Data Quality Measurement Framework

Marcos Ferreira
Programa de Pós-Graduação em
Engenharia Elétrica e Computação
Universidade Presbiteriana Mackenzie
São Paulo, Brasil
Email: marcosferreira220863@hotmail.com

Leandro A. Silva
Faculdade de Computação e Informática &
Programa de Pós-Graduação em
Universidade Presbiteriana Mackenzie
São Paulo, Brasil
email leandroaugusto.silva@mackenzie.br

***Abstract*—Data Quality evaluation is a key fundamental in Knowledge Data Discovery projects. There are some project frameworks, like CRISP-DM and DAMA DMBOK, that recommend the preparation of the Data Quality Report, as a tool to describe the found problems during the data exploration phase and to describe an approach to fix those problems. However, those frameworks are very generic in their guidelines and neither tell what exactly should be measured nor how to associate any measure to the data quality. Data Profiling tools and some ETL(Extraction, Transformation and Loading) tools as well, implement some basic Statistical Description tooling, but they do not propose any general methodolgy to evaluate quantitatively the quality of a set of data, except, perhaps, in the IBM Watson Analytics tool. This article proposes a quantitative measure for data quality evaluation, based on Statistical Description tools.**

***Index Terms*—Data Quality, Data Profiling, Dat Mining, Data Governance, preprocessing**

I. Introdução

Qualidade de Dados é um tema relevante em projetos de Mineração de Dados. Dados ausentes, ruídos, dados desbalanceados ou anomalias (*outliers*) afetam negativamente o desempenho dos algoritmos e comprometem o resultado da análise [1], [4], [5].

Existem alguns modelos de Governança de Dados, como o *CRISP* DM e o *DAMA DMBOK*, que fornecem um referencial de trabalho para tratamento da qualidade de dados, indicando os processos e as melhores práticas a serem seguidas em projetos de tal natureza. Um aspecto indicado em ambos modelos é o relatório para garantir que os dados sejam armazenados com qualidade e que também é usado para fornecer um diagnóstico da qualidade dos dados analisados. Nesse relatório deve se descrever quais medidas serão adotadas para saneamento dos dados [2], [3].

Os modelos de governança acima mencionados apenas destacam a importância da avaliação da qualidade dos dados, mas não oferecem maior direcionamento para avaliação dessa medida. Como, por exemplo, quais medidas devem ser usadas nessa avaliação? Nesse sentido, a estatística descritiva fornece um conjunto de ferramentas simples, porém eficazes para aferição da qualidade dos dados [4], [5].

Ferramentas de ETL (do inglês,*extraction, transformation and loading*) e ferramentas de ajuste de dados (do inglês, *data profiling*) costumam implementar algumas análises estatísticas simples como histogramas, medidas de posição e dispersão, quantidade de valores ausentes e problemas de inconsistência, identificando valores incompatíveis com o domínio do atributo.

Naumann(2014) lista algumas tarefas simples realizadas por esse tipo de ferramenta e cita por exemplo as ferramentas da Microsoft e IBM [6].

Ferramentas como Talend e Rapid Miner também oferecem gráficos e histogramas. Já a ferramenta Metanome fornece a medida de entropia para cada atributo de dados [7].

O IBM Watson Analytics, por exemplo, avalia a quantidade de dados ausentes, desbalanceamento, assimetria (*skewness*), *outliers*, valores constantes ou semiconsoantes e o peso do atributo, dando uma nota para cada um deles, avaliando sua qualidade. Em seguida, calcula uma média geral composta da média das notas dos atributos, atribuindo-a ao conjunto todo. Essa nota geral dá uma indicação simples sobre a qualidade geral dos dados analisados. Com base nela o analista pode limpar a amostra e prosseguir com a análise. Não são claros, porém, os critérios usados para a atribuição daqueles pesos nas notas dos atributos e na nota geral [8].

O objetivo deste trabalho é usar as técnicas da estatística descritiva para medir a qualidade de dados e criar um referencial para mensuração da sua qualidade.

A proposta é a criação de um *framework* que implemente um módulo de qualidade de dados. Sua função é analisar os atributos de dados, procurando por ocorrência de dados faltantes, inconsistências e desbalanceamento. Outra funcionalidade é a de atribuir notas aos atributos, relacionadas com a sua qualidade. As médias das notas são aferidas. A nota final é obtida, verificando-se a ocorrência de instâncias repetidas de dados e *outliers* e fazendo os devidos descontos. O *framework* também aponta onde se encontram as ocorrências, auxiliando dessa forma, na elaboração do Relatório de Qualidade de Dados.

Para a realização deste trabalho será usada a ferramenta R [16]. O R é uma linguagem de programação e um ambiente de desenvolvimento de código aberto, mantido pela fundação R(©*The R Foundation*) [16]. Um dos grandes motivos para sua utilização é a sua facilidade de uso, seu conjunto de pacotes de funções e documentação abrangente, além do fato de ser um software livre que implementa muitas das técnicas usadas neste trabalho.

Também, apenas para comparação, usa-se a ferramenta *IBM Watson Analytics* [17]. O IBM Wason Analytics é uma

ferramenta abrangente de inteligência artificial. Uma de suas funcionalidades é a de permitir a carga de dados que serão analisados, dando uma nota de qualidade geral a eles. Trata-se de uma ferramenta proprietária, mas o uso para avaliação com funcionalidades limitadas e por tempo restrito é permitido.

As bases de dados utilizadas são todas públicas e disponíveis na base de dados da UCI. Para esse estudo, utilizou-se a base de dados *Mushroom.csv*, com 8124 observações sobre propriedades de cogumelos comestíveis e venenosos [15]. Foram feitas algumas modificações nessa base, injetando dados duplicados e valores de atributos fora do domínio, apenas para comparar com a nota obtida para os dados originais.

