UNIVERSIDADE ESTADUAL DE PONTA GROSSA
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

FABIO DOS SANTOS

ALGORITMO kNN NA IMPUTAÇÃO DE DADOS DE ESPECTROS DE MASSA DO TIPO MALDI-TOF

Uma análise da influência da imputação com kNN sobre desempenho de classificadores logísticos para identificação de bactérias

PONTA GROSSA

2018

FABIO DOS SANTOS

ALGORITMO kNN NA IMPUTAÇÃO DE DADOS DE ESPECTROS DE MASSA DO TIPO MALDI-TOF

Uma análise da influência da imputação com kNN sobre desempenho de classificadores logísticos para identificação de bactérias

Dissertação submetida ao Programa de Pós-Graduação em Computação Aplicada, curso de Mestrado em Computação Aplicada - Área de concentração Computação para Tecnologias em Agricultura - da Universidade Estadual de Ponta Grossa, como requisito parcial para obtenção do título de Mestre.

Orientador: Prof. Dr. José Carlos Ferreira da Rocha

PONTA GROSSA

2018

Santos, Fabio dos
S237 Algoritmo kNN na imputação de dados de espectros de massa do tipo MALDI-TOF/ Fabio dos Santos. Ponta Grossa, 2018.
82 f.;

Dissertação (Mestrado Computação Aplicada - Universidade Estadual de Ponta Grossa.

Orientador: Prof. Dr. José Carlos Ferreira da Rocha

1. Imputação com kNN. 2. Espectrometria de massa. 3. Regressão logística. 4. Classificação de bactérias. I. Rocha, José Carlos Ferreira da. II. Universidade Estadual de Ponta Grossa. Mestrado em Computação Aplicada. III. T.

CDD : 004.3

Ficha catalográfica elaborada por Maria Luzia F. Bertholino dos Santos– CRB9/986

TERMO DE APROVAÇÃO

Fábio dos Santos

"ALGORITMO KNN NA IMPUTAÇÃO DE DADOS DE ESPECTROS DE MASSA DO TIPO MALDI-TOF: UMA ANÁLISE DA INFLUÊNCIA DA IMPUTAÇÃO COM KNN SOBRE O DESEMPENHO DE CLASSIFICADORES LOGÍSTICOS PARA IDENTIFICAÇÃO DE BACTÉRIAS"

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre no Programa de Pós-Graduação em Computação Aplicada da Universidade Estadual de Ponta Grossa, pela seguinte banca examinadora:

Prof. Dr. José Carlos Ferreira da Rocha
UEPG

Prof. Dr. André Pinz Borges
UTFPR

Profª Drª Carolina Weigert Galvão
UEPG

Ponta Grossa, 14 de setembro de 2018.

Dedico este trabalho aos meus pais Edite e Altamir, que sempre acreditaram em mim e em meu potencial mesmo quando eu duvidei de minha capacidade. A minha namorada Janaína que sempre me apoiou e incentivou a seguir sempre em frente.

AGRADECIMENTOS

Agradeço a Deus por ser sempre minha força em todos os momentos mais difíceis, guiando-me e por ser fonte de um Amor incondicional.

Aos meus pais, Edite e Altamir por sempre me apoiarem em todas as minhas decisões sejam acadêmicas ou profissionais, valorizando meus esforços em busca dos meus objetivos e toda apoio incondicional.

A minha namorada, Janaína que esteve comigo durante toda essa etapa, me apoiou e incentivou a continuar em busca deste sonho.

Ao meu orientador Professor Doutor José Carlos, que me aceitou como seu orientando, pela sua ajuda incondicional em todos os momentos deste trabalho, acreditando em mim quando eu não mais acreditava.

Aos professores Rafael e Carol, que sempre estiveram dispostos a me auxiliar durante o desenvolvimento desta dissertação, e em outros trabalhos durante esses mais de dois anos de mestrado.

Aos demais professores, que sempre estiveram prontos a me auxiliar durante as aulas e no desenvolvimento do espírito e mentalidade de pesquisador durante as aulas e com conselhos.

A Universidade Estadual de Ponta Grossa, e ao programa de pós-graduação de Computação Aplicada por me aceitarem como discente, me dando toda capacidade de realizar este trabalho.

A Capes pela bolsa de estudos concedida.

Ao grande parça Henrique Silveira, pelas diversas horas de conversas.

Aos meus amigos, com os quais convivi por muitas horas de laboratório e sala de aula, realizando diversos trabalhos e compartilhando diversos momentos durante esses mais de dois anos, Giancarlo, Alessandro, David, Douglas, Renann (Calheiros), Rodrigo (Digão), Luiz e Alisson.

RESUMO

O processo de identificação de bactérias relacionadas ao crescimento vegetal, é alvo de diversos estudos na área de bioinformática. Uma das formas para realizar esta identificação é utilizar dados de espectrometria de massa do tipo MALDI-TOF para detectar a presença de proteínas ribossomais em uma amostra, e então, usar classificadores para processar estes dados e selecionar o rótulo com a maior probabilidade. Durante o processo de geração dos espectros de massa para classificação é comum a não detecção de algum dos picos relacionados a proteínas ribossomais. Considerando isto, este trabalho apresenta um estudo sobre o uso do algoritmo kNN para imputação desses casos. O estudo foi desenvolvido com o uso de classificadores logísticos para identificação de bactérias da espécie *Staphylococcus aureus* e do gênero *Bacillus*. Durante os experimentos foram testados três técnicas para imputar dados: imputação com zero, imputação com a média do atributo faltante, e a imputação com kNN. Desta última foram usadas duas abordagens: função de agregação de média e função de agregação de mediana. O protocolo experimental implementado possibilitou avaliar a influência da imputação sobre os resultados de classificação sob diferentes cenários no que se refere ao número de variáveis faltantes. Os resultados obtidos mostram que o emprego do kNN não levou à uma redução do desempenho dos classificadores, em relação àquele observado quando do uso de dados completos. Além disto, a classificação de dados submetidos a imputação pelo kNN apresentou desempenho superior àquele verificado quando do uso dos demais métodos.

Palavras-chave: Imputação com kNN, Espectrometria de Massa, Regressão Logística, Classificação de Bactérias

ABSTRACT

It is subject of several studies in bioinformatics area the plant growth promoting bacteria identification process. An approach to performing it is to process sample's ribosomal proteins data obtained by MALDI-TOF mass spectrometry through a classifier and select the highest probability label. However, at the time of mass spectra generation, it is common not detecting some ribosomal proteins related peaks data. With this in mind, this work presents a study about data imputation through the kNN algorithm. Logistic classifiers were applied to identify bacteria of the *Bacillus* genus and the *Staphylococcus aureus* species while three data imputation techniques were tested: with zero, with the average of the missing attribute, and with kNN algorithm. From this latter imputation technique, two approaches were considered: average aggregation function and median aggregation function. The adopted experimental protocol investigated the imputation influence on classification results under different scenarios regarding missing variables number. The results show that both kNN's approaches did not promote significant reduction on classifiers' performance when compared with complete data approach and that the classification of imputed data by kNN presented superior performance to that of other considered methods.

Keywords: Imputation with kNN, Mass Spectrometry, Logistic Regression, Bacterial Classification

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

AIC	*Akaike Information Criterion*
API	*Application Programming Interface*
BPCV	Bactérias que Promovem Crescimento Vegetal
CART	*Classification And Regression Tree*
EM	Espectrometria de Massa
GLM	*Generalized Linear Models*
IRLS	*Iteratively Re-weighted Least Squares*
KDD	*Knowledge Discovery in Databases*
kNN	k *Nearest Neighbor*
lbdi	Lista de Bases de Dados Incompletas
MAR	*Missing At Random*
MCAR	*Missing Completely At Random*
MNAR	*Missing Not At Random*
NCBI	*National Center for Biotechnology Information*
NVF	Número de Variáveis Faltantes
RMSE	*Root Mean Squared Error*
SE	*Squared Error*
SVM	*Support Vector Machine*
VCSDI	Validação Cruzada com Simulação de Dados Incompletos

SUMÁRIO

1 INTRODUÇÃO

A utilização de classificadores de padrões para a identificação da espécie ou gênero de bactérias que influenciam a produção agrícola tem se mostrado útil no planejamento das atividades agrícolas e para a manutenção da segurança alimentar. Um exemplo é o emprego de classificadores na detecção de Bactérias que Promovem o Crescimento Vegetal (BPCV) (SOUZA; AMBROSINI; PASSAGLIA, 2015), em amostras de solo e plantas a fim de se obter informações sobre a carência de nutrientes. Outra aplicação é identificação computacional de bactérias que estão relacionadas a doenças em humanos (XIE *et al.*, 2012).

Uma das formas de abordar esta tarefa é desenvolver classificadores que analisam, para as diferentes espécies e gêneros, os padrões de resposta das proteínas ribossomais em espectros de massa (EM) do tipo MALDI-TOF (Matrix-assisted laser desorption ionization/Time-of-flight, em tradução livre Matriz Assistida por Dessorção e Ionização por Tempo de Voo). Deve ser notado que o formalismo empregado na construção de tais classificadores deve permitir o tratamento da incerteza. Além disso, o procedimento usado no treinamento do classificador deve ser robusto a existência viés de seleção na amostra de treinamento. Conforme observado por, Lachish e Murray (2018), ambos os problemas são comuns quando do desenvolvimento de classificadores para aplicações biológicas.

Considerando isto, este trabalho propõe o uso da regressão logística no desenvolvimento de classificadores probabilísticos para identificação da espécie/gênero de bactérias a partir de dados de relacionados à detecção de proteínas ribossomais em espectros de massa. A escolha do classificador logístico se deve ao fato do mesmo implementar uma regra de decisão que utiliza a probabilidade posterior de cada classe para selecionar a hipótese de classificação mais provável. Além disso, segundo Krautenbacher, Theis e Fuchs (2017), a regressão logística não é particularmente sensível ao viés de seleção quando aplicada em tarefas que envolvem a predição.

Uma dificuldade que pode surgir quando do uso de classificadores na identificação da espécie/gênero de bactérias com dados de MALDI-TOF é que a instância a ser processada pode estar incompleta. Isto acontece quando o espectro a ser processado não informa a posição dos picos espectrais de algumas das proteínas usadas pelo classificador para discriminar cada classe. Segundo Eckel-Passow *et al.* (2009) e Souto, Jaskowiak e Costa (2015), as duas principais estratégias para tratar o problema da classificação de padrões com dados incompletos são o descarte das instâncias e a imputação de dados, isso é, completar registros que possuem dados faltantes. A primeira estratégia nem sempre é viável, em virtude de impossibilitar a realização de tarefas que dependem deste resultado. Quanto a segunda, Alasalmi *et al.* (2015) e Zhang

(2016) salientam que os algoritmos de imputação empregam métodos estatísticos para estimar os valores das variáveis não observadas. Como isto adiciona incerteza aos dados e aos resultados obtidos pelo classificador, é necessário escolher aqueles que não reduza significativamente o desempenho do classificador.

Neste contexto, de acordo com Lu *et al.* (2011) e Kim *et al.* (2014), o algoritmo kNN (k *Nearest Neighbor*) tem se mostrado um procedimento eficaz na imputação de valores faltantes em dados biológicos. Em vista do exposto, o objetivo geral deste trabalho é avaliar a influência de métodos de imputação em padrões de dados incompletos extraídos de pesos moleculares de proteínas ribossomais sobre o desempenho de classificadores baseados em regressão logística para identificação de bactérias. Sendo os objetivos específicos:

- avaliar como o desempenho do classificador é afetado pelo uso do método kNN na imputação de dados;
- comparar os resultados obtidos pelo kNN com aqueles obtidos com às demais estratégias de imputação e utilizando dados completos;
- determinar a influência do número de atributos faltantes sobre o desempenho do classificador e sobre o erro da imputação com o kNN.

A fim de atingir estes objetivos executou-se um conjunto de experimentos em que a tarefa alvo era a identificação de bactérias do gênero *Bacillus* e da espécie *Staphylococcus aureus*. A escolha do gênero *Bacillus* se deve ao fato deste estar ligado tanto ao crescimento vegetal quanto ao fato de algumas de suas espécies estarem associadas a doenças em humanos. A espécie *Bacillus cereus*, por exemplo, já foi utilizada na nodulação de soja em experimentos realizados em campo e câmaras de crescimento (SILVEIRA, 2008). Já a espécie *S. aureus* está intimamente ligada a casos de intoxicação a alimentar (ARGUDÍN; MENDOZA; RODICIO, 2010). Além disso, Michelin *et al.* (2005), Souza *et al.* (2015) e Miranda *et al.* (2015) relatam que a *S. aureus* é alvo de pesquisas relacionadas a aplicação de plantas medicinais para tratamento de enfermidades geradas pela mesma.

Os dados usados neste trabalho foram extraídos da *Base de dados PUKYU* criada por Tomachewski, Galvão e Etto (2018) e descrita nos trabalhos de Tomachewski (2017) e Tomachewski *et al.* (2018). Os dados da base *PUKYU* foram usados para gerar dois conjuntos de dados, um para o gênero *Bacillus* e outro para a espécie *S. aureus*. O protocolo experimental aplicou um procedimento chamado de Validação Cruzada com Simulação de Dados Incompletos (VCSDI) no treinamento e avaliação dos classificadores. No VCSDI, a partição de teste de

cada iteração da validação cruzada foi submetida à uma rotina que simulava dados faltantes e gerava bases de dados incompletas. Os dados faltantes eram tratados com os seguintes métodos de imputação: preenchimento com valor médio, preenchimento com zero e preenchimento com kNN. Em seguida, mediu-se a acurácia, a acurácia balanceada, o RMSE (*Root Mean Square Error*), a sensibilidade e a especificidade do classificador tanto nas base de teste completa quanto nas bases de teste com dados imputados.

