for_absencee = 26: 2 (33.0/9.0)

Reason_for_absencee = 21: 2 (6.0/2.0)

Reason_for_absencee = 0: 0 (3.0)

Social_drinker = 0 AND
Reason_for_absencee = 10: 2 (11.0/3.0)

Social_drinker = 0 AND
Reason_for_absencee = 7: 1 (9.0/3.0)

Reason_for_absencee = 24: 2 (3.0)

Reason_for_absencee = 5: 2 (3.0/1.0)

Social_drinker = 0 AND
Reason_for_absencee = 14: 1 (6.0)

Reason_for_absencee = 16: 1 (3.0)

Reason_for_absencee = 15: 2 (2.0)

Reason_for_absencee = 4: 1 (2.0/1.0)

Reason_for_absencee = 18 AND
Day_week = 5: 2 (2.0)

Reason_for_absencee = 18 AND
work_load_discretizado = 1: 2 (3.0/1.0)

Reason_for_absencee = 18: 1 (3.0)

Reason_for_absencee = 8 AND
work_load_discretizado = 2: 2 (3.0/1.0)

Reason_for_absencee = 1: 2 (16.0/4.0)

Education = 3 AND
Reason_for_absencee = 13: 1 (13.0/4.0)

Reason_for_absencee = 6: 2 (8.0/2.0)

Reason_for_absencee = 12 AND
age_discretizado = 1 AND
Trans_Exp_Discretizado = 1: 3 (3.0)

Reason_for_absencee = 11 AND
hit_target_discretizado = 3: 2 (20.0/11.0)

Reason_for_absencee = 10: 2 (14.0/5.0)

Reason_for_absencee = 11: 1 (6.0/1.0)

Day_week = 5 AND
Trans_Exp_Discretizado = 2: 2 (6.0/1.0)

Day_week = 5: 1 (17.0/9.0)

Reason_for_absencee = 12: 1 (5.0/1.0)

Reason_for_absencee = 14 AND
Trans_Exp_Discretizado = 3 AND
Day_week = 2: 4 (3.0/1.0)

Reason_for_absencee = 14: 2 (5.0/1.0)

Month_of_absence = 6: 2 (6.0/2.0)

Reason_for_absencee = 13: 2 (31.0/16.0)

Month_of_absence = 8: 2 (3.0/1.0)

Month_of_absence = 11: 3 (3.0/1.0)

work_load_discretizado = 3: 2 (9.0/3.0)

Education = 1 AND
work_load_discretizado = 1: 2 (19.0/9.0)

Para interpretação das regras, é necessário entender o que os elementos apresentados representam. Por exemplo:

Reason_for_absencee = 23 AND <- primeira instância e operador "E"

Education = 1: 1 (129.0/9.0) <- segunda instância e, após os dois pontos, a decisão.

O significado de (A / B) é o seguinte:

A: o número total (peso) de instâncias cobertas pela regra;

B: o número (peso) de instâncias classificadas incorretamente.

Neste caso a interpretação da regra é: se a razão da ausência é uma consulta médica e o nível educacional for ensino médio, então é registrado um atraso.

Para:

Disciplinary_failure = 1: 0 (40.0)

Se a pessoa já teve alguma falha disciplinar, ela não falta.

E assim por diante.

6.2.2. Random Subspace

Para o segundo algoritmo, em relação aos parâmetros, destaca-se que o classificador padrão REPTree foi mantido por sua velocidade, no entanto o número

de interações foi aumentado de 10 para 50. Os demais parâmetros também foram mantidos.

Como resultado, observa-se na figura 10, a taxa de acerto de 78,8% e a criação de 50 árvores de decisão.

FIGURA 10 - RESULTADOS OBTIDOS COM O RSM

```
Time taken to build model: 0.39 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         554               74.8649 %
Incorrectly Classified Instances       186               25.1351 %
Kappa statistic                          0.5239
Mean absolute error                      0.1376
Root mean squared error                  0.2448
Relative absolute error                 70.2366 %
Root relative squared error             78.3792 %
Total Number of Instances              740

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0,977     0,000     1,000       0,977   0,989      0,988    0,980     0,979     0
               0,917     0,402     0,754       0,917   0,828      0,555    0,903     0,916     1
               0,584     0,111     0,674       0,584   0,626      0,495    0,864     0,702     2
               0,000     0,000     ?           0,000   ?          ?        0,683     0,124     3
               0,000     0,000     ?           0,000   ?          ?        0,775     0,111     4
               0,000     0,000     ?           0,000   ?          ?        0,725     0,058     5
Weighted Avg.  0,749     0,262     ?           0,749   ?          ?        0,880     0,790

=== Confusion Matrix ===

   a   b   c   d   e   f   <-- classified as
  43   1   0   0   0   0 |   a = 0
   0 389  35   0   0   0 |   b = 1
   0  87 122   0   0   0 |   c = 2
   0  21  14   0   0   0 |   d = 3
   0  10   3   0   0   0 |   e = 4
   0   8   7   0   0   0 |   f = 5
```