Além da introdução, este trabalho está organizado da seguinte maneira. A Seção II apresenta a fundamentação teórica necessária para entendimento do *framework* proposto. A Seção III, a metodologia utilizada.A Seção IV apresenta a análise e resultados. Finalmente, a Seção V apresenta a conclusão deste trabalho, indicando os trabalhos futuros.

II. Referencial Teórico

A lista de problemas com qualidade de dados, listados brevemente, a seguir, abrange alguns dos principais casos mencionados na literatura: inconsistências, anomalias (*outliers*) em atributos e nos objetos, incompletude (valores ausentes em atributos), assimetria de dados (*skweness* e curtose), dados repetidos, atributos correlacionados ou que pouco contribuem para a análise preditiva em tarefas de classificação [4].

Inconsistência é a ocorrência de valores fora do domínio de um atributo de um objeto de dados. Em geral, pode-se medir a inconsistência contando o número de valores não esperados para um determinado atributo e dividindo esse valor pelo número de objetos presentes na amostra. Para que isso possa ser feito é necessário conhecer previamente os valores permitidos para o atributo e o seu tipo [1].

Já a anomalia é um tipo de inconsistência, onde um determinado valor de atributo apresenta um valor muito discrepante em relação aos demais, embora ainda possa ser um valor válido dentro do domínio daquele atributo. Para dados uni variados numéricos discretos, uma possível medição para o grau de anomalia da amostra seria a contagem de valores que estão acima ou abaixo de um determinado limiar crítico. Como exemplo, poderia se usar o valor do atributo em relação ao primeiro e terceiro quartil [4]:

$$\varphi = \mathbf{Q1} \ - \ \mathbf{k} * (\mathbf{Q3} - \mathbf{Q1}) \tag{1}$$

$$\mathbf{\Phi} = \mathbf{Q3} + \mathbf{k} * (\mathbf{Q3} - \mathbf{Q1}) \tag{2}$$

Onde φ representa o limiar inferior e Φ, o superior. Costuma-se utilizar o valor k=1,5 em aplicações práticas. Valores abaixo do limiar inferior φ ou acima do limiar superior Φ representam *outliers*. Este método de detecção de anomalias tem a vantagem de não pressupor o tipo de distribuição dos dados [4].

Para atributos numéricos, pode-se usar a estatística de Grubbs [4]. Para cada valor do atributo x em um conjunto de dados, define-se o score-z como:

$$\mathbf{z} = \frac{|\mathbf{x} \ - \ \bar{\mathbf{x}}|}{\mathbf{s}} \tag{3}$$

onde $\bar{x}$ e s são a média e o desvio padrão da amostra de dados de tamanho N, respectivamente. O valor z é considerado anômalo se:

$$\mathbf{z} \ \geq \ \frac{\mathbf{N-1}}{\sqrt{\mathbf{N}}} * \sqrt{\frac{\mathbf{t^2_{\alpha/2,N-2}}}{\mathbf{N-2+t^2_{\alpha/2,N-2}}}} \tag{4}$$

onde $t^2_{\alpha/2,N-2}$ é o valor obtido da distribuição 't-Student', com o nível de significância $\alpha/2$ (teste bicaudal) e **N-2** graus de liberdade [1], [4].

O teste de Grubbs pressupõe que os dados sejam distribuídos normalmente. Já o método da distância interquartil na equação (1) não exige qualquer conhecimento prévio sobre a distribuição dos dados [1].

Ao se estudar dados multivariados, pode-se utilizar a distância de Mahanalobis para detectar anomalia de objetos [11].

A Distância de Mahalanobis é dada por:

$$\mathbf{r_i} = \sqrt{(\mathbf{x_i} - \bar{\mathbf{x}})^{\mathbf{t}} \mathbf{\Sigma^{-1}} (\mathbf{x_i} - \bar{\mathbf{x}})} \tag{5}$$

Onde:

- x_i é um objeto da base de dados (nxm), $i = 1..n$ objetos e cada objeto tem m atributos;
- $\bar{x}$ é o vetor médio da amostra, calculado da seguinte forma:

$$\bar{\mathbf{x}} = \frac{\mathbf{1}}{\mathbf{n}} \sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{x_i}; \tag{6}$$

- $(\mathbf{x_i} - \bar{\mathbf{x}})^{\mathbf{t}}$ é o transposto do vetor que subtrai o vetor $\mathbf{x_i}$ de sua média;
- $\mathbf{\Sigma^{-1}}$ é o inverso da matriz de covariância;
- $\mathbf{r_{(n \times 1)}}$ é o vetor coluna, com a distância de Mahalanobis para cada objeto, em relação ao objeto médio.

Pode-se aplicar o teste de Grubbs sobre os valores de r para determinar se o valor é um outlier ou não. Embora a estatística para o teste de Grubbs pressuponha que a distribuição seja aproximadamente normal, essa restrição pode ser relaxada em caso de um conjunto de dados grandes não desbalanceados [4].

Se os dados forem categóricos pode-se usar o algoritmo 1

$$\mathbf{AVFScore}(\mathbf{x_i}) = \frac{\mathbf{1}}{\mathbf{m}} \sum_{\mathbf{j=1}}^{\mathbf{m}} \mathbf{f}(\mathbf{x_{ij}}) \tag{7}$$

Onde:

- $\mathbf{x_i}$ é um objeto i da base de dados com m atributos;
- $\mathbf{f}(\mathbf{x_{ij}})$ é a frequência da ocorrência do valor do atributo $\mathbf{x_{ij}}$ do objeto i, com $j = 1..m$, onde m é o número de atributos.