A fim de verificar a eficácia dos métodos de imputação sob diferentes configurações da base de teste, os experimentos foram processados considerando casos em que havia apenas um atributo faltante até casos em que haviam três atributos faltantes. Também foram realizados teses com um número variável de atributos faltantes por instância. O resultados dos testes foram sujeitos a análise estatística.

Os resultados obtidos mostraram que a imputação com kNN não ocasionou uma queda no desempenho dos classificadores. Os erros de imputação com o kNN foram menores que aqueles observado quando da utilização da imputação com a média e a imputação com zero. Este comportamento colaborou para a redução do erro quadrático no cômputo da probabilidade posterior. Isto contribuiu para a obtenção de um desempenho similar àquele obtido na base de dados completa.

Este trabalho é organizado como segue. O Capítulo 2, apresenta a revisão bibliográfica onde são descritos os conceitos para o desenvolvimento deste trabalho, finalizando com os trabalhos correlatos. O Capítulo 3 descreve os materiais e métodos aplicados no desenvolvimento dos experimentos, o procedimento de VCSDI e de análise dos resultados. O Capítulo 4 apresenta os resultados e discussão do trabalho, finalizando este trabalho com as conclusões no Capítulo 5.

2 REVISÃO BIBLIOGRÁFICA

O problema da identificação automática de bactérias pode ser abordado com o uso de classificadores (VIJAYKUMAR, 2016; LEE *et al.*, 2017; TOMACHEWSKI *et al.*, 2018). Seguindo a mesma estratégia, este trabalho emprega a regressão logística para determinar se um dado objeto (bactéria) pertence a uma espécie/gênero de bactérias. A utilização deste tipo de classificador frequentemente requer a execução de um processo de aprendizagem de máquina (KANG *et al.*, 2018) a fim de estimar os parâmetros do mesmo. Uma vez realizado o aprendizado, o classificador permite calcular a probabilidade posterior de cada rótulo de classificação e, a partir daí a seleção da hipótese mais provável.

Ao realizar a classificação das bactérias a partir da análise dos padrões observados em dados massa/carga extraídos de espectros do tipo MALDI-TOF relacionados a proteínas ribossomais é importante observar que este padrões podem ser incompletos (ZIEGLER *et al.*, 2015). Uma forma usual de contornar este problema é aplicar um algoritmo de imputação de dados. Contudo, antes disto é necessário avaliar como a eficácia do classificador é afetada pela utilização de um algoritmo de imputação, como é o caso do kNN aqui proposto.

Como a definição de critérios de avaliação passa por um levantamento das características do problema, este capítulo apresenta uma descrição dos principais termos e conceitos utilizados neste trabalho. Assim, a Seção 2.1 descreve a espectrometria de massa do tipo MALDI-TOF e seu emprego na aquisição de biomarcadores para detecção e identificação de bactérias. A Seção 2.2 apresenta os principais conceitos relacionados aos classificadores, seu desenvolvimento e a análise de seu desempenho. Em particular, são detalhadas as etapas de pré-processamento dos dados, treinamento e teste do classificador, finalizando com análise estatística dos resultados. Essa também demonstra o funcionamento de um classificador dentro do KDD, suas métricas de avaliação e como se dá o processo de seleção de atributos. A Seção 2.3 complementa esses conceitos demonstrando a forma de se tratar casos onde há dados faltantes com ênfase na técnica do kNN. Os conceitos finais são apresentados na Seção 2.4 sobre Regressão Logística e estimação dos parâmetros do modelo gerado. E a Seção 2.5 apresenta os trabalhos correlatos.

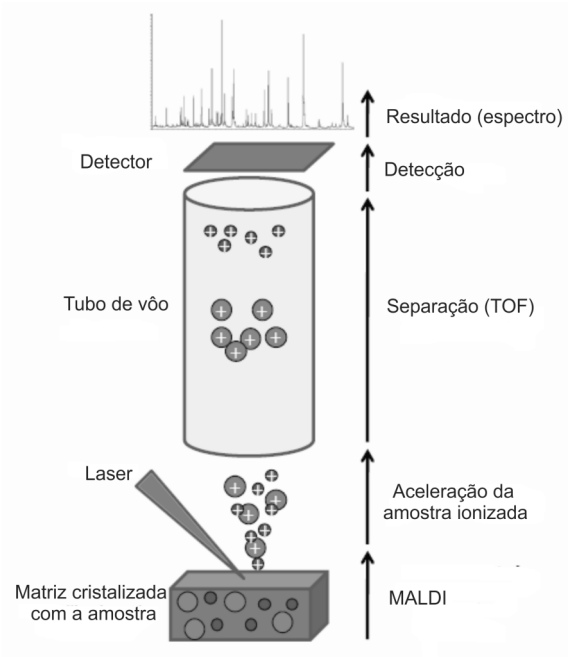
2.1 IDENTIFICAÇÃO DE BACTÉRIAS USANDO MALDI-TOF MS

A Espectrometria de Massa (EM) é uma técnica da Química Analítica usada para identificar e quantificar moléculas presentes na amostra de uma substância (GOULART; RESENDE, 2013). Em termos gerais consiste em empregar um aparelho chamado espectrômetro de massa

que quebra moléculas de uma amostra do material a ser analisado. Para a realização deste procedimento, inicialmente as moléculas são ionizadas fazendo com que estes fiquem com uma carga elétrica maior ou menor do que o elemento original. Posteriormente, o aparelho realiza a separação dos átomos e moléculas com base na relação massa/carga (*m/z*) permitindo assim sua identificação e quantificação (GROSS, 2011). Aqui *m/z* denota a quantidade em daltons que se obtém pela divisão do número de massa do íon pelo seu número de carga (TODD, 1991).

De acordo com Ziegler *et al.* (2015) um tipo de EM que tem se destacado no processo de identificação de bactérias é o MALDI-TOF (ver Figura 1). Em um espectrômetro de massa por tempo de voo as partículas que foram carregadas durante o processo de ionização são atraídas em direção a um detector usando um campo elétrico. A tensão usada para atrair os íons é constante e a velocidade com que cada partícula se move em direção ao detector depende da sua relação *m/z*. O tempo gasto para cada partícula atingir o detector é chamado de tempo de voo e também depende da razão *m/z*. Basicamente, moléculas com menor *m/z* são mais velozes e chegam mais rapidamente ao detector. Logo, o tempo de voo pode ser visto como uma característica do átomo ou molécula que atingiu o detector (GOULART; RESENDE, 2013).

Figura 1: Esquema de funcionamento da técnica MALDI/TOF passando por todas as etapas até a produção do espectro de massa



Fonte: Goulart e Resende (2013)

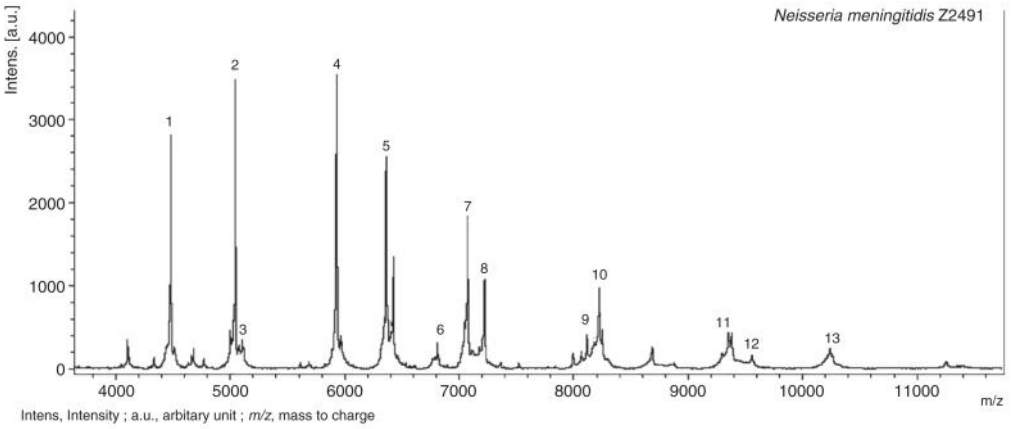
O espectro de massa obtido por MALDI-TOF é um gráfico de duas dimensões em que o eixo das abcissas indica a relação *m/z* e o eixo das ordenadas indica a intensidade do pico (GROSS, 2011). Como a relação *m/z* é proporcional ao tempo de voo, um valor da ordenada maior que zero em uma abcissa indica a presença de uma determinada partícula na amostra que

deu origem ao espectro (TODD, 1991; GROSS, 2011).

De acordo com Tamura, Hotta e Sato (2013) determinados picos do espectro de massa, são associados à proteínas ribossomais. A característica que torna a proteína ribossomal interessante está no fato delas participarem de funções vitais ao longo de toda a vida da célula. Além disso, as proteínas ribossomais também possuem um alto grau de conservação, isto é, quase não sofrem alterações nas sequências de aminoácidos ao longo dos estágios de vida celular (ZIEGLER *et al.*, 2015).

Esta perenidade fez com que as proteínas ribossomais fossem consideradas como biomarcadores confiáveis para a identificação de bactérias (ZIEGLER *et al.*, 2015). Um biomarcador, ou marcador biológico é definido como um quantificador de características em processos biológicos, isso é, permite que características sejam medidas de maneira objetiva e associadas a um processo biológico (STRIMBU; TAVEL, 2010). Essas proteínas podem ser associadas a padrões de comportamento/resposta que são observáveis em testes laboratoriais. Adicionalmente, tais padrões fornecem informações que podem ser usados para discriminar a espécie ou gênero de uma bactéria (TAMURA; HOTTA; SATO, 2013).

Figura 2: Exemplo de Espectro de Massa com indicação de picos para a estirpe *Neisseria meningitidis Z2491*



Fonte: (SUAREZ *et al.*, 2013)

Para evidenciar o padrão dos biomarcadores em um espectro de massa, inicialmente determina-se a abcissa, valor de m/z, relativo a cada pico do espectro. Em seguida seleciona-se aquelas abcissas de m/z que estão associadas às proteínas ribossomais que funcionam como biomarcadores para o objeto alvo. Esta seleção define um vetor de abcissas $(x_1, \ldots, x_n)$ com as posições dos picos relacionadas às proteínas que tem alta intensidade no espectro de massa de uma determinada espécie/estirpe/gênero. Uma função de classificação pode então, comparar os dados do vetor com o padrão de distribuição dos picos de uma classe a fim de determinar se o primeiro é uma instância do segundo.

A Figura 2 representa um espectro de massa retirado de Suarez *et al.* (2013). O espectro mostra 13 picos específicos para a espécie *Neisseria meningitidis* sendo estes dados obtidos da base de dados *Andromas*.

2.2 CLASSIFICADORES DE PADRÕES

A classificação de padrões tem como objetivo desenvolver modelos e algoritmos que possam ser aplicados para identificar a categoria de um objeto a partir de um conjunto atributos ou características. Em termos matemáticos, um classificador é uma função $f : \mathbf{X} \rightarrow Y$ que implementa um modelo preditivo que permite a determinação do valor da variável categórica Y a partir dos valores das variáveis $x_1 \ldots x_p \in \mathbf{X}$ (FACELI *et al.*, 2011). Os valores de Y, denotados por $y_1 \ldots y_l$, são rótulos que enumeram as possíveis hipóteses de classificação. As variáveis em $\mathbf{X}$ representam os atributos descritores. Uma instanciação conjunta das variáveis em $\mathbf{X}$ é denotada por $\mathbf{x}$. Uma instanciação da variável X_j é simbolizada por x_j.