Fonte: Interface do Weka 3.8

Para exemplificar a apresentação das árvores de decisão foi selecionada uma delas:

@attribute distance_discretizado {1,2,3,4}

@attribute work_load_discretizado {1,2,3}

@attribute Education {1,2,3,4}

@attribute Disciplinary_failure {1,0}

@attribute Day_week {5,2,3,4,6}

@attribute Trans_Exp_Discretizado {1,2,3,4}

@attribute age_discretizado {1,2,3}

@attribute hit_target_discretizado {1,2,3}

@attribute absenteeism_discretizado {0,1,2,3,4,5}

@data

Classifier Model

REPTree

==========

Disciplinary_failure = 1 : 0 (28/0) [12/0]

Disciplinary_failure = 0

| Trans_Exp_Discretizado = 1 : 1 (206/57) [101/30]

| Trans_Exp_Discretizado = 2

| | Day_week = 5

| | | distance_discretizado = 1 : 1 (4/1) [4/2]

| | | distance_discretizado = 2 : 1 (16/2) [3/1]

| | | distance_discretizado = 3 : 2 (2/0) [3/0]

| | | distance_discretizado = 4 : 2 (1/0) [0/0]

| | Day_week = 2

| | | work_load_discretizado = 1

| | | | age_discretizado = 1 : 2 (8/3) [5/1]

| | | | age_discretizado = 2 : 1 (3/1) [2/1]

| | | | age_discretizado = 3 : 1 (2/1) [0/0]

| | | work_load_discretizado = 2

| | | | distance_discretizado = 1 : 1 (2/1) [2/1]

| | | | distance_discretizado = 2 : 1 (6/3) [6/1]

| | | | distance_discretizado = 3 : 2 (1/0) [1/0]
| | | | distance_discretizado = 4 : 1 (2/1) [0/0]
| | | work_load_discretizado = 3 : 2 (4/1) [1/0]
| | Day_week = 3 : 1 (34/12) [13/2]
| | Day_week = 4 : 1 (32/10) [12/3]
| | Day_week = 6 : 1 (24/8) [12/7]
| Trans_Exp_Discretizado = 3 : 1 (78/42) [52/21]
| Trans_Exp_Discretizado = 4 : 2 (40/12) [18/4]

Size of the tree : 26

Sua visualização é dada pelos níveis da árvore partindo do mais alto até o mais baixo (a folha) com a decisão após os dois pontos.

Neste caso, o primeiro nível (nó) é a falha disciplinar, em que para “sim” ela é podada e já se obtém a decisão (não falta) e para não, se ramifica em até mais 4 níveis. Outro exemplo de interpretação seria então: quando não há falha disciplinar e o custo de transporte é o mais alto, a falta é de um dia todo, enquanto para o custo mais baixo, ocorre apenas o atraso.

6.3. DISCUSSÃO DOS RESULTADOS

As observações são elencadas a seguir:

Os dois algoritmos geraram taxas de acerto muito próximas, no entanto, observando a matriz de confusão, os erros foram distintos;

Apesar da visualização ser muito diferente, a interpretação pode ser muito parecida, pois a partir da árvore se obtém a regra, e é isto que o PART faz;

O próprio PART era uma opção para compor o classificador utilizado pelo RSM, no entanto, para maior variedade dos resultados e melhor comparatividade foi optado por manter o REPTree;

Ainda assim alguns resultados foram iguais, como os exemplos citados, uma das descobertas mais relevantes reforçada por ambos os algoritmos: o funcionário que já teve falhas disciplinares tende a não faltar no trabalho;

No entanto, por não se resumir a regras pontuais com a árvore apresentada é possível observar outros fatores de influência (mais óbvios) como os custos mais altos de transporte acarretarem em mais horas faltadas, bem como o tempo de serviço. Já de fator que não aparentou exercer muita influência, indo contra o senso comum, é a distância do trabalho, já que muitas vezes quem está mais distante não têm mais ausências;

Por fim é interessante a combinação de métodos e técnicas de mineração para de obter mais informações relevantes ou reforçar àquelas que têm maior acurácia, sendo validadas por mais de um método. Inclusive, nesta base ainda poderiam haver aplicações de outras tarefas de mineração, como a associação, para analisar por outro ângulo os resultados, obtendo mais conhecimento.