O algoritmo 1 descreve os passos necessários para cálculo do *AVF Score* [13]:

Algorithm 1 AVFSscore

INPUT: D; {Matriz de dados nxm }
INPUT: k; {Número de Outliers }
OUTPUT: Outliers; {Vetor com os K outliers, identificando os objetos}
OUTPUT: AVFScore; {Vetor de Scores para cada objeto}
freq(i,j):=0; {Inicializa a matriz de frequência de dados}
AVFScore=NULL;{Vetor com os scores de cada objeto}
Outliers=NULL;{Vetor de retorno com k outliers}
for $i = 1$ TO N **do**
  for $j = 1$ TO m **do**
    f(i,j):=CalculaFrequenqcia($x_{i,j}$);{Calcula a matriz de frequencias f(i,j) para cada elemento x(i,j)}
  end for
end for
for $i = 1$ TO n **do**
  for $j = 1$ TO m **do**
    $AVFScore(x_i) += f(i,j)$;
  end for
  $AVFScore(x_i)/ = AVFScore(x_i)/M$ {Divide o valor de cada linha pelo número de atributos}
end for
Outliers:=Seleciona(AVFScore,k); {Seleciona os k menores AVFScore}
return Outliers, AVFScore;

Objetos que possuem um baixo valor de AVF Score são aqueles com menor frequência de ocorrência, e podem ser considerados outliers, por serem menos prováveis. Para determinar se o objeto é ou não um outlier, pode-se utilizar como critério de valor crítico [12], [13]:

$$\mathbf{AVFScore_{cr} = Media(AVFScore) - 3 * DP(AVFScore)} \tag{8}$$

onde a Média é calculada sobre os valores do vetor ***AVFScore*** e ***DesvPadr*** é o desvio padrão DP, também sobre os elementos de AVFScore. Deve-se assumir, nesse caso, que a distribuição dos valores do vetor AVFScore seja aproximadamente normal, que é uma hipótese razoável se o número de objetos for grande. Dessa forma, como se deseja os k objetos com os menores valores, define-se o 3 desvios padrões abaixo da média para determinação de $AVFScore_{critico}$.

Na categoria denominada Incompletude, encontra-se o problema de valores ausentes em determinados atributos. É fácil mensurar dados faltantes em atributos e ter uma ideia da percentagem destes sobre o número de objetos totais da base [4].

Uma distribuição pode ser assimétrica positivamente, indicando que possui uma 'cauda' prolongada à direita ou assimétrica negativa, indicando que possui uma 'cauda' prolongada em direção ao eixo dos 'y' ou à esquerda. A medida usada para assimetria (*skewness*) é [4]:

$$\gamma = \frac{\mathbf{E}(\mathbf{x} - \bar{\mathbf{x}})^{\mathbf{3}}}{\sigma^{\mathbf{3}}} \tag{9}$$

Onde $\mathbf{E}(\mathbf{x} - \bar{\mathbf{x}})^{\mathbf{3}}$ é o terceiro momento $\mu_{\mathbf{3}}$ da variável x e σ o desvio padrão.

Uma distribuição normal possui um valor de *skewness* igual a 0. A assimetria em relação aos atributos indica desvio em relação à distribuição normal e desbalanceamento. Essa informação é útil caso se esteja usando algum método de mineração de dados que pressuponha a distribuição normal nos valores dos atributos e pode ser aplicada a dados numéricos [4].

Embora menos usada na prática, a Curtose também é uma medida do afastamento da distribuição em relação à normal [9], [14]. Uma distribuição pode, por exemplo, ser simétrica, isto é, não apresentar *skeweness* e ainda assim desviar-se da curva normal. Por exemplo, a distribuição $t - Student$ é uma distribuição simétrica, mas apresenta 'caudas' mais grossas comparadas com a distribuição normal. A curva da distribuição $t-Student$ também é menos achatada do que a da distribuição normal. A medida de Curtose é dada por [9]:

$$\beta = \frac{\mathbf{E}(\mathbf{x} - \bar{\mathbf{x}})^{\mathbf{4}}}{\sigma^{\mathbf{4}}} - \mathbf{3} \tag{10}$$

Onde $\mathbf{E}(\mathbf{x} - \bar{\mathbf{x}})^{\mathbf{4}}$ é o quarto momento em relação à variável x e σ o desvio padrão.

A Curtose, juntamente com a medida de *skewness*, pode ser usada para medir o desvio da distribuição de um atributo em relação à distribuição normal. Curvas com o valor de $\beta \sim 0$ e $\gamma \sim 0$ indicam uma distribuição muito próxima da normal. Existem testes de aderência específicos para determinar se uma determinada distribuição segue uma distribuição normal, como o teste de Kolmogorov-Smirnov [9].

Dados repetidos ou duplicados também afetam a qualidade de dados. Algoritmos como k-Means, por exemplo, são afetados pelo valor médio dos dados. Se houver grande número de repetições para um determinado objeto, a média se desloca, provocando alterações no resultado [1].

O desbalanceamento de um atributo também pode ser medido pela sua entropia, que está associada à quantidade de informação que o atributo carrega ou ao seu grau de desordem [5].

A entropia de um atributo com k valores distintos, é definida como [1]:

$$\mathbf{S} = -\sum_{\mathbf{i=1}}^{\mathbf{k}} \frac{\mathbf{n_i}}{\mathbf{n}} * \log_{\mathbf{2}}(\frac{\mathbf{n_i}}{\mathbf{n}}) \tag{11}$$

Onde n_i é o número de ocorrências de um determinado valor do atributo e n o número total de elementos.

Se um determinado atributo A_j possuir apenas um único valor, a sua entropia é igual a zero, uma vez que $p(A_k) = \frac{n_k}{n} = 1$.