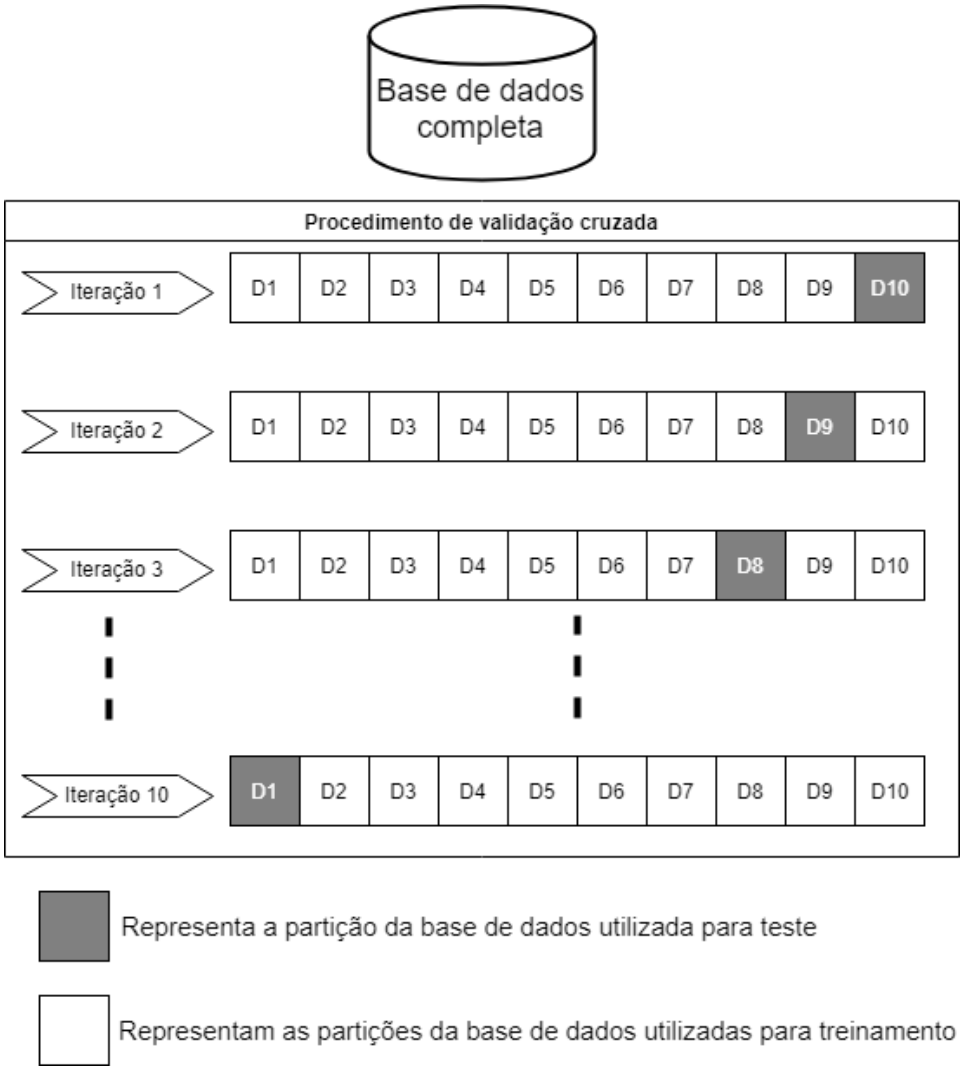
O desenvolvimento e o estudo de classificadores está inserido no escopo da Descoberta de Conhecimento em Base de Dados (KDD, do inglês *Knowledge Discovery in Databases*). O KDD é um processo que emprega métodos computacionais para de transformar dados em informações úteis para tomada de decisões em um certo domínio (TAN *et al.*, 2006). Dada a sua complexidade o KDD é, usualmente, dividido em três etapas: o pré-processamento dos dados, a mineração de dados e o pós-processamento. A etapa de pré-processamento tem como objetivo primordial a melhora da qualidade dos dados com a remoção de ruídos, inconsistências e discrepâncias. Na mineração de dados empregam-se a análise estatística e algoritmos para reconhecimento de padrões e aprendizagem de máquina a fim de se detectar associações entre as variáveis do problema alvo. O pós-processamento tem o objetivo de validar os resultados obtidos (análise estatística, teste de hipóteses, análise de desempenho, dentre outras) e verificar a sua utilidade e interessabilidade.

Neste contexto, o desenvolvimento de classificadores é visto como um problema de aprendizagem supervisionada. Neste caso, um algoritmo de aprendizagem de máquina é utilizado para minerar a estrutura (forma de f) e os parâmetros da função de classificação a partir de observações que informam os valores de $\mathbf{X}$ e Y para uma coleção de objetos previamente classificados (RUSSEL; NORVIG, 2004). As observações são armazenadas em uma base de dados $\mathbf{D}$ com m entradas da forma $(x_1, x_2, ..., x_p, y_a)$.

Quando a estrutura do modelo é especificada previamente, o aprendizado se reduz ao problema do treinamento de parâmetros. Assim, seja $\mathbf{B}$ um vetor cujos elementos representam os parâmetros da função de classificação f. O treinamento de parâmetros diz respeito a estima-

ção dos elementos de **B** a partir da inspeção dos casos em **D** (WITTEN *et al.*, 2016). Neste tipo de problema é usual que se particione a base de dados $\mathbf{D}$ em dois subconjuntos, treinamento e teste (FACELI *et al.*, 2011). O conjunto de treinamento é usado na estimação dos parâmetros $\mathbf{B}$ e é definido como $\mathbf{D}_{treino} = \{(\mathbf{x}_i, y_i), i = 1..., p\}$, onde p denota o número total de atributos. O conjunto de testes é definido como $\mathbf{D}_{teste} = \{(\mathbf{x}_i, y_i), i = 1..., p\}$, e é usado para avaliar o desempenho do classificador. Durante o teste do classificador, o valor da classe, y, não é informado e deve ser computado pela função de classificação. A principal motivação para o particionamento dos dados é assegurar que as medidas de desempenho serão obtidas a partir de dados diferentes dos utilizados na fase de treinamento (WITTEN *et al.*, 2016).

Figura 3: Exemplo de validação cruzada para b = 10



Fonte: O Autor

Uma abordagem convencional para a partição dos dados é a validação cruzada (do inglês, *cross-validation*) (FACELI *et al.*, 2011). Nela, a base é particionada em *b* subconjuntos de tamanho similar usando um procedimento que distribui aleatoriamente os casos de $\mathbf{D}$ entre as

partições. Durante o treinamento e validação do classificador uma das partições é selecionada como base de teste enquanto as $b-1$ restantes são usadas para o treinamento. Este processo é repetido b vezes de forma que na l-ésima iteração da validação, a l-ésima partição é usada como base de teste. O desempenho do classificador é então avaliado observando-se a média do desempenho obtido em todas as iterações.

A Figura 3 ilustra um exemplo de validação cruzada para $b = 10$. Inicialmente tem a *Base de dados completa* que é particionada em dez partes $D_1, D_2, D_3..., D_{10}$. Em cada iteração uma das dez partições, $\mathbf{D}_i$, é escolhida para a realização dos testes ao passo que a união dos conjuntos restantes, $\cup_{b=1,b\neq i}^{10}\mathbf{D}_b$, é processada pelo algoritmo de aprendizagem de parâmetros. Assim, as dez iterações produzem dez classificadores diferentes, cada um com seu escore de desempenho. Após a última iteração os escores de desempenho aferidos são usados para estimar o desempenho médio do classificador em relação às observações contidas na base de dados.

2.2.1 Medidas de desempenho de classificadores

De acordo com Faceli *et al.* (2011), uma das métricas usadas para avaliar o desempenho de um classificador é a matriz de confusão. É uma matriz quadrada $M_{k\times k}$ em que as linhas indicam as classes verdadeiras e as colunas as classes preditas pelo classificador. O elemento $m_{i,j}$ armazena o número de vezes que o classificador selecionou a hipótese $Y = j$ para um caso rotulado como $Y = i$ na base de teste. A diagonal da matriz se refere às instâncias classificadas corretamente.

A Tabela 1 mostra um exemplo de matriz de confusão para o caso em que Y é uma variável binária. Neste exemplo, uma das classes é denotada como positiva (+) e a outra como negativa (-). O símbolo VP indica o número de casos casos verdadeiros positivos, instâncias da classe positiva classificados corretamente. VN representa o número de casos verdadeiros negativos, instâncias que pertencem a classe negativa e foram classificados de maneira correta. FP denota os falsos positivos e informa o número de instâncias que pertencem a classe negativa, mas foram classificados como pertencentes a classe positiva. Por fim, FN indica os casos falsos negativos, instâncias que pertencem a classe positiva, mas foram erroneamente classificados como negativos. A soma de todos os valores presentes na matriz de confusão retorna o número total de instâncias utilizadas na etapa de teste do mesmo (N) (FACELI *et al.*, 2011).

A medida de desempenho mais comumente utilizada para avaliar classificadores é a acurácia (ac) (TAN *et al.*, 2006). Ela pode ser obtida a partir da matriz de confusão, indica a taxa de acerto do classificador e é calculada como a soma dos elementos da diagonal principal divida por N (Equação 1).

Tabela 1: Matriz de confusão para um caso binário

		Classe Predita	
		+	-
Classe real	+	VP	FN
	-	FP	VN

$$ac = \frac{VP + VN}{N} \tag{1}$$

Outras duas medidas que podem ser extraídas da matriz de confusão são a Sensibilidade (Equação 2), que corresponde a taxa de acerto da classe positiva, e a Especificidade (Equação 3), que corresponde a taxa de acerto na classe negativa.

$$sensibilidade = \frac{VP}{VP + FN} \tag{2}$$

$$especificidade = \frac{VN}{VN + FP} \tag{3}$$

Deve ser notado que, em determinados domínios, ocorre da base de dados usada durante o treinamento ser desbalanceada. Segundo Faceli *et al.* (2011), uma base de dados é desbalanceada se um subconjunto de classes apresenta uma frequência maior do que outros. Por exemplo, 80% dos casos da base de dados pertencem a classe "+"e apenas 20% à classe "−". Nesta situação é sugerido que a taxa de acerto do classificador seja estimada com a acurácia balanceada (Equação 4) (BEKKAR; DJEMAA; ALITOUCHE, 2013). Isto porque como *acBal* é média aritmética da sensibilidade e da especificidade, ela permite detectar situações em que o desempenho do classificador é satisfatório apenas para a classe majoritária.

$$acBal = \frac{sensibilidade + especificidade}{2} \tag{4}$$

Se a função de classificação retornar um valor real, como é o caso dos classificadores logísticos, que calculam a probabilidade posterior de cada rótulo dada uma instância $\mathbf{x}$, é possível utilizar a medida do RMSE (Raiz do Erro Médio Quadrático, do inglês *Root Mean Squared Error*) para avaliar o desempenho do classificador (AMBLER; OMAR; ROYSTON, 2007; JAPKOWICZ; SHAH, 2011). O RMSE é dado pela expressão:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(\hat{p_{y_t}} - p_{y_t})^2}{N}} \tag{5}$$

Onde, N indica o total de linhas da base de teste, t corresponde ao índice de uma linha da base de teste. Seja y_t a hipótese de classificação correta para a linha t, $\hat{p}_{y_t}$ é o valor da probabilidade atribuído pelo classificador a y_t e p_{y_t} é o valor verdadeiro da probabilidade para a linha t.

2.2.2 Seleção de atributos

Um dos critérios que norteiam o desenvolvimento de classificadores é a Navalha de Occam (RUSSEL; NORVIG, 2004). Segundo este critério, dados dois modelos, M_1 e M_2, com desempenhos equivalentes deve-se escolher o mais simples dos dois como resultado do processo de aprendizagem. Ao preferir o modelo mais simples a Navalha de Occam evita o custo computacional relacionado ao treinamento e à validação de um modelo mais complexo.

O número de atributos que compõe a função de classificação é um dos fatores que incrementam a complexidade da tarefa de treinamento de classificadores. Isto ocorre porque o treinamento de modelos com muitos atributos, usualmente, demanda a disponibilidade de grandes bases de dados para o seu treinamento (THEODORIDIS; KOUTROUMBAS, 2008) e estas nem sempre estão disponíveis. Por outro lado, o uso de bases de dados com poucos casos pode não fornecer uma amostra representativa do comportamento de cada classe (BRAIN; WEBB, 1999). Além disto há problemas como o *overfitting*, que consiste em modelos que se tornam muito específicos, não sendo capazes de predizer de maneira satisfatória novos conjuntos de dados. Isto é algo que pode ocorrer devido ao grande número de atributos. Este problema está diretamente ligado aos princípios básicos que navalha de Occam estabelece. Ou seja, ter mais atributos do que o necessários para a descrição do atributo meta implica diretamente no custo computacional que o classificador estabelece (HAWKINS, 2004).

Outro problema é que a quantidade de atributos está relacionada é a maldição da dimensionalidade. De acordo com Bellman (2015), isso refere-se ao fato de que um número elevado de atributos necessita de um número elevado de instâncias para que a tarefa de classificação seja feita de maneira confiável, a fim de atribuir todos os objetos a classe correta.

Uma maneira de simplificar a estrutura dos classificadores a serem gerados pelo aprendizado de máquina é aplicar um procedimento de seleção de atributos durante a etapa de pré-processamento (TAN *et al.*, 2006; FACELI *et al.*, 2011). Tal procedimento consiste na escolha de um subconjunto de atributos, que dado um critério objetivo, descrevam satisfatoriamente o comportamento da base de dados. A ideia é evidenciar os atributos mais relevantes para a criação do classificador de forma a se obter um modelo mais simples, com desempenho adequado aos objetivos da aplicação alvo (TAN *et al.*, 2006).

As técnicas automáticas para seleção de atributos podem ser agrupadas em três catego-

rias: interna, baseada em filtro e baseada em *wrapper*[1]. Na abordagem interna, a seleção de atributos é feita durante a aprendizagem do modelo, isto é, o próprio algoritmo decide quais serão os atributos que serão utilizados e quais serão eliminados. Isso acontece por exemplo, na aprendizagem de árvores de decisão com o algoritmo J48 (KARABULUT; ÖZEL; IBRIKCI, 2012). Na abordagem baseada em filtro, as características são selecionadas durante o pré-processamento por meio do emprego de heurísticas que estimam o poder de discriminação dos atributos. Como é o caso do *Correlation Based Feature Selection*(YILDIRIM, 2015). Na abordagem baseada em *wrapper*, emprega-se um procedimento de busca do melhor subconjunto de atributos dado um critério definido sobre o desempenho do classificador. Para tanto, o procedimento de busca executa o algoritmo de aprendizagem usando diferentes conjuntos de descritores e o desempenho do modelo é então otimizado em relação a algum escore como acurácia e erro quadrático.