7. CONSIDERAÇÕES FINAIS

O trabalho cumpriu o objetivo de descobrir conhecimento em bases de dados utilizando as técnicas de mineração, contribuindo com o entendimento do processo de mineração de dados e suas dificuldades.

Entre as maiores dificuldades, estão a falta de literatura e informações disponíveis na web, assim como desacordo entre diferentes fontes. Isso representou um grande desafio, mas o maior deles foi de fato a aplicação prática dos algoritmos. Se mostrou muito essencial ao processo ter domínio sobre as técnicas de mineração, os algoritmos, a matemática e a ciência da informação, para que se obtivessem os melhores resultados.

Quanto aos resultados, como requisito para a descoberta do conhecimento por mineração de dados, este deve ser não óbvio e relevante. Algumas informações descobertas se enquadram neste requisito. O exemplo da regra que indica que falhas disciplinares no passado, acarretam em não ausências do funcionário, por si não é óbvia e é relevante, exigindo uma abstração para se criar hipóteses que expliquem isso, e promover o entendimento do comportamento de funcionários.

Do conhecimento descoberto neste processo, pode-se haver diversos usos e decisões, desde contratações a desligamentos de funcionários com o objetivo de evitar um alto índice de absenteísmo até mudanças na estruturas de gestão de

pessoas na empresa evitando fatores que possam levar a maiores índices de ausências e assim aumentar a produtividade.

REFERÊNCIAS

AMO, Sandra De. Técnicas de mineração de dados. **Jornada de Atualização em Informática**, 2004. Disponível em: http://files.sistemas2012.webnode.com.br/200000095-bf367bfb43/Tecnicas%20de%20Minera%C3%A7%C3%A3o%20de%20Dados.pdf. Acesso em: 15 de nov. de 2019.

CAMILO, Cássio Oliveira; SILVA, João Carlos da. Mineração de dados: Conceitos, tarefas, métodos e ferramentas. **Universidade Federal de Goiás (UFC)**, p. 1-29, 2009. Disponível em: https://rozero.webcindario.com/disciplinas/fbmg/dm/RT-INF_001-09.pdf. Acesso em: 15 de nov. de 2019.

COSTA, Jefferson; BERNARDINI, Flávia Cristina; FILHO, José. A mineração de dados e a qualidade de conhecimentos extraídos dos boletins de ocorrência das rodovias federais brasileiras. **AtoZ: novas práticas em informação e conhecimento**, v. 3, n. 2, p. 139-157, 2014.

FELIX, Waldyr. Porque investir em Data Science em 2018. 2018. Disponível em: https://waldyrfelix.com.br/porque-investir-em-data-science-em-2018-307da7c69a4. Acesso em 14 de nov. de 2019.

FRANK, Eibe; WITTEN, Ian H. Generating accurate rule sets without global optimization. 1998.

GALVÃO, Noemi Dreyer; DE FÁTIMA MARIN, Heimar. Técnica de mineração de dados: uma revisão da literatura. **Acta Paulista de Enfermagem**, v. 22, n. 5, p. 686-690, 2009. Disponível em: https://www.redalyc.org/pdf/3070/307023846014.pdf. Acesso em: 15 de nov. de 2019.

HO, Tin Kam. The Random Subspace Method for Constructing Decision Forests. **IEEE Transactions On Pattern Analysis And Machine Intelligence**, v. 20, n. 8, p. 832-844, 1998.

KAMEL, Mohamed S.; DUIN, Robert PW. Random subspace method in text categorization. In: **2010 20th International Conference on Pattern Recognition**. IEEE, 2010. p. 2049-2052.

PIRES, José Ramon Trindade. Inteligência Empresarial em um Ambiente Acadêmico. 2011. Disponível em: http://www2.uesb.br/computacao/wp-content/uploads/2014/09/Intelig%C3%AAncia-Empresarial-em-um-Ambiente-Acad%C3%AAmico.pdf. Acesso em 16 de nov. de 2019.

STEINER, Maria Teresinha Arns *et al*. Abordagem de um problema médico por meio do processo de KDD com ênfase à análise exploratória dos dados. **Gest Prod**, v. 13, n. 2, p. 325-37, 2006. Disponível em: http://www.scielo.br/pdf/gp/v13n2/31177.pdf. Acesso em: 16 de nov. de 2019.