Se o mesmo atributo possuir n valores distintos, igualmente distribuídos, de modo que $p(A_k) = \frac{1}{n}$ e se $n \to \infty$, a entropia tende a 0, uma vez que [18]:

$$S = -\sum_{i=1}^{k} p(x_i \log_2(p(x_i))) = \quad (12)$$

$$S = -\sum_{i}^{n} \frac{1}{n} \log_2(\frac{1}{n}) \quad (13)$$

$$(14)$$

e,

$$\lim_{n\to\infty} -\sum_{i}^{n} \frac{1}{n} \log_2(\frac{1}{n}) = \quad (15)$$

$$-\sum_{i}^{n} \lim_{n\to\infty} (\frac{1}{n} \log_2(\frac{1}{n})) \quad (16)$$

Aproximando n por uma variável do tipo contínua e aplicando a regra de L´Hospital, então a equação (16) tende a 0 quando $n \to \infty$, já que cada termo dentro da soma tende a zero.

Se houver apenas dois valores distintos, igualmente distribuídos, de modo que $\frac{n_1}{n} = 0.5$ e $\frac{n_2}{n} = 0.5$ com $n_1+n_2 = n$, então o valor da entropia é 1.

Valores de entropia muito próximos de zero podem indicar que o atributo ou possui valores quase constantes ou possui valores igualmente distribuídos com baixa probabilidade de ocorrência.

Em atributos categóricos poder-se-ia utilizar um histograma de frequências, mas esse método, por ser visual, não permite a mensuração do grau de desbalanceamento do atributo. Por outro lado, valores altos de entropia indicam que os valores do atributo estão igualmente distribuídos e a probabilidade de ocorrência de um valor particular é baixa [4], [18].

Alguns algoritmos de classificação, baseados em árvores de decisão como ID3, C4.5(C5.0) usam a entropia para a tarefa de predição. Quando o atributo é totalmente balanceado, isto é, todos os valores possuem igual distribuição, conforme foi visto, a entropia daquele atributo atinge o seu valor máximo. Por outro lado, se o atributo for desbalanceado, sua entropia tende a zero. Isso faz com que esses algoritmos apresentem altas taxas de falsos negativos se estes não forem robustos o suficiente para tratar conjunto de dados desbalanceados [18].

Um último problema que afeta a qualidade dos dados -e, de certa forma também ligado à entropia - é a correlação entre atributos. Atributos de conjunto de dados não normalizados podem ser combinações de outros atributos, como soma, multiplicação ou outra transformação.

Um dos métodos para se detectar a correlação dos atributos e a importância destes no modelo é método intitulado Incerteza Simétrica (*Simmetrical Uncertainty*), medida definida como [10]:

$$\mathbf{SU(X,Y)} = \mathbf{2}[\frac{\mathbf{IG(X|Y)}}{\mathbf{H(X)} + \mathbf{H(Y)}}] \quad (17)$$

onde $IG(X|Y)$ é o *Ganho de Informação*, que mede a diminuição da entropia do atributo X quando ocorre a informação adicional proporcionada pelo atributo Y. O *Ganho de Informção* é definido como [10]:

$$\mathbf{IG(X|Y)} = \mathbf{H(X)} - \mathbf{H(X|Y)} \quad (18)$$

Onde:

$$\mathbf{H(X)} = -\sum_{\mathbf{i=1}}^{\mathbf{l}} \mathbf{p(X_i)} * \mathbf{\log_2(p(X_i))} \quad (19)$$

é a entropia do atributo X, que pode assumir l valores distintos.

De forma semelhante H(Y) é a entropia do atributo Y

$$\mathbf{H(Y)} = -\sum_{\mathbf{i=1}}^{\mathbf{k}} \mathbf{p(Y_i)} * \mathbf{\log_2(p(Y_i))} \quad (20)$$

onde o atributo Y_j assume k valores distintos. A probabilidade $p(Y_i)$ é a probabilidade de ocorrência do atributo Y_i.

Finalmente,

$$\mathbf{H(X|Y)} = -\sum_{\mathbf{j=1}}^{\mathbf{l}} \mathbf{p(Y_j)} \sum_{\mathbf{i=1}}^{\mathbf{k}} \mathbf{p(X_i|Y_j)} * \mathbf{\log_2(p(X_i|Y_j))} \quad (21)$$

$H(X|Y)$ é a entropia condicional do atributo X quando o atributo Y ocorreu. $p(X_i|Y_j))$ é probabilidade condicional do atributo X quando o atributo Y ocorreu [10].

Um atributo Y é considerado mais correlacionado ao atributo X do que um atributo Z se $SU(Y,X) > SU(Z,X)$. A medida $SU(X,Y)$ pode ser usada para selecionar atributos mais influentes para o modelo. Atributos com baixo valor de $SU(Y,X)$ podem ser filtrados do modelo, sem que este perca muita informação [10].

A incerteza simétrica pode ser usada para seleção de atributos mais relevantes para o modelo de predição; é uma alternativa ao método de análise de componentes principais. A vantagem da incerteza simétrica é que pode ser aplicado tanto sobre atributos categóricos como atributos numéricos. Essa medida é usada no Algoritmo $FCBF$ [10].

Outros métodos de teste de correlação entre variáveis são as medidas de co-variância e o método do Chi-Quadrado [1]. Estes só podem ser usados em atributos numéricos.

A tabela I resume os principais problemas que podem ser tratados com a qualidade de dados e como eles impactam atributos numéricos, categóricos e as principais tarefas de Mineração de Dados.

Na próxima seção discute-se a metodologia geral deste trabalho e como as medidas estatísticas discutidas acima podem ser usadas para implementar um referencial de medição de qualidade de dados.