Uma técnica que implementa a abordagem *wrapper* é a *stepwise* (HAIR *et al.*, 2009). Este método executa a seleção de atributos em dois modos básicos *backward* ou *forward*. No modo *backward* o conjunto de atributos selecionados é inicializado com todas as variáveis da base de dados $\mathbf{D}$ e, a cada passo do processo de busca o algoritmo remove um dos descritores pré-selecionados. No modo *forward*, o conjunto de atributos é criado como um conjunto vazio. Na sequência o algoritmo de seleção adiciona uma variável ao conjunto de descritores a cada iteração. Em ambos os casos, o algoritmo é interrompido quando o maior poder discriminatório é encontrado.

O procedimento *stepwise com AIC*, também denominado *stepAIC*, provê uma implementação do algoritmo *stepwise* em que o desempenho do modelo é avaliado com o score AIC (Critérios de Informação de Akaike, do inglês *Akaike Information Criterion*) (VENABLES; RIPLEY, 2013). Assim, o *stepAIC* procura determinar o conjunto de atributos que minimiza o AIC do modelo classificador definido como:

$$AIC = 2Q - 2log(L(\mathbf{B}|y)) \tag{6}$$

onde Q indica o número de parâmetros do modelo e $L(\mathbf{B}|y)$ é o valor máximo da verossimilhança do modelo.

2.3 CLASSIFICAÇÃO COM DADOS INCOMPLETOS

Seja $\mathbf{D}$ uma base de dados, conforme definido na Seção 2.2. $\mathbf{D}$ é dita incompleta (ou com dados faltantes) se existe $\mathbf{D}_f \subseteq \mathbf{D}$ tal que $\mathbf{D}_f = \{(\mathbf{x}_i, y_i) : (\mathbf{x}_i, y_i) \in \Omega_{\mathbf{X}} \times \Omega_{\mathbf{Y}} \wedge m(\mathbf{x_i})\}$

[1]Também denominada envoltória.

(WILLIAMS *et al.*, 2007). Aqui, $m(\mathbf{x_i})$ é um predicado que é verdadeiro se e somente se existe no mínimo um atributo X_k cujo valor não é informado em $\mathbf{x_i}$. Neste caso diz-se que x_k é um dado faltante.

De acordo com Rubin (1976) e Little e Rubin (2014) problemas que envolvem bases de dados incompletas são organizados em três categorias: MCAR (Perdas Completamente ao acaso, do inglês *Missing Completely At Random*), quando não existe uma relação que associe a falta de um atributo aos demais atributos da base de dados, ausentes ou observados. MAR (Perdas ao acaso, do inglês *Missing At Random*), quando a ausência do dado referente a uma variável X_j na observação $\mathbf{x}$ está relacionada com os valores observados para um subconjunto dos elementos de $\mathbf{X} \setminus \{X_j\}$ mas não está relacionada aos demais valores ausentes. MNAR (Perdas não aleatórias, do inglês *Missing Not At Random*), trata-se do caso em que valores faltantes de um conjunto de variáveis específicas estão relacionados aos valores observados deste conjunto.

Segundo e Ambler, Omar e Royston (2007), as duas alternativas básicas para o tratamento de dados faltantes são a eliminação de instâncias incompletas e a imputação de dados. Na primeira abordagem a ideia é remover os registros incompletos e executar o classificador apenas sobre os casos completos. Na segunda alternativa, emprega-se um procedimento que estima os valores faltantes dos atributos de forma a completar as instâncias. Só então elas são submetidas ao classificador.

Dentre as diversas técnicas para imputação de dados Hastie *et al.* (1999), Saar-Tsechansky e Provost (2007), Beretta e Santaniello (2016) e Williams *et al.* (2007) destacam as seguintes:

1. Imputação com zero: atribui um valor zero aos dados faltantes; isto permite que todas as instâncias sejam mantidas, porém o poder de discriminação fica comprometido devido ao valor zero fazer com que o atributo seja desconsiderado do processo de classificação;

2. Imputação com a média: utiliza os casos observados para calcular a média para cada atributo X_j, este valor é então atribuído aos casos em que o valor de X_j não é conhecido;

3. Imputação com a mediana: similar ao item anterior, porém atribui a mediana de X_j ao dado faltante;

4. Imputação com modelos de predição: emprega um modelo funcional (obtido com alguma técnica de regressão) para estimar, para cada instância, os valores dos dados faltantes a partir dos valores dos atributos observados;

5. Imputação com métodos não paramétricos baseados em distâncias ou similaridade: não

exploram a distribuição da população, de forma que os dados são responsáveis por definirem sua própria estrutura de maneira explícita ou implícita.

Outras abordagens para imputação podem ser encontradas em Batista e Monard (2002), Ambler, Omar e Royston (2007), Remus *et al.* (2008), Kang (2013).

2.3.1 Imputação com o algoritmo kNN

Um procedimento de imputação não paramétrica que considera a distância entre uma instância incompleta e as demais amostras do conjunto de casos a ser classificado é aquele que usa o algoritmo kNN (k-Vizinhos Mais Próximos, do inglês *k-Nearest Neighbors*) (LU *et al.*, 2011; KIM *et al.*, 2014). O Algoritmo 1 apresenta o pseudocódigo de um procedimento imputação baseado em kNN (WEINBERGER; SAUL, 2009).

Algoritmo 1: IMPUTAÇÃO DE VALORES FALTANTES UTILIZANDO KNN

Entrada: Base de dados **D**, **k**, $h(\cdot)$

Saída: Base de dados **D** completa

```
1  início
2      para cada registro x_i ∈ D com pelo menos 1 valor faltante faça
3          para cada x_j ∈ D com x_j ≠ x_i faça
4              Calcule a distância (x_i,x_j)
5              Salvar a distância e x_j em vetor de Similaridade (S)
6          fim
7          Ordenar o vetor (S) deixando os vizinhos mais próximos por primeiro
8          Selecionar os k valores do topo de S e imputar o(s) valor(es) do(s) atributo(s) faltante(s) de x_i com base
             nos valores de S, utilizando h(·) (média, mediana ou média ponderada)
9      fim
10 fim
11 retorna D
```

Como pode ser visto, o Algoritmo 1 não usa uma função matemática (linear, não linear) ou uma distribuição de probabilidades para quantificar o relacionamento entre os atributos e então estimar cada atributo faltante a partir dos demais. Em vez disso, ele explora a similaridade entre os casos para construir uma estimativa. O algoritmo recebe como entrada a base de dados ($\mathbf{D}$) com dados completos e incompletos, o número de vizinhos mais próximos (k) e a função de agregação aplicada, denotada por $h(\cdot)$. Durante a execução, cada instância $\mathbf{x}_i$, com valores faltantes, tem sua distância d_{ij} para cada outro $\mathbf{x}_j \in \mathbf{D}$ calculada. As distâncias obtidas são armazenadas em um vetor $\mathbf{S}$ cujas entradas são da forma $(\mathbf{x}_j, d_{ij})$. Em seguida, o vetor $\mathbf{S}$ é então ordenado de maneira decrescente e o registro $\mathbf{x}$ é completado com base na função de

agregação dos $\mathbf{k}$ primeiros elementos de $\mathbf{S}$. Este processo é repetido até que todas as instâncias de $\mathbf{D}$ estejam completas. Algumas funções de agregação usuais são a média, mediana ou média ponderada (WEINBERGER; SAUL, 2009).

A distância mais utilizada para calcular a diferença entre cada par de objetos é a distância Euclidiana (HU *et al.*, 2016). Nesta norma, dados dois pontos (ou vetores) $\mathbf{p}$ e $\mathbf{q}$ de um espaço euclideano de dimensão n, com $\mathbf{p} = (p_1, p_2, ..., p_n)$ e $\mathbf{q} = (q_1, q_2, ..., q_n)$, a distância entre esses desse pontos é $d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n}(\mathbf{q}_i - \mathbf{p}_i)^2}$.

Outra distância que também usada é a distância de Gower (GOWER, 1971), a qual permite a processamento de entradas com dados numéricos, lógicos, categóricos ou textos, ou mesmo a combinação desses. Para o cálculo de valores numéricos a distância pode ser obtida por $S_{ijk} = 1 - |x_{ik} - x_{jk}|/r_k$, onde S_{ijk} denota i e j sendo os casos comparados na k-ésima variável, e r_k corresponde ao alcance para a k-ésima variável. E os valores das distâncias se alteram na escala de 0 até 1, onde 1 indica que os valores comparados são idênticos, $x_{ik} = x_{jk}$, e 0 para casos onde x_{ik} e x_{jk} são extremos.

A Figura 4 ilustra diferentes funções de agregação ($h(\cdot)$). Os conjuntos de dados $\mathbf{O}$ e $\mathbf{L}$, tem como objetivo encontrar os k pontos mais próximos, sendo considerado $k = 1$, de O para todos os pontos em L utilizando uma determinada função de agregação. Inicialmente são apresentados L_1, L_2 e L_3 que são os pontos de base L, onde deseja-se calcular os objetos agregados de k mais próximos. O conjunto de pontos O possuí os objetos O_1 a O_{15}. Na Figura 4(a) considera-se como função de agregação a soma do vizinho mais próximo, onde O_{14} é o objeto cuja a soma das distâncias de L_1, L_2 e L_3 é mínima. Na Figura 4(b) é considerado o caso do máximo vizinho mais próximo, onde O_9 é o objeto em que a distância máxima para qualquer ponto de L_1, L_2 e L_3 é mínima. Por fim, para a Figura 4(c) apresenta a situação em que a função de agregação é a miníma do vizinho mais próximo, sendo O_6 o objeto cuja distância é mínima para qualquer ponto de L_1, L_2 e L_3.

Figura 4: Exemplo de diferentes funções de agregação para kNN

(a) Soma (b) Max (c) Min

Fonte: (ELMONGUI; MOKBEL; AREF, 2013)

O número de vizinhos (k) é outro parâmetro que tem efeito sobre o desempenho do kNN (BATISTA; MONARD, 2002; LU *et al.*, 2011; SCHMITT; MANDEL; GUEDJ, 2015). Se o valor k for pequeno, pontos com ruído podem afetar a escolha do valor, por outro lado se k é grande, objetos com pouca relação podem ser utilizados para a estimação. Hassanat *et al.* (2014) afirmam que a escolha do k ótimo para o número de vizinhos deve ser feito de maneira empírica. Segundo os autores diferentes valores de k podem ser utilizados em uma situação e a escolha do melhor valor vai depender do critério escolhido para avaliar o desempenho da imputação, como a acurácia caso a base imputada seja utilizada para classificação.

2.4 CLASSIFICAÇÃO BINÁRIA E REGRESSÃO LOGÍSTICA

De acordo com Kotu e Deshpande (2014), um classificador binário[2] é aquele que possui apenas duas classes como respostas possíveis para a variável dependente. Por exemplo, um classificador que pode atribuir os rótulos 0 e 1 para indicar se um objeto pertence ou não a uma determinada categoria. Assim, seja Y uma variável aleatória binária com espaço amostral $\Omega_Y = \{0, 1\}$, cujos valores indicam as hipóteses de classificação, e seja $\mathbf{X} = \{X_1 \dots X_p\}$ o conjunto de descritores cujo espaço amostral é $\Omega_i \in \mathbb{R}$. Como antes, a instanciação de X_i é denotada por x_i e uma instanciação conjunta de $\mathbf{X}$ é simbolizada por $\mathbf{x}$. Um classificador logístico é um classificador dicotômico que computa as probabilidades posteriores de cada hipótese para um determinado objeto e então emprega um critério de decisão para selecionar a mais provável (HOSMER; LEMESHOW; STURDIVANT, 2013).

As probabilidades posteriores das hipóteses $P(Y = 1|\mathbf{x})$ e $P(Y = 0|\mathbf{x})$ são calculadas de acordo com o modelo descrito nas Equações 7 e 8:

$$P(Y = 1|\mathbf{x}) = \frac{e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p}}{1 + e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p}} \tag{7}$$

$$P(Y = 0|\mathbf{x}) = 1 - P(Y = 1|\mathbf{x}) = \frac{1}{1 + e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p}} \tag{8}$$

Nas Equações 7 e 8, os termos $\beta_0, \beta_1 \dots \beta_p$, com $\beta_i \in \mathbb{R}$ para $i = 0..p$, são os parâmetros do modelo, também denotados por $\mathbf{B} = (\beta_0, \beta_1 \dots \beta_p)$.

De acordo com Duda, Hart e Stork (2012), o critério convencional para inferir a categoria do objeto de interesse usando um classificador probabilístico é a regra de decisão bayesiana. Segundo esta regra, o classificador deve identificar o objeto com o rótulo da hipótese mais

[2]Também denominado binomial ou dicotômico.

provável (com maior probabilidade posterior). No caso de classificadores binários, a regra especifica que o mesmo deve selecionar o rótulo $Y = 1$, se $P(Y = 1|\mathbf{x}) > 0{,}5$, e $Y = 0$, caso contrário (ALTMAN, 1968). Na regressão logística isto equivale a computar a razão $OR = \frac{P(Y=1|\mathbf{x})}{P(Y=0|\mathbf{x})}$, também chamada de razão das probabilidades, e reportar $Y = 1$ quando $OR > 1$. Senão, o classificador deve rotular o objeto com $Y = 0$.