III. METODOLOGIA

A figura 1 esboça, em alto nível, uma arquitetura para o framework de avaliação de qualidade de dados constituída de 3 componentes principais. Esses módulos ainda estão em fase de desenvolvimento e estão sendo implementados com uso de scripts em linguagem **R**. O Módulo Identificador de Problemas com o Dataset é responsável por identificar os problemas

Table I
RESUMO DOS PROBLEMAS COM QUALIDADE DE DADOS

Qualidade	Numéricos::Medição	Categóricos::Medição	Impacto nos Algoritmos
Inconsistência	Proporção de Instâncias Inconsistentes	Proporção de Instâncias Inconsistentes	Algoritmos de Classificação ou agrupamento não param, mas podem apresentar resultados errados
Outliers	Proporção de Outliers	Não se Aplica	Algoritmos de agrupamento como k-Means são sensíveis ao desbalanceamento dos atributos
Ausência de Dados	Proporção de Valores Ausentes	Proporção de Valores Ausentes	Algoritmos de Classificação e Agrupamento não rodam se houver valores ausentes
Desbalanceamento	*Skewness*	Entropia	Algoritmos de classificação baseados em entropia - árvores de decisão - são afetados negativamente com o desbalanceamento de dados
Instâncias de Dados Repetidos	Proporção de dados Repetidos na amostra	Proporção de Dados Repetidos na Amostra	Afetam conjunto de dados com atributos numéricos e categóricos. Duplicidade de dados afeta algoritmos como k-Means
Atributos Fortemente Correlacionados entre si ou fracamente correlacionados com o atributo de classe	Se a incerteza simétrica do atributo for muito baixa, esse atributo contribui pouco para o modelo em tarefas de classificação e pode ser removido do modelo	Se a entropia de um atributo categórico for zero ou próximo de zero, significa que os valores estão provavelmente desbalanceados. Esse atributo pode ser removido do modelo	Modelos de dados com grande número de atributos podem apresentar forte correlação entre os atributos, que acabam se tornando redundantes no modelo. Por outro lado, atributos fracamente correlacionados com o atributo de classe podem ser removidos do modelo. Esse problema afeta principalmente tarefas de classificação e regressão

discutidos na seção anterior para os atributos de dados. Os problemas investigados podem depender do tipo de atributo: alguns problemas são comuns tanto para atributos numéricos quanto para atributos categóricos, como por exemplo, ausência de valores. Outros problemas são específicos para o tipo de atributo investigado.

O Módulo Gerenciador de Medidas é usado para a seleção do tpo de tarefa de mineração que será usada. Algumas tarefas são mais afetadas pelo tipo de problema encontrado do que outras. Por exemplo, em árvores de decisão, o desbalanceamento de dados afeta negativamente o resultado do algoritmo; o algoritmo k-Means,de forma semelhante, é afetado pela presença de outliers na amostra de dados. Esse módulo atribui pesos maiores aos problemas de dados encontrados, dependendo do tipo de tarefa selecionada.

O módulo de qualidade de dados é aquele responsável por computar as notas dos atributos e dar a nota geral para o conjunto de dados.

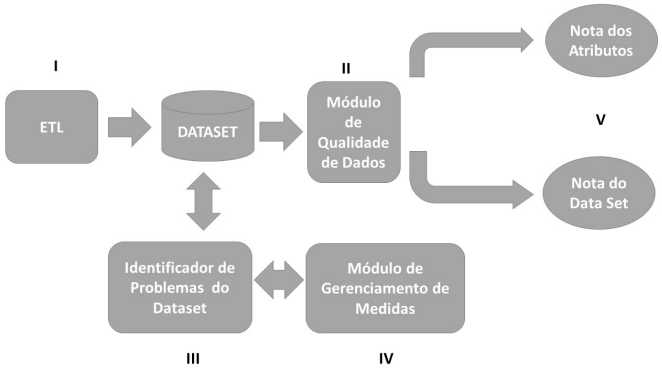

Figure 1. Framework para Medição de Qualidade de Dados

- I-Trata e Carrega os dados;
- II- Armazena e equação das notas, por atributo e a ponderação. Neste módulo, define-se as medidas feitas para diferentes tipos de dados e tarefas de mineração;
- III - Gerencia os tipos de problemas e o tipo de medida que será adotado na análise;
- IV - Armazena as medidas de Qualidade;
- V - Saída: atribuição de notas para os atributos e para a base.

Para o experimento do framework, será usada uma versão modificada da base de dados *mushroom*, obtida no projeto da UCI [15]. Para a análise, será usado o pacote estatístico R (v.3.4.3 x86 64) como ferramenta para a aplicação da valoração da qualidade de dados e análises estatísticas do referencial proposto [16].

A escolha do R foi motivada pelo fato de ser esta uma ferramenta de código aberto, com vasta documentação e aceitação na comunidade de dados. A ferramenta implementa vários pacotes estatísticos e de aprendizagem de máquina, úteis em nossa análise [4], [5].

A base *mushroom* é usada em tarefas de classificação, como árvores de decisão. Originalmente, possui 8124 objetos com 22 atributos e um atributo de rótulo- category -, que indica se o cogumelo é venenoso (p-*poisonous*) ou comestível (e-*edible*). Um dos atributos -*veil type*- possui um único valor, sendo que sua entropia é igual a zero. A nota de qualidade deste atributo, em particular, é zero, assim como sua incerteza simétrica [15]. A base original não possui dados duplicados.

Todos os atributos da base *mushroom* são categóricos. Existem valores ausentes, indicados com o caractere '?' na base original. Na base modificada para este estudo, foram inseridas algumas linhas duplicadas e dados fora do domínio em alguns atributos, *'sujando'* a base. A base final modificada para testes possui 8129 objetos. Inicialmente, verifica-se a frequência de valores ausentes, *outliers*, isto é, valores fora do domínio do atributo e valores constantes. Nessa análise, não se avalia o valor do *skweness* e nem a curtose que dão uma indicação do desbalanceamento dos dados.