A tomada de decisão quanto ao rótulo de um objeto é realizada a partir da equação:

$$logit(\mathbf{x}) = ln\left[\frac{P(Y = 1|\mathbf{x})}{1 - P(Y = 1|\mathbf{x})}\right] = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p \tag{9}$$

Nesta equação, *logit* denota a transformação logarítmica que permite converter a expressão da razão das probabilidades posteriores de cada hipótese em um modelo linear. A seleção da hipótese $Y = 1$ acontece quando $logit(\mathbf{x}) > 0$.

2.4.1 Estimação de parâmetros por verossimilhança

A regressão logística tem por objetivo estimar os parâmetros $\boldsymbol{\beta}$ da função de classificação descrita na Equação 7. Para tanto, usualmente, emprega o critério da máxima verossimilhança (HOSMER; LEMESHOW; STURDIVANT, 2013). Segundo este critério, o objetivo do treinamento de parâmetros é obter uma estimativa para $\mathbf{B}$ que maximize a expressão:

$$L(\mathbf{B}) = P(\mathbf{D}|\mathbf{B}) \tag{10}$$

Aqui, $L(\mathbf{B})$ é a função de verossimilhança, que calcula a probabilidade das instâncias observadas na base de treinamento $\mathbf{D}$ para uma instanciação dos valores de $\beta_0, \beta_1 \ldots \beta_p$.

Uma estimativa de $\mathbf{B}$ que maximiza $L(\mathbf{B})$ é chamada de estimador de máxima verossimilhança e é denotada por $\hat{\boldsymbol{\beta}}$ (Equação 11). Assumindo que $\mathbf{D} = \{d_1, \ldots d_r\}$, onde r é a última linha na base de dados, tal que os casos em $\mathbf{D}$ são independentes e identicamente distribuídos, tem-se que:

$$L(\mathbf{B}) = \prod_{i=1}^{r} P(x_i|\mathbf{D}). \tag{11}$$

De acordo com Shalizi (2013), no caso da regressão logística, a Equação 11 pode ser definida como:

$$L(\mathbf{B}) = \prod_{i=1}^{r} p(\mathbf{x}_i, \beta_i)^{y_i}(1 - p(\mathbf{x}_i, \beta_i))^{1-y_i} \tag{12}$$

Nesta equação $\mathbf{x}_i$ é uma instância dos atributos de $\mathbf{D}_{treino}$, y_i. p é a probabilidade da classe para $y_i = 1$ e $1 - p$ para $y_i = 0$.

Calculando-se o logaritmo de L(**B**) transforma-se o produto da Equação 12 em uma somatória. Isto produz um problema de otimização cujo o objetivo é maximizar:

$$l(\beta) = \sum_{i=1}^{r} (y_i ln(p(\mathbf{x}_i, \beta_i)) + (1 - y_i) ln(1 - p(\mathbf{x}_i, \beta_i))) \tag{13}$$

Um algoritmo que possibilita implementar a otimização da Equação 13 é o método de mínimos quadrados com ponderação interativa (IRLS, do inglês *Iteratively Re-weighted Least Squares*), descrito por Komarek (2004).

2.5 TRABALHOS CORRELATOS

2.5.1 Identificação de bactérias usando espectro MALDI-TOF

Diversos autores tem reportado resultados experimentais que indicam que o emprego de métodos de aprendizado de máquina, sobre dados de espectros MALDI-TOF, é uma estratégia eficaz para o desenvolvimento de sistemas para identificação de bactérias. Este é o caso do trabalho de Bruyne *et al.* (2011), no qual os autores usaram classificadores do tipo máquinas de vetores de suporte (SVM, do inglês *Support Vector Machine*) e *random forests* para reconhecer bactérias dos gêneros *Leuconostoc e Fructobacillus*. Para tanto, os autores desenvolveram um protocolo experimental para a geração de espectros de massa MALDI-TOF. Contudo, não especificaram os picos que foram usados no processo de identificação, apenas que os espectros gerados estiveram no intervalo de 2 a 20 kDa, tendo sido utilizados picos que apresentaram intensidade superior a $5x10^2$. O desempenho dos classificadores foi avaliado usando a validação cruzada com dez partições e foi medido com a acurácia e *F-score*[3]. A acurácia 98,4% para o SVM e 94,1% para a *random forests*. O *F-escore* foi de 96,8% para o SVM e 89,7% *random forests*. Os autores observam que esses resultados apresentam um *viés de seleção* em virtude das bases de dados utilizadas estarem desbalanceadas.

Lee *et al.* (2017) também fizeram o uso de SVM para classificação de bactérias utilizando dados de MALDI-TOF. O objetivo foi classificar espécies bacterianas dos grupos *Mycobacterium abscessus* e *Mycobacterium fortuitum*, onde o primeiro grupo continha 3 espécies de bactérias e 309 espectros e o segundo 5 espécies de bactérias e 285 espectros. Os dados espec-

[3]O cálculo deste escore consiste na definição de dois outros escores precisão p, e recall r, a precisão é definida por o número de casos positivos corretos (VP) dividido pelo total de resultados positivos (VP + FN) retornados pelo classificador, e o recall número de resultados positivos corretos (VP) divido pelos total de casos relevantes (VP + FN), sendo o F-score definido por $2.\frac{p*r}{p+r}$. sua importância é na verificação de precisão e robustez do classificador (SASAKI *et al.*, 2007)

trais usados no estudo foram providos pela companhia ASTA[4]. O desempenho do classificador foi mensurado pela a acurácia. Para o grupo *M. abscessus* obteve-se uma acurácia de 91,61% e para o grupo *M. fortuitum* 92,25%.

Tomachewski *et al.* (2018) apresentaram o software Ribopeaks para a identificação de bactérias a partir da análise dos padrões de picos no espectros MALDI-TOF associados às proteínas ribossomais. O sistema proposto utiliza um classificador probabilístico do tipo "Bayes Ingênuo"(RUSSEL; NORVIG, 2004) para codificar a incerteza no relacionamento entre as proteínas e as hipóteses de classificação, também computando a probabilidade posterior de cada hipótese. Durante o desenvolvimento do Ribopeaks os autores criaram uma base de dados, denominada *base de dados PUKYU*, que possuí 57 tipos diferentes de proteínas ribossomais, com um total de 28.505 registros, referentes a 6.936 espécies e 1.949 gêneros diferentes. Os dados foram extraídos do repositório NCBI[5]. Os autores testaram o desempenho do software Ribopeaks em dados extraídos de Ziegler *et al.* (2015), que possuía 116 registros de amostras bacterianas, que não foram utilizadas para o treinamento do classificador utilizado pelo software, tendo obtido uma acurácia de 87,93% para o nível de espécie e 90,51% para o nível de gênero.

Os dados contidos no repositório NCBI são, em sua maioria, oriundos de museus, herbários, inventários ou repositórios disponíveis na internet tal que cada uma destas fontes realizou coletas em períodos diferentes e com técnicas de amostragens distintas. Em decorrência disto a base apresenta algumas limitações como a não representatividade dos padrões dos dados brutos devido a problemas de viés de amostragem, falta de uma avaliação da representatividade da base de dados quando no que diz respeito à distribuição dos seres em diversas condições geográficas e ambientais. Tais fatores implicam em uma falsa representação das verdadeiras distribuições das espécies (HORTAL; LOBO; JIMÉNEZ-VALVERDE, 2007; SCHULMAN; TOIVONEN; RUOKOLAINEN, 2007; RUETE, 2015)

Hurlbert e Jetz (2007) também afirmam que outro problema encontrado no repositório NCBI é que a base de dados é incompleta. Isso acontece porque existem dados MALDI-TOF que não foram coletados ou não estavam disponíveis durante o processo de construção da base. Outro motivo para a existência de casos com dados faltantes, segundo Hrydziuszko e Viant (2012), é que na etapa de detecção de picos, algumas proteínas podem apresentar valores superiores ao limite pré-estabelecido.

Kim *et al.* (2014) utilizaram dados de proteínas extraídos de espectrometria de massa MALDI-TOF como biomarcadores para a identificação em estágios iniciais de câncer. Nesse intuíto foram aplicadas técnicas de classificação para grupos de controle e discriminação do

[4]Applied Surface Technology Ascend. Disponível em http://www.ascendx.com/.
[5]https://www.ncbi.nlm.nih.gov/

caso de estudo. Para tanto foram utilizados 78 amostras biológicas, onde 55 foram utilizadas para treinamento e 23 para teste. Foram geradas 6 replicatas técnicas para cada uma das 78 amostras biológicas gerando um total de 468 registros. Durante a fase de treinamento do classificador foram encontrados 6.860 dados faltantes, foram testados 3 métodos de imputação, o PPC (*Peak Probability Contrast*), LOESS e MAD (*Median Absolute Deviation*), após este passo foram utilizados 4 técnicas de classificação CART(*Classification And Regression Tree*), *Bagging*, *Random Forest* e Regressão Lasso, essa última que consiste na aplicação da técnica Lasso em conjunto com a regressão logística. A análise de resultado foi feita por meio do erro de predição, onde para a Regressão Lasso foi obtido um erro de 0,1269 com desvio padrão de 0,0949, e o melhor resultado foi para o método *Bagging* com a contraste de probabilidade de pico com um erro de 0,0713 e desvio padrão de 0,0410.

2.5.2 Imputação de dados com o kNN

Lu *et al.* (2011) abordam um caso de estudo para a determinação de estado de doenças em pessoas, para tanto faz o uso do kNN para realizar a imputação de dados. Os autores destacam que em casos de base de dados onde se tem como atributos descritores proteínas que atuam como biomarcadores o uso do kNN deve ser considerado pelo fato desta técnica ser sensível e robusta. Nesta condição, a literatura relata que o número de vizinhos ideal para ser utilizado é entre 10 e 20, utilizando comumente a distância Euclidiana. Foi feita uma eliminação manual de proteínas que continham 40 ou mais dados faltantes, restando 330 proteínas para serem analisadas, para as proteínas restantes foram consideradas duas situações para a imputação, caso a proteína em questão tivesse mais do que 50% de dados faltantes a média era aplicada para a imputação, caso contrário era utilizado o kNN. Para a seleção das proteínas foi utilizado o algoritmo LARS e posteriormente aplicado o *StepWise* com regressão logística que obteve um acerto de 100%.

Waljee *et al.* (2013) destacam que a questão da ausência de dados é algo comum quando se trata de dados laboratoriais médicos, afirmando que a existência de um único método que abranja todas as situações que envolvam imputação deste dados não pode ser feita. Os autores realizaram um estudo a fim de comparar a acurácia e o erro médio de imputação de 4 métodos de imputação em bases laboratoriais médicas de características MCAR, utilizando a linguagem R e realizando testes em duas bases distintas. Os métodos aplicados foram *missforest*, MICE (utilizou o pacote *mice*), imputação com média e imputação com kNN (utilizou o pacote VIM com a função kNN) utilizando 5NN. Para a classificação foram usados dois modelos, regressão logística e *random forest*, sendo criados modelos distintos para dados discretos e contínuos, as bases de teste utilizadas para classificação não possuíam dados faltantes, de

forma que, randomicamente, foram gerados conjuntos de teste com 10%, 20% e 30% de dados faltantes.

A primeira base consistiu em dados obtidos de 446 pacientes da Universidade de Michigan com 21 atributos, dos quais 200 foram destinados para teste, os resultados indicaram que o melhor desempenho foi para o *missforest* tanto em variáveis contínuas como categóricas, MICE, 5NN tiveram desempenho similar em ambas as análises de variáveis e imputação com média, sendo que a acurácia destes testes variaram entre 75% e 94%, os erros de imputação seguiram o que foi apresentado pela acurácia, variando de 15% até 50%, em média. A segunda base também consistiu de dados obtidos da Universidade de Michigan, com dados de 395 pacientes e 26 atributos descritores, desses dados 250 foram destinados para treinamento e 145 para teste. Os resultados indicaram que o *missforest* teve o melhor resultado novamente, sendo seguido por MICE, imputação com média e 5NN. Os valores da acurácia variaram entre 85% e 93%, os erros de imputação para esta base apresentou uma variação média entre 18% e 40%. Foi concluído que o *missforest* teve um alto desempenho em ambas as análises sendo uma boa alternativa para dados laboratoriais médicos.

3 MATERIAIS E MÉTODOS

Este capítulo descreve a sistemática utilizada para avaliar como o desempenho dos classificadores logísticos para identificação de bactérias é afetado pelo emprego do kNN na imputação de dados faltantes. Para tanto, a metodologia proposta faz uso de um conjunto de métricas para aferir o desempenho dos classificadores sobre diferentes condições experimentais e então emprega a análise estatística na interpretação dos resultados. O capítulo também descreve como os resultados obtidos pelo o kNN são comparados com aqueles atingidos por outros métodos de imputação.

Os experimentos realizados dizem respeito ao uso de instâncias de dados que informam aos pesos moleculares de um conjunto de proteínas ribossomais em espectros MALDI-TOF para identificar a espécie/ gênero do objeto associado às mesmas. Neste sentido, foram realizados dois grupos principais de experimentos. O primeiro, pretendia classificar instâncias como pertencentes ou não à espécie *Staphylococcus aureus*. O segundo tinha por objetivo classificar as instâncias com relação ao gênero *Bacillus*.

Os classificadores usados durante os testes implementavam um modelo baseado em regressão logística e foram executados sobre bases de dados *in silico* geradas a partir da *Base de Dados PUKYU* (TOMACHEWSKI, 2017; TOMACHEWSKI *et al.*, 2018; TOMACHEWSKI; GALVÃO; ETTO, 2018). As bases de dados foram pré-processados de acordo com o procedimento descrito na Seção 3.2. A forma do regressor logístico e o algoritmo de treinamento usado em cada um dos experimentos estão especificados na Seção 3.3.