O algoritmo 2 resume o método de investigação.

Algorithm 2 Algoritmo de Aferição de Notas para Qualidade de Dados

1: INPUT: D; {Matriz de dados nxm}
2: OUTPUT Nota; {Vetor mx1 com a nota de cada atributo}
3: **for** $i = 1$ TO m **do**
4: $Nota_i := 100\%$ {Itera sobre os m atributos; inicializa nota com 100%}
5: S:=Entropy(A_i)
6: **if** $S == 0$ **then**
7: $Nota(i) := 0$;
8: Continue; {Vai para o próximo atributo}
9: **end if**
10: ValAusentes:=CalculaValAusentes(A_i);
11: **if** $valAusentes \neq 0$ **then**
12: $Nota_i := Nota_i - \frac{valAusentes}{n}$;
13: **end if**
14: $ValDiscrep := CalculaValDiscrepante(A_i)$
15: **if** $ValDiscrep \neq 0$ **then**
16: $Nota_i := Nota_i - \frac{ValDiscrep}{n}$;
17: **end if**
18: **if** $isNumeric(A_i)$ **then**
19: $valOutlier := CalculaOutliers(A_i$;
20: $Nota_i := Nota_i - \frac{valOutlier}{n}$ {Calcula núm. de Outliers }
21: **end if**
22: **end for**
23: **return** $(Nota)$ {Retorna o vetor de notas dos atributos}

Para contagem de valores discrepantes, deve-se analisar o domínio dos atributos e os valores permitidos para cada um. Essa informação faz parte da definição do problema e é fornecida a priori. Pode-se usar o algoritmo 3 para detecção de outliers, armazenando a quantidade por atributo em um vetor, para posterior uso na atribuição das notas

Algorithm 3 Contagem de Valores Discrepantes por atributo

1: INPUT: D(nxm); {Matriz de Dados nxm}
2: INPUT: ListaValPerm; {Lista com Valores Permtidos de cada atributo}
3: OUTPUT: VetElementosDiscrep(mx2); {Matriz r com os elementos discrepantes} {m: numero de atrbiutos;a coluna 1 indica o atributo e a col 2 o número de violações}
4: **for** $i = 1$ TO m **do**
5: $C := ContarViolacoes(D[, i])$; {Conta violações por atributo}
6: $VetElementosDiscrep.add(i, C)$;
7: **end for**
8: **return** $(VetElementosDiscrep)$

Para contagem de atributos com valores constantes usa-se o cálculo da entropia, usando a função *entropy*, do pacote do mesmo nome do R. O algoritmo 4 pode ser usado para o cálculo da entropia.

Algorithm 4 Cálculo de Entropia, por Atributo

1: INPUT: D(nxm); {Matriz de Dados nxm}
2: OUTPUT: EntropyAttr(mx2); {Matriz de entropia por atributo}
3: **for** $i = 1$ TO m **do**
4: $s := Entropy(D[, i])$;
5: $EntropyAttr.add(i, s)$;
6: **end for**
7: **return** $(EntropyAttr)$

A descrição da base Mushroons da UCI [15], indicava explicitamente que os valores ausentes podiam ser identificados com o símbolo $'?'$. Mas para outras bases, talvez seja necessário um trabalho de pré-processamento para identificar esse tipo de atributo. Por exemplo,no R, a função $isNa(valordoatributo)$ retorna TRUE se o valor do atributo estiver preenchido com NA, indicando tratar-se de um valor ausente. As ferramentas de ETL, por outro lado, podem interpretar o valor 'NA' como um valor válido, a menos que configuradas para considerar tais valores como ausentes. Para contagem dos valores ausentes, pode-se usar o algoritmo 5:

Algorithm 5 Cálculo de Valores Ausentes

1: INPUT: D(nxm); {Matriz de Dados nxm}
2: OUTPUT: Ausentes(mx2); {Matriz de valores ausentes}
3: **for** $i = 1$ TO m **do**
4: $s := ContaAusentes(D[, i])$;
5: $Ausentes.add(i, s)$;
6: **end for**
7: **return** $(Ausentes)$

Após avaliar a nota de cada atributo, usa-se a média das notas para obter uma nota da qualidade dos dados. Em seguida,

essa nota intermediária vai ser subtraída da percentagem de valores duplicados e da percentagem de *outliers*. A nota da qualidade de dados então fica:

$$Nfinal = \frac{1}{m}\sum_{i=1}^{m} NotaAttr_i - \%Outliers - \%Duplicados \quad (22)$$

Onde m é a quantidade de atributos da amostra de dados.

Nesta análise, não se está levando em conta a Incerteza Simétrica, os valores de *Skewness* e a Curtose. A entropia só é usada para determinar se um atributo possui valor constante. Atributos com valor constante possuem nota de qualidade igual a zero.

IV. Resultados e Análise

A tabela II resume os resultados aplicados à base de dados mushroom.csv, modificada, incluindo *outliers* e valores ausentes.

Table II
Resumo: Base de Dados Mushroom

Núm.Obj	8129	Núm. Atrib:	23

Avalia-se em seguida as notas para cada atributo, não se levando em conta a influência da Incerteza Simétrica na qualidade dos atributos. Para cada ocorrência - dados ausentes, *outliers*- subtraí-se a percentagem da nota do atributo, que começa com uma nota máxima de 100%. O resumo da análise encontra-se na tabela III.