O protocolo experimental proposto (Seção 3.4) estimou o desempenho médio dos classificadores usando um procedimento inspirado na validação cruzada, a Validação Cruzada com Simulação de Dados Incompletos (VCSDI). Como descrito na Seção 3.4.1, o VCSDI permite a execução de algoritmos que simulam de dados faltantes e de algoritmos que realizam imputação de dados à cada iteração da validação cruzada. Durante os experimentos com o VCSDI as seguintes estratégias de imputação foram testadas: (a) valor constante zero; (b) valor constante média e; (c) imputação com kNN. Na execução do algoritmo kNN foram testadas duas diferentes configurações. A primeira configuração aplicou uma função de agregação baseada na média enquanto a segunda usou uma função de agregação baseada na mediana. O desempenho dos classificadores foi medido com os escores RMSE, acurácia, acurácia balanceada, sensibilidade e especificidade.

Os testes realizados em cada experimento estão especificados na Seção 3.5. A Seção 3.6 define os critérios usados na análise dos resultados. As rotinas desenvolvidas durante o

trabalho foram implementadas com a versão 3.4.3 do software R (R Core Team, 2014).

3.1 DESCRIÇÃO DAS BASES DE DADOS USADAS NOS EXPERIMENTOS

As bases de dados utilizadas nos experimentos foram geradas a partir da base de dados criada por Tomachewski *et al.* (2018) e é denominada *Base de Dados PUKYU*. Os atributos da *Base PUKYU* se referem as abcissas de picos em espectros de massa MALDI-TOF que são associados à proteínas ribossomais. Cada registro daquela base informa 60 massas moleculares de proteínas ribossomais e um atributo que descreve a taxonomia da bactéria. Mais especificamente, a base de dados contém os dados referentes as seguintes proteínas: L1, L2, L3, L4, L5, L6, L7, L7a, L7ae, L7.L12, L9, L10, L11, L12, L13, L14, L15, L16, L17, L18, L19, L20, L21, L22, L23, L24, L25, L27, L28, L29, L30, L31, L32, L33, L34, L35, L36, S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S14, S15, S16, S17, S18, S19, S20, S21, S22 e S31e. A *Base de Dados PUKYU* foi obtida a partir do repositório NCBI (Centro Nacional de Informação Biotecnológica, do inglês *National Center for Biotechnology Information*), na data de 13/06/2016, por meio da API(Interface de aplicação a programação, do inglês *Application Programming Interface) E-utilities* disponibilizada pelo repositório. Adicionalmente, a base *Ribopeaks* assume que as leituras do espectro de massa pertencem ao intervalo entre 0 e 40kDa.

A fim de desenvolver os classificadores dicotômicos usados na execução dos experimentos deste trabalho, foram criadas duas novas bases de dados a partir da *Base de Dados PUKYU*. A primeira base de dados, denominada *Base Bacillus*, armazena dados para identificação de indivíduos do gênero *Bacillus*. A segunda base de dados, denominada *Base S. aureus*, é formada por instâncias relacionadas à identificação de bactérias da espécie *Staphylococcus aureus*. Além dos atributos descritos anteriormente cada uma das bases possuí um atributo binário, $TaxNum$, que indica para cada caso se o mesmo pertence à classe de interesse. Desta forma, na *Base S. aureus*, $TaxNum = 1$ indica que o caso foi pré-classificado como pertencente à espécie *Staphylococcus aureus* (gênero *Bacillus*) e $TaxNum = 0$ informa que o caso não é uma instância daquela classe. A interpretação do atributo $TaxNum$ na base *Base Bacillus* é análoga. A Tabela 2 apresenta um exemplo de registro presente na *Base Bacillus*, onde *NA* indica dados faltantes.

Tomachewski (2017) destaca que em decorrência da grande variação do número de registros nas diferentes de estirpes presentes no NCBI, a base *PUKYU* é desbalanceada. Como resultado deste desbalanceamento, a *Base Bacillus* possuí 552 registros do gênero *Bacillus* e 31474 registros relacionados a gêneros diferentes de *Bacillus*. Para a *Base S. aureus*, a espécie com maior número de registros na *Base de Dados PUKYU*, existem 4064 casos da espécie

Tabela 2: Exemplo de instância extraído da Base Bacillus

L1	L2	L3	L4	L5	L6	L7	L7a
24447,31	29992,54	22641,04	22337,64	20133,56	19450,29	NA	NA
L7ae	L7.L12	L9	L10	L11	L12	L13	L14
10840,4	12279,05	16616,24	17442,33	14900,56	NA	16304,77	13222,36
L15	L16	L17	L18	L19	L20	L21	L22
15526,75	16328,1	14058,18	13153,2	12908,11	13420,95	11198,05	12752,84
L23	L24	L25	L27	L28	L29	L30	L31
11092,86	10909,7	NA	10236,76	6908,14	7875,23	6427,57	9544,59
L32	L33	L34	L35	L36	S1	S2	S3
NA	6103,17	5494,64	7508,74	NA	NA	28629,81	24130,83
S4	S5	S6	S7	S8	S9	S10	S11
22867,01	17372,16	11588,13	17859,77	14286,51	14136,37	11480,41	14063,29
S12	S13	S14	S15	S16	S17	S18	S19
15123,6	13545,73	NA	10483,07	9718,33	10304,98	9415,09	10561,26
S20	S21	S22	S31e	TaxNum			
9257,62	6768,81	NA	NA	0			

Staphylococcus aureus e 27962 registros para as demais espécies. Vale ressaltar que a *Base Bacillus* possuí registros de bactérias que estão relacionadas ao crescimento vegetal e bactérias que causam doenças em humanos.

A principal motivação para os testes com o gênero *Bacillus* é o fato deste gênero estar relacionado ao crescimento vegetal, também estando relacionado a doenças em humano. Os testes com a *Staphylococcus aureus* foram movidos pela sua importância no que se refere a saúde alimentar, por exemplo, em estudos sobre intoxicação alimentar (ARGUDÍN; MENDOZA; RODICIO, 2010) e uso de plantas medicinais Michelin *et al.* (2005), Souza *et al.* (2015), Miranda *et al.* (2015).

3.2 PRÉ-PROCESSAMENTO DAS BASES DE DADOS

O primeiro passo da etapa de pré-processamento submeteu as bases *Base Bacillus* e *Base S. aureus* a um procedimento de seleção de atributos cujo o objetivo foi definir quais proteínas eram relevantes para a predição da espécie/gênero em estudo. Durante sua execução, a classe alvo de cada base de dados foi denotada pelo rótulo $Y = 1$ e a classe alternativa por $Y = 0$. Para tanto foi executada uma rotina que calculou o número de casos faltantes de cada

atributo e verificou se o mesmo estava acima de um limiar, esse que define a quantidade de instâncias que um determinado atributo deve apresentar para não ser retirado nesta etapa, tendo esse valor sido obtido de maneira empírica. Quando o limitar não era atingido, a proteína era removida da base de treinamento. O procedimento descrito foi executado separadamente para os casos em que $Y = 0$ e $Y = 1$ e teve o propósito de mitigar desbalanceamento dos dados ao evitar a escolha de atributos com um número de casos muito pequeno.

A Tabela 3 enumera o limiar que foi estabelecido de maneira experimental para cada base de dados e as proteínas removidas em relação as classes 0 e 1.

Tabela 3: Proteínas excluídas com base nas classes

Bactéria	Limiar	Proteínas excluídas	
		Y = 1	Y = 0
Bacillus	250	L7, L7a, L7ae, L12, S1, S22 e S31e	S31e
S. aureus	100	L7, L12, S1, S22 e S31e	S31e

Após a aplicação do procedimento descrito acima, as duas bases foram submetidas a um processo de imputação de dados utilizando a média. Este procedimento de imputação foi utilizado apenas durante a seleção de atributos e teve a finalidade de contornar o problema do algoritmo de seleção *StepAIC* não tratar dados faltantes. Na sequência executou-se, para cada base de dados, uma análise de correlação de *pearson* entre os atributos a fim de remover descritores linearmente relacionados. Basicamente, a correlação ρ de cada par de atributos X_1 e X_2 foi calculada e sempre que $|\rho| > 0.7$ um dos atributos era removido, esse valor foi utilizado pois já indica uma correlação forte entre os atributos. Como nenhum par de atributos teve correlação superior a 0,7; nenhuma variável foi excluída das bases de dados neste procedimento.

Na sequência, as bases foram processadas pelo algoritmo de seleção *wrapper StepAIC*, descrito na Seção 2.2.2. O Algoritmo 2 ilustra a rotina de seleção de variáveis. Como pode ser visto, o algoritmo recebe como entrada a base de dados **D** completada no passo anterior e gera um modelo baseado em regressão logística, chamado modeloGLM. Em seguida, executa o stepAIC e o melhor conjunto de atributos encontrado é retornado pela variável *atributos*. Esta rotina foi executada uma vez para cada base.

A função *glm*, chamada no algoritmo, permite a construção de Modelos Lineares Generalizados (GLM, do inglês *Generalized Linear Models*) e possuí três parâmetros. O parâmetro *família*, configurado como *binomial(link = "logit")*, estabelece que será utilizada uma regressão logística na geração do modelo. O parâmetro *dados*, base de treinamento, foi definido como o subconjunto de dados usado em cada teste e o parâmetro *fórmula* define a equação a ser ajustada para cada base. Esta função faz parte do pacote *stats* presente de maneira nativa no R.

A implementação do stepAIC usada nos teste foi aquela disponibilizada pelo pacote *MASS*[1]. Esta função possuí dois parâmetros de entrada, o *modeloGLM* e a *direção* da busca. O modeloGLM foi especificado como o modelo criado no primeiro passo e a direção foi definida como *both*, o que indica que o será avaliado tanto o caso *forward*, como o *backward*.

Algoritmo 2: EXECUÇÃO DO ALGORITMO STEPAIC

Entrada: Base de dados imputada com valores médios (completa) **D**
Saída: Atributos selecionados pelo StepAIC
1 **início**
2 modeloGLM = glm(formula,familia,dados)
3 atributos = stepAIC(modeloGLM,direção)
4 **fim**
5 **retorna** *atributos*

Por fim, em cada base de dados, o comportamento de cada atributo foi comparado considerando as instâncias em que $Y = 0$ e $Y = 1$. Para tanto foi realizado um teste T de Student não pareado com $\alpha = 0,05$ que mostrou que as médias das leituras MALDI-TOF das proteínas L1 e S10 não apresentavam diferença significativa entre as classes no caso da *Base Bacillus*. Em decorrência deste resultado estes atributos foram removidos da base.

Após a seleção de atributos as seguintes proteínas foram escolhidas para compor a *Base Bacillus*, L2, L3, L4, L5, L6, L13, L16, L17, L19, L21, L23, L24, L28, L29, L31, L32, L33, L34, L35, S2, S3, S4, S6, S7, S8, S9, S12, S13, S16, S17, S18, S20, S21. Para a *Base S. aureus* foram selecionadas as proteínas L1, L3, L4, L6, L7a, L7.L12, L9, L10, L11, L14, L16, L17, L18, L19, L20, L21, L22, L23, L24, L25, L27, L28, L29, L30, L31, L32, L33, L35, L36, S2, S3, S4, S6, S8, S9, S13, S14, S16, S17, S19, S20, S21. Após a seleção de atributos as bases originais, sem imputação, foram retomadas, sendo então retirados os registros que continham dados faltantes. As cardinalidades das bases de dados resultantes do processo de seleção são listadas na Tabela 4.

Tabela 4: Cardinalidades das bases

Base	Nº Atributos	Nº Registros Y = 1	Nº Registros Y = 0	Total de Registros
Bacillus	33	212	8635	8847
S. aureus	42	3094	118	3212

[1]Disponível em https://cran.r-project.org/web/packages/MASS/index.html

3.3 FUNÇÃO DE CLASSIFICAÇÃO

Durante a execução da função *glm*, descrita na Seção anterior, os parâmetros do modelo (**B**) foram calculados, conforme o processo descrito na Seção 2.4.1, a função no R, GLM, que gera um modelo de Regressão Logística é apresentada pela Equação 1.

$$glm(formula = TaxNum \sim ., family = binomial(link = "logit"), \\ data = baseTreinamento) \tag{1}$$

Para a *Base Bacillus*, que possuí 33 atributos descritos, a fórmula ajustada para criação do modelo de regressão logística é aquela mostrada na Equação 2. Naquela expressão, cada x_i, com $i = 1..33$ indica um atributo descritor de acordo com a seguinte ordem: L2, L3, L4, L5, L6, L13, L16, L17, L19, L21, L23, L24, L28, L29, L31, L32, L33, L34, L35, S2, S3, S4, S6, S7, S8, S9, S12, S13, S16, S17, S18, S20, S21.

$$P(Y = 1|\mathbf{x}_i) = \frac{e^{\beta_0+\beta_1x_1+\beta_2x_2+\beta_3x_3...+\beta_{33}x_{33}}}{1 + e^{\beta_0+\beta_1x_1+\beta_2x_2+\beta_3x_3...+\beta_{33}x_{33}}} \tag{2}$$

A Equação 3 mostra a função de ajuste para a *Base S. aureus*. Mais uma vez, x_i, $i = 1..42$, indica uma proteína da sequência L1, L3, L4, L6, L7a, L7.L12, L9, L10, L11, L14, L16, L17, L18, L19, L20, L21, L22, L23, L24, L25, L27, L28, L29, L30, L31, L32, L33, L35, L36, S2, S3, S4, S6, S8, S9, S13, S14, S16, S17, S19, S20, S21.

$$P(Y = 1|\mathbf{x}_i) = \frac{e^{\beta_0+\beta_1x_1+\beta_2x_2+\beta_3x_3...+\beta_{42}x_{42}}}{1 + e^{\beta_0+\beta_1x_1+\beta_2x_2+\beta_3x_3...+\beta_{42}x_{42}}} \tag{3}$$

3.4 PROTOCOLO EXPERIMENTAL

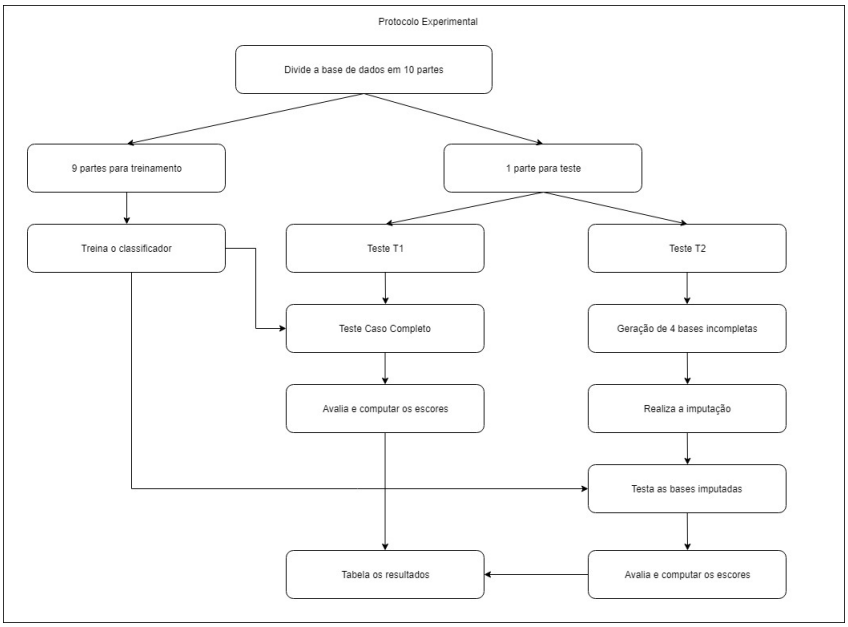
A influência do uso dos diferentes métodos de imputação baseados no kNN sobre o desempenho dos classificadores foi avaliada com a realização de dois experimentos em cada base de dados. No primeiro experimento, o desempenho do classificador foi mensurado em ensaios em que todos os casos a serem classificados tinham o mesmo número de variáveis faltantes. No segundo experimento, o número de dados faltantes em cada caso foi determinado por um procedimento onde cada instância variava de 1 até 8 atributos faltantes.

Durante os experimentos, o procedimento Validação Cruzada com Simulação de Dados Incompletos (VCSDI) (descrito na Seção 3.4.1) foi usado para mensurar o desempenho do

classificador logístico em testes que envolviam bases de dados completas e incompletas. A medição do desempenho foi realizada assumindo a imputação de dados com os algoritmos kNN, imputação com a média e imputação com zero. Os escores usados para medir o desempenho foram: acurácia, acurácia balanceada, especificidade, sensibilidade e RMSE considerando testes em bases completas e incompletas.

A Figura 5 apresenta uma visão geral da metodologia empregada neste trabalho, sendo essa explorada em cada uma das seções deste capítulo.

Figura 5: Protocolo experimental



3.4.1 Validação Cruzada com Simulação de Dados Incompletos (VCSDI)

O Algoritmo 3 mostra o procedimento VCSDI. Neste procedimento, a base de dados $\mathbf{D}$ é dividida em 10 partições. O particionamento é efetuado por um procedimento de amostragem aleatória estratificada (KIM *et al.*, 2013) como implementado pela função *createFolds* do pacote *caret*[2] e que possibilita a criação de partições sem a repetição de elementos (HYNDMAN; ATHANASOPOULOS, 2014).

Em seguida, o algoritmo inicializa as variáveis r_C, r_0 r_m e r_{kNN}. A variável r_C armazena os resultados de cada iteração da validação cruzada para o caso em que a base de teste está completa. As demais variáveis guardam os resultados obtidos quando o tratamento dos dados faltantes é realizado pelos algoritmo de imputação com zero, imputação com a média e imputação com kNN, respectivamente.

[2]Disponível em https://cran.r-project.org/web/packages/caret/.

Algoritmo 3: VCSDI para EXPERIMENTOS DE IMPUTAÇÃO

Entrada: Base de dados **D**, tamanho para validação cruzada **b**, Número de vizinhos **k**, função de agregação *h(.)*, $\mathbf{N_v}$ número variável de dados faltantes
Saída: Resultados dos escores médios de cada execução

1 **início**
2 **kFolds =** Particiona **D** em **b** partes;
3 Cria tabelas de resultados r_C, r_0 r_m e r_{kNN} vazias
4 **para** *cada* $\mathbf{T}_b$ *de* ***kFolds*** **faça**
(a) $\mathbf{T}_b$ é destinada para base de teste e os demais registros são destinados para base de treinamento;
(b) Treinar o classificador com a base de treinamento;
(c) Dividir a base de teste $\mathbf{T}_b$ pela metade, criando $\mathbf{T}_{b1}$ e $\mathbf{T}_{b2}$;
(d) Realizar a predição e calcular os escores para a $\mathbf{T}_{b2}$ e armazenar os resultados em r_C;
(e) A partir da $\mathbf{T}_{b2}$ gerar a lista *lbdi*, que armazena 4 bases contendo dados incompletos em diferentes proporções, com as seguintes configurações 1 a 3 variáveis faltantes por instância, e utilizando o número de valores variável de dados faltantes informado por $\mathbf{N_v}$;
(f) **para** *cada base em lbdi* **faça**
para *cada registro da base* **faça**
(i) Imputar os valores faltantes com zero, executar predição e calcular os escores
(ii) Imputar atributos com valores faltantes com a média do atributo faltante, executar predição e calcular os escores
(iii) Imputar os valores faltantes com kNN utilizando o **k** e *h(.)* informados, sendo que a função *h(.)* utiliza $\mathbf{T}_{b1}$ para estimação, executar predição ao final são calculados os escores
fim
fim
5 **fim**
6 **fim**
7 **retorna** *Retorna uma lista contendo os resultados de todas as execuções*

Na sequência, o algoritmo executa laço de repetição do passo 4. À cada iteração deste laço uma das partições de $\mathbf{D}$ é utilizada como conjunto de teste do classificador. As demais partições compõem o conjunto de treinamento. A partição de teste é selecionada no passo 4 (a) e é denotada por $\mathbf{T}_b$, $b = 1..10$. As demais partições são reunidas em um conjunto de dados $\mathbf{D}_b$. Na sequência, $\mathbf{D}_b$ é usada para treinar o classificador logístico (passo 4 (b)) e obter os modelos de classificação descritos na Seção 3.3.

No passo 4 (c) a partição de teste, $\mathbf{T}_b$, é dividida em duas partes, $\mathbf{T}_{b1}$ e $\mathbf{T}_{b2}$. $\mathbf{T}_{b2}$ é utilizada no passo 4 (d) para realizar o teste do regressor logístico com o modelo criado no passo 4(b) para a predição com dados completos salvando os resultado em r_C, Posteriormente, $\mathbf{T}_{b2}$ é submetida a um procedimento que utiliza o algoritmo (MEEYAI, 2016) para simular a ocorrência de dados faltantes. Este procedimento, isto é apresentado no passo 4(e), gera duas séries de experimentos. A primeira com um número fixo de dados faltantes por instância. A segunda gera instâncias com um número variável de dados faltantes.

A geração de bases de dados contendo números fixos de atributos faltantes por ins-

tância faz uso da função *ampute* presente no pacote *mice*[3]. Esta função tem como entrada um parâmetro u que estipula a quantidade de dados faltantes por instância. Como $u = 1, 2, 3$ são criadas três bases de dados com dados faltantes. Estas bases são então adicionadas à lista *lbdi* (Lista de Bases de Dados Incompletas).

Para a criação de bases de testes com um número variável de valores faltantes por instância o VCSDI gera, para cada instância de $\mathbf{T}_{b2}$, um número aleatório g com base na função *runif* do pacote base da linguagem R. Essa função simula valores aleatórios segundo uma distribuição uniforme e foi usada para determinar a quantidade de atributos faltantes que uma instância de teste teria. A chamada a *runif* é configurada de forma que cada instância tem entre 1 e $\mathbf{N_v}$ atributos faltantes; neste trabalho $\mathbf{N_v} = 8$.

Após a determinação de k a função *sample.int* é chamada. Esta função recebe como parâmetros o total de colunas da base de dados $\mathbf{D}$ e o número inteiro k. Como resultado, ela retorna um vetor de k elementos cujos valores simbolizam os índices das colunas que devem ter suas observações removidas do caso selecionado. Ao final do passo 4(e) a base criada tem atributos falantes e é adicionada a variável *lbdi*, totalizando 4 bases armazenadas em *lbdi*.

Após a geração das 4 bases para imputação, o VCSDI executa dois laços de repetição. O primeiro para percorrer cada um das 4 bases criadas e o segundo para percorrer os registros de cada uma das bases, sendo isto descrito na passo 4(f). Para cada registro foram executadas as 4 abordagens de imputação, imputação com zero, etapa 4(f)(i), imputação com valores médios dos atributos faltantes, etapa 4(f)(ii), e com kNN que contou com a utilização de 2 funções de agregação, $h(.)$, média e mediana, na etapa etapa 4(f)(iii). As abordagens com média e kNN fizeram uso a partição $\mathbf{T}_{K1}$ para a estimação dos valores faltantes. Para cada algoritmo é chamado o método *predict* para realizar a predição de qual classe instância pertence, e o final das execuções os resultados da classificação foram salvos nas variáveis r_0 r_m e r_{kNN}. Posteriormente é realizada a análise estatísticas dos resultados.

3.4.2 Seleção do valor de *k* para execução do algoritmo kNN

Segundo Hassanat *et al.* (2014), uma forma usual de escolher o parâmetro k do algoritmo de imputação com o kNN é proceder testes empíricos. Assim para determinação da configuração do kNN durante a execução do VCSDI, aquele algoritmo foi executado sobre a base incompleta $\mathbf{D_0}$ usando valores de k entre 3 e 30. Os dados imputados foram classificados e o procedimento de teste avaliou o desempenho do classificador em termos de acurácia balanceada.

[3]Disponível em https://cran.r-project.org/web/packages/mice/.

Ao final deste processo foi escolhido o valor que maximizou o desempenho do modelo. Isto resultou em $k = 3$ para a função de agregação de média e $k = 5$ para mediana na base *Base Bacillus*. Para a *Base S. aureus* foi utilizado $k = 3$ para as duas funções de agregação.

3.5 TESTES REALIZADOS

O protocolo experimental descrito na Seção 3.4 foi repetido em uma série de testes sobre as duas bases descritas na Seção 3.1. Os testes realizados com as *Base Bacillus* e *Base S. aureus* estão listados nas Tabelas 5 e 6, respectivamente. Como especificado no protocolo, em todos os experimentos foram computadas as médias dos escores de acurácia balanceada, acurácia, RMSE, sensibilidade e especificidade obtidos pela regressão logística. Para a realização do cálculo da sensibilidade e especificidade, foi considerada $Y = 1$ como a classe positiva e $Y = 0$ como classe negativa, isto é, a sensibilidade expressa a taxa de acerto dos registros do gênero *Bacillus* (espécie *S. aureus*) e a especificidade a taxa de acerto dos registros não pertencentes ao gênero *Bacillus* (espécie *S. aureus*).

Tabela 5: Experimentos para a Base Bacillus

Experimento	Método	Critério
EBac.Co	Completo	Caso completo
EBac.IZ1	Imputação com Zero	1 variável faltante por instância
EBac.IZ2	Imputação com Zero	2 variáveis faltantes por instância
EBac.IZ3	Imputação com Zero	3 variáveis faltantes por instância
EBac.IZNv	Imputação com Zero	Número variável de dados faltantes
EBac.IM1	Imputação com Média	1 variável faltante por instância
EBac.IM2	Imputação com Média	2 variáveis faltantes por instância
EBac.IM3	Imputação com Média	3 variáveis faltantes por instância
EBac.IMNv	Imputação com Média	Número variável de dados faltantes
EBac.kM1	Imputação com kNN - Média	1 variável faltante por instância
EBac.kM2	Imputação com kNN - Média	2 variáveis faltantes por instância
EBac.kM3	Imputação com kNN - Média	3 variáveis faltantes por instância
EBac.kMNv	Imputação com kNN - Média	Número variável de dados faltantes
EBac.kMd1	Imputação com kNN - Mediana	1 variável faltante por instância
EBac.kMd2	Imputação com kNN - Mediana	2 variáveis faltantes por instância
EBac.kMd3	Imputação com kNN - Mediana	3 variáveis faltantes por instância
EBac.kMdNv	Imputação com kNN - Mediana	Número variável de dados faltantes

Nestas tabelas, os testes dos experimentos com um número constante de variáveis faltantes foram identificados por rótulos da forma E<base><método><r>, em que E identifica o experimento, <base> indica sobre qual base de dados o teste foi realizado, <método> o procedimento de imputação usado e r é o número de variáveis faltantes. O prefixo <base> pode assumir um de dois valores: Bac (indicando que a base considerada é a *Bacillus*) e St (indicando a Base *S. aureus*). O termo <método> informa o procedimento de imputação usado, os casos possíveis são, *Co*, caso completo, *IZ*, imputação com zero, *IM*, imputação com a média, *kM*, kNN com a função de agregação de média, e *kMd*, kNN com função de agregação de mediana.