Table III
Resultados com a base de dados Mushroom

Attr	Entrop	SU	Ausentes %	Dicrep %	Nota %
Category	0.999	Na	0.000	0.000	100
Cap Shape	1.654	0.036	0.010	0.010	99.98
Gill Attach.	0.175	0.123	0.000	0.000	100
Gill Color	2.510	0.207	0	0	100
stalk.surface.above.ring	1.220	0.256	1.69	0.000	98.31
stalk.color.below.ring	1.398	0.162	0.000	0.000	100
ring.number	0.420	0.005	0.000	0.000	100
Population	2.000	0.134	0.000	0.000	100
Cap.color	2.051	0.021	0.000	0.000	100
Bruises	0.979	0.194	0.000	0.000	100
Gill.spacing	0.637	0.024	0.000	0.000	100
Stalk.shape	0.987	0.001	0.000	0.000	100
Stalk.surface.below.ring	1.398	0.0226	0.00	0.000	100
Veil.type	0.000	0.000	0.000	0.000	**0.00**
Ring.type	1.543	0.251	0.000	0.06	99.94
Habitat	2.285	0.095	0.000	0.09	99.1
Cap.surface	1.577	0.022	0.000	0.001	99.99
Odor	2.319	0.546	0.000	0.000	100
Gill.size	0.892	0.243	0.000	0.000	100
Stalk.root	1.822	0.095	30.5	0.000	69.5
Stalk.color.above.ring	1.975	0.171	0.000	0.45	99.55
Veil.color	0.199	0.041	0.000	0.000	100
Spore.print.color	2.205	0.300	0.000	0.000	100
xxx	xxx	xxx	xxx	Média Atributos:	94.14

Somente para comparação, indica-se o valor da entropia de cada atributo e o valor da Incerteza Simétrica do atributo em relação ao atributo da classe. Note que a variável *'Veil.type'* possui entropia igual a 0, assim como sua incerteza simétrica. Isso se deve ao fato de que esta variável apresenta apenas um valor.

A tabela IV, mostra o resultado da nota final, usando o valor da média da nota dos atributos, subtraído da percentagem de *Outliers* e de dados duplicados.

Table IV
Resultados Finais: base de dados mushroom

Dados Duplicados %	Outliers %	Nota Final %
0.01	10	84.13

Com o uso da ferramenta R, foi possível detectar alguns problemas, como duplicidade, *outliers* e inconsistências nos atributos, de modo que a frequência dessas ocorrências foi usada para o cálculo da nota de qualidade. Uma nota alta atribuída à qualidade dos dados não significa que os problemas encontrados não devam ser tratados. Dependendo do tipo de tarefa de mineração executada, os problemas podem ter grande impacto. Por exemplo, em tarefas de classificação, valores ausentes influenciam negativamente os resultados. Este fato pode indicar que o peso da nota tem de ser maior para 'atributos ausentes'. Se a tarefa for de agrupamento, por exemplo, usando o algoritmo K-means, o peso dos *outliers* assume uma importância maior do que o peso das outras notas, assim como o peso dos dados duplicados.

O teste usou uma base de dados com atributos exclusivamente categóricos. Se fosse usado uma outra base com atributos numéricos, poder-se-ia ter usado a mesma metodologia, mas seria necessário incluir também na investigação os critérios de desbalanceamento como *skweness*. No caso da detecção de *outliers*, o score AVFScore teria de ser substituído pela distância de *Mahalanobis*, por exemplo.

Para comparação do método, foram usados os resultados obtidos pelo uso da ferramenta IBM Watson Analytics contra base de dados *Mushroom* modificada. O resultado está sintetizado na tabela V:

A ferramenta da IBM não forneceu os valores de entropia ou da Incerteza Simétrica, apenas as notas finais. Para indicação de valores ausentes, os dados tiveram de ser modificados, substituindo o caracter '?' por um espaço em branco, pois a ferramenta da IBM interpreta '?' como um valor válido de atributo.

Os resultados obtidos são muito discrepantes. Enquanto o método proposto neste trabalho atribuiu uma nota de 84% para qualidade de dados, o método da IBM atribuiu uma nota mais baixa: 64%. Conforme já mencionado, não são claros os critérios usados pela ferramenta da IBM para atribuição das notas. Por exemplo, a variável 'odor' recebeu uma nota de 61% e, no método proposto, recebeu uma nota 100%, pois não apresentou dados ausentes ou discrepantes. Além disso, foi a variável que apresentou o maior valor de Incerteza Simétrica, indicando ser uma das mais influentes no modelo.

Já as variáveis *Stalk Shape* e *Bruises* apresentaram entropia quase próxima de 1. Na metodologia da IBM, essas variáveis apresentaram notas, respectivamente, 88% e 86%. As respectivas entropias foram 0.98 e 0.97 (a entropia foi calcualda fora da ferramenta da IBM).

Fazendo-se uma análise de regressão simples, a nota atribuída pelo IBM Watson está correlacionada com a entropia

Table V
COMPARAÇÃO COM OS RESULTADOS OBTIDOS COM O IBM WATSON

Atributo	Nota
Category	97
Cap.shape	62
Gill.attachment	52
Gill.collor	76
stalk.surface.above.ring	61
stalk.color.below.ring	60
ring.number	52
Population	64
Cap.color	71
Bruises	86
Gill.spacing	63
Stalk.shape	88
Stalk.surface.below.ring	63
Veil.type	0
Ring.type	59
Habitat	61
Cap.surface	72
Odor	61
Gill.size	75
Stalk.root	69
Stalk.color.above.ring	59
Veil.color	51
Spore.print.color	71
Nota Total	64

da seguinte forma:

$$\mathbf{Nota} = \mathbf{38.71} * \mathbf{Entropia} \tag{23}$$

Os coeficientes de correlação R^2 e R são, respectivamente, 0.89 e 0.80, indicando alto grau de correlação entre a nota e a entropia, para um intervalo de confiança de 95%. No método proposto, como se levou em conta apenas dados ausentes, constantes ou discrepantes, as notas foram 100% para ambos atributos. Não fica claro qual a relação entre a nota atribuída pela ferramenta da IBM e a entropia.

Quando se analisa os atributos *Stalk Shape* e *Bruises*, considerando-se apenas a Incerteza simétrica, verifica-se que estes valores são baixos, comparados com os outros atributos. Por exemplo, para *Stalk.Shape* e *Bruises*, a incerteza simétrica é 0.007 e 0.19, respectivamente. No entanto, a nota atribuída ao atributo *Stalk.shape* é maior do que a nota atribuída ao atributo Bruises, pela metodologia da IBM. Na metodologia proposta pelo IBM Watson, a entropia parece ter um peso maior do que a incerteza simétrica na atribuição das notas de qualidade.

A inclusão dos critérios de Entropia e Incerteza Simétrica e como ela afetam a qualidade de dados, fica para um trabalho futuro de investigação. Outro aspecto que será incluído em trabalhos futuros é a ponderação das notas com os valores de *skweness* e da curtose em atributos numéricos. É necessária uma análise um pouco mais criteriosa para levar em conta essas duas propriedades na aferição da qualidade de dados.

V. CONCLUSÕES

Este trabalho apresentou um *framework* para medição e quantificação da qualidade de dados, propondo a implementação de um módulo de avaliação que se baseia em ferramentas de estatística descritiva. O objetivo é usá-lo nas subtarefas de avaliação de qualidade de dados propostos nos modelos de governança, como CRISP-DM ou DAMA DMBOK.

Para avaliação do *framework* proposto, utilizou-se o conjunto de dados da UCI, modificando-os para analisar alguns problemas encontrados em bases de dados. A Qualidade de Dados de cada base foi avaliada com base nos critérios enumerados no *framework*, atribuindo-se uma nota de qualidade final.

Por limitação de escopo e espaço, analisou-se apenas uma amostra com dados puramente categóricos. A incerteza simétrica também pode ser usada como medida de influência de atributos numéricos, mas também poderia se usar outra técnica como *'Principal Component Analysis'* ou outras técnicas de seleção de atributos para avaliar a importância dos mesmos e atribuir uma nota à qualidade. Isso também fica para um trabalho futuro.

Reconhece-se que a metodologia proposta nesse trabalho ainda encontra-se em um estágio preliminar. Faltou analisar como a melhoria da qualidade de dados afeta o desempenho dos algoritmos. No entanto, tentou-se mostrar aqui a importância desse assunto para trabalhos futuros em qualidade de dados e *data profiling*, chamando a atenção para a lacuna que existe sobre o tema, hoje, na literatura.

AGRADECIMENTOS

Gostaria de agradecer ao MackPesquisa e à Universidade Presbiteriana Mackenzie, juntamente com a CAPES, pelo fomento concedido para a realização desta pesquisa.

REFERENCES

[1] J.Han, J. Pei, and M. Kamber. Data Mining: Concepts and Techniques. Elsevier, 2011.
[2] P.Chapman, J.Clinton,R.Kerber,T.Kabhaza et all.CRISP-DM-1.0 Step by Step Data Mining Guide(2000).
[3] M. Brackett, P.S. Earley. The DAMA Guide to the Data Management Body of Knowledge -The Dama-DMBOK guide-2009.
[4] L.N. Castro, D.Ferrari. Introdução à Mineração de Dados:Conceitos Básicos,Algoritmos e Aplicações.São Paulo. Saraiva.2016.
[5] L.A. Silva, S.M. Peres, C. Boscariolli. Introdução à Mineração de Dados:Com Aplicações em R. Elsevier, Brasil, 2017.
[6] F.Naumman.Data profiling Revisited. ACM SIGMOD Record, v. 42, n. 4, p. 40-49, 2014.
[7] T.Papenbrock, T. Bergman et all. Data Profiling with Metanome. Proceedings of the VLDB Endowment, v. 8, n. 12, p. 1860-1863, 2015.
[8] M.Stacker.'Quality In, Quality Out',2015. [Online].Available:https://www.ibm.com/communities/analytics/watson-analytics-blog/quality-in-quality-out/.[Accessed:01-Apr-2018].
[9] W.D. Bussab, P.Morettin. Estatística Básica, 8ed, São Paulo, Saraiva, 2013
[10] L.Yu, H.Liu.Feature Selection for High Dimensional Data: A Fast Correlation-based filter solution. In: Proceedings of the 20th International Conference on Machine Learning(ICML-03)[S.l. s.n.]p:856-863,2003.
[11] J.F.Hair Jr, W.C.Black, B.J.Babin, R.E.Anderson-Multivariate Data Analysis, 7th Ed. Upper Saddle River, NJ, Prentice-Hall,1998
[12] D.L.S Reddy,B.R.Babu,A.Govardhan.Outlier Analysis of Categorical Data using NAVF.Informatica Economica, v17, no 1/2013, p5. 2013.

[13] M.R.P Jakkulwar, R.A. Fadanavis.Analysis of Outlier Detection in Categorical Dataset
[14] L.T. DeCarlo. On the Meaning and Use of Kurtosis.Psychological methods, v. 2, n. 3, p. 292, 1997.
[15] D. Dua and E.K. Taniskidou (2017).Donator J. Schlimmer. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.
[16] R Core Team (2017). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.[Acessed:1st-Apr-2018]
[17] IBM Corp. 'IBM Watson Analytics',2017,[Online]. Available:https://ibm.com/analytics.[Accessed:1-Apr-2018]
[18] A. Kirshners,S.Parshutin and H.Gorskis. "Entropy-Based Classifier Enhancement to Handle Imbalanced Class Problem." Procedia Computer Science 104. p: 586-591, 2017.