Tabela 6: Experimentos para a Base S. aureus

Experimento	Método	Critério
ESt.Co	Completo	Caso completo
ESt.IZ1	Imputação com Zero	1 variável faltante por instância
ESt.IZ2	Imputação com Zero	2 variáveis faltantes por instância
ESt.IZ3	Imputação com Zero	3 variáveis faltantes por instância
ESt.IZNv	Imputação com Zero	Número variável de dados faltantes
ESt.IM1	Imputação com Média	1 variável faltante por instância
ESt.IM2	Imputação com Média	2 variáveis faltantes por instância
ESt.IM3	Imputação com Média	3 variáveis faltantes por instância
ESt.IMNv	Imputação com Média	Número variável de dados faltantes
ESt.kM1	Imputação com kNN - Média	1 variável faltante por instância
ESt.kM2	Imputação com kNN - Média	2 variáveis faltantes por instância
ESt.kM3	Imputação com kNN - Média	3 variáveis faltantes por instância
ESt.kMNv	Imputação com kNN - Média	Número variável de dados faltantes
ESt.kMd1	Imputação com kNN - Mediana	1 variável faltante por instância
ESt.kMd2	Imputação com kNN - Mediana	2 variáveis faltantes por instância
ESt.kMd3	Imputação com kNN - Mediana	3 variáveis faltantes por instância
ESt.kMdNv	Imputação com kNN - Mediana	Número variável de dados faltantes

O termo $r \in \{1, 2, 3, Nv\}$ informa o número de variáveis faltantes em cada caso, de forma que o valor *Nv* indica um número variável de dados faltantes por instância. O método Completo (*Co*) consistiu no caso onde a $\mathbf{T}_{b2}$ foi utilizada para testar e mensurar o classificador (linha 4(d) pelo Algoritmo 3).

3.6 ANÁLISE DOS RESULTADOS

Após a execução de todos os experimentos que geraram 10 resultados cada foi executada uma análise estatística por meio do teste de Friedman com *post-hoc* (FRIEDMAN, 1937; BORTZ; LIENERT; KLAUS, 2008) para verificar se houve diferença significativa entre o escores do caso completo em relação aos escores para os casos imputados. O *teste T de Student* (MANKIEWICZ, 2000) foi utilizado para determinar quais combinações, quando aumentado o número de atributos faltantes, para o mesmo algoritmo de imputação houve uma diferença estatística identifica. Ambos os testes consideraram um α de 0,05.

Durante a discussão foi realizada uma análise dos erros de imputação e sua influência nos erros de predição, para o caso em que o número de atributos faltantes era variável. Também foi aplicado um teste T unicaudal entre as abordagens do kNN e média. Ao final foi realizada uma análise das correlações entre o número de variáveis imputadas com o erro médio de imputação e o erro de predição, para verificar se com o aumento do número de atributos faltantes houve aumento do erro de imputação e/ou predição.

4 RESULTADOS E DISCUSSÃO

Este capítulo apresenta, nas seções 4.1 e 4.2, os resultados dos experimentos descritos no Capítulo 3. Para cada um dos experimentos é apresentado o resultado da análise do desempenho do algoritmo classificador para as estratégias de imputação testadas. Durante a análise, foram utilizados testes estatísticos a fim de avaliar as diferenças de desempenho dos classificadores, dadas as abordagens de imputação utilizadas. Também foi analisado o efeito do número de variáveis faltantes. A Seção 4.3 apresenta a discussão. Além disto, aquela seção discute o relacionamento entre o erro de imputação e o número de variáveis faltantes em caso.

4.1 RESULTADO DE CLASSIFICAÇÃO COM IMPUTAÇÃO DE DADOS PARA O GÊNERO *BACILLUS*

4.1.1 Resultados do experimento para o caso completo

A Tabela 7 mostra os resultados do experimento de classificação de bactérias na *base Bacillus*, a partir de dados completos (EBac.Co). Nestas condições, o classificador obteve uma acurácia de 99% e um RMSE inferior a 0,1 (com desvio de 0,01). Se for assumido que a distribuição dos erro quadrático é gaussiana, a probabilidade deste classificador apresentar um valor superior a 0,25 para este erro é zero (no limite).

Tabela 7: Resultado da classificação para o gênero *Bacillus* considerando o caso em que base é completa

Experimento	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
EBac.Co	0,9350	0,0522	0,8754	0,1042	0,9946	0,0028	0,9918	0,0040	0,0766	0,0195

Uma vez que a *Base Bacillus* é desbalanceada (ver Seção 3.2) é apropriado usar os escores da acurácia balanceada, especificidade e sensibilidade para avaliar o desempenho do classificador em substituição à acurácia. O desbalanceamento dos casos desta base tende para a classe $Y = 0$ (98% das amostras não pertencem ao gênero *Bacillus*) e, como visto na tabela, a especificidade do classificador ficou próxima de 100%. Este resultado está em acordo com a literatura que relata que, em bases desbalanceadas, o modelo tende a ser mais preciso na predição da classe majoritária (PHUNG; BOUZERDOUM; NGUYEN, 2009). Na aplicação em questão, este valor indica que o classificador raramente identifica como *Bacillus* um objeto que não pertence a esta categoria. Por outro lado, o resultado da sensibilidade, superior a 87%, mostra que apesar da classe minoritária ($Y = 1$) representar apenas 2% da base completa, o modelo gerado foi capaz de identificar corretamente a maioria dos casos que pertenciam ao gênero *Bacillus*.

4.1.2 Resultados dos experimentos com imputação de um número constante de valores

As Tabelas 8, 9 e 10 relacionam os resultados dos experimentos em bases de teste cujos casos tinham uma, duas ou três variáveis não observadas. A simulação dos dados faltantes e a imputação foram realizadas conforme os experimentos descritos na Tabela 5 do Capítulo 3.

Tabela 8: Resultados da classificação para o gênero *Bacillus* com caso completo e com 1 variável faltante por instância

Experimento	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
EBac.Co	0,9350	0,0522	0,8754	0,1042	0,9946	0,0028	0,9918	0,0040	0,0766	0,0195
EBac.IZ1	0,6656	0,0921	0,6208	0,1769	0,7105	0,0335	0,7084	0,0339	0,5293	0,0298
EBac.IM1	0,9183	0,0461	0,8404	0,0912	0,9962	0,0019	0,9925	0,0037	0,0755	0,0208
EBac.kM1	0,9350	0,0525	0,8754	0,1042	0,9946	0,0030	0,9918	0,0044	0,0748	0,0184
EBac.kMd1	0,9350	0,0525	0,8754	0,1042	0,9946	0,0028	0,9918	0,0040	0,0757	0,0190

Na Tabela 8, o único teste em que a especificidade ficou abaixo de 0,99, em comparação ao experimento EBac.Co foi o da imputação com zero (EBac.IZ1); média de 0,7105. Essa queda de desempenho de aproximadamente 28% resultou em uma diferença significativa, isso é, apresentou um valor inferior a 0,05 quando aplicado o teste de Friedman com *post-hoc*. Além disso, o *post-hoc* do teste de Friedman também apontou uma diferença significativa na especificidade do classificador quando se comparou a imputação com zero em relação à imputação com a média e com o kNN.

Os classificadores testados apresentaram uma queda de desempenho, em relação ao caso completo, em termos de sensibilidade quando da imputação com zero e com a média. A maior redução foi observada na imputação com zero, 25%. Este resultado é confirmado pelo teste de Friedman, que detectou uma diferença estatística significativa na sensibilidade do classificador quando se compara a imputação com zero e o caso completo. A imputação com a média levou a uma queda de 3% na sensibilidade, contudo, isto não acarretou uma diferença estatística quando se compara seus resultados aos da classificação. Por outro lado, a imputação com o kNN teve um desempenho idêntico aos obtidos na base de dados completa. O teste de Friedman também mostra que a sensibilidade do classificador em bases dados imputadas com zero foi estatisticamente diferente da sensibilidade observada no processamento de dados completos, e aos demais métodos de imputação.

A Tabela 8 também mostra uma queda na acurácia balanceada para os experimentos EBac.IZ1 e EBac.IM1 em relação ao caso completo. Este resultado, queda de 27% e 2% para a imputação com zero e imputação com a média, respectivamente, reflete o fato desta medida ser a média aritmética da especificidade e a sensibilidade. Mais uma vez, o teste de Friedman mostra

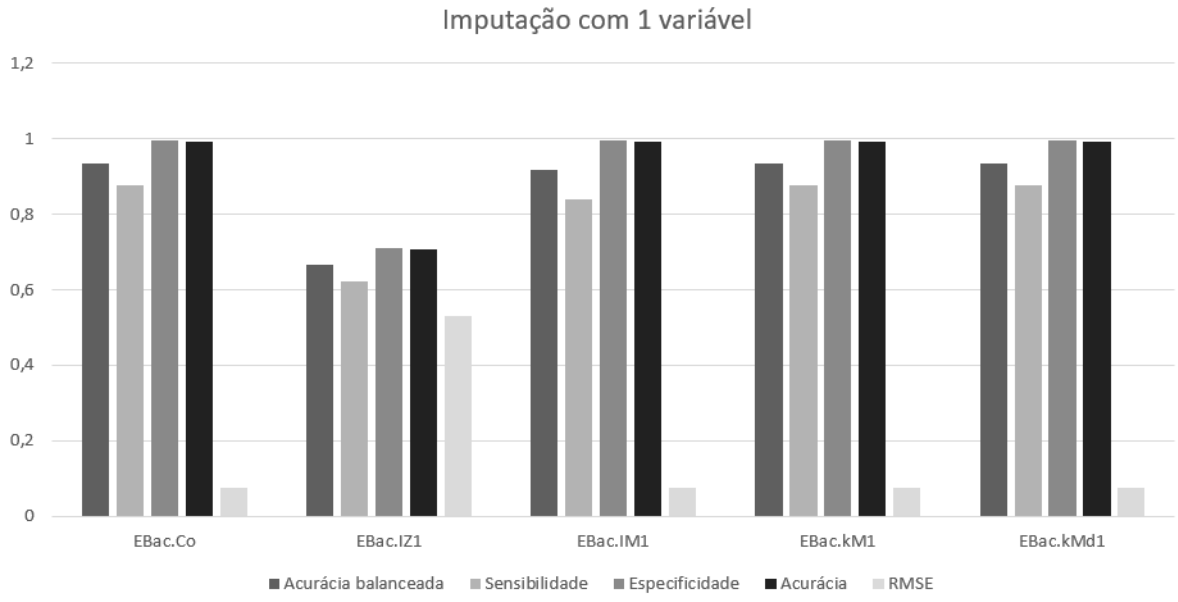
que os resultados permitem concluir que o desempenho do classificador diminuí significativamente quando se compara a classificação com dados completos e a classificação com dados submetidos à imputação com zero. Mais ainda, o teste de *post-hoc* para a *acBal* indicou que as diferenças apareceram em todas as comparações que envolveram o experimento EBac.IZ1.

A comparação da acurácia balanceada do classificador em dados imputados com a média e aquela obtida no caso completo retornou um *p-value* superior a 0,9. Já a imputação com kNN (ambas as abordagens) apresentou um *p-value* de 0,99. Estes valores fornecem uma evidência estatística para afirmar que o *acBal* do classificador não foi substancialmente diferente nos testes EBac.Co, EBac.IM1, EBac.kM e EBac.kMd. Arsham (1988) afirma que *p-values* acima de 0,10 já indicam a não existência de diferença significativa entre os valores apresentados.

Os testes de classificação também mostraram uma queda no desempenho em termos RMSE sempre que se usou a imputação com zero. Para aquele método, a ocorrência de um atributo faltante, multiplicou o RMSE em mais de seis vezes. Para as demais técnicas de imputação, o RMSE também foi alterado, porém não tão expressivamente. Na imputação com média e kNN o RMSE teve um decréscimo inferior a 1%. A aplicação do teste de hipóteses indicou a diferença estatística entre os resultados, onde o *post-hoc* indicou a ocorrência de *p-values* inferiores a 0,05 para todos os pareamentos que envolveram a imputação com zero.

Os resultados da Tabela 8 são ilustrados na Figura 6.

Figura 6: Resultados médios da classificação na imputação de uma variável faltante



Do exposto, é possível concluir que os resultados da Tabela 8, em relação ao desempenho do classificador logístico na presença de um atributo faltante por instância apresentaram

as seguintes interpretações:

- A imputação com kNN e com média não apresentaram diferença estatística em nenhum escore para o caso completo;
- O uso da imputação com zero acarretou uma redução significativa de desempenho em relação ao caso completo e aos demais métodos de imputação (para todos os escores);
- A imputação com a média apresentou um queda em relação a *acBal* e a sensibilidade, porém seu desempenho não se mostrou estatisticamente diferente do caso completo.

Tabela 9: Resultados da classificação para o gênero *Bacillus*, com caso completo e contendo uma variável faltante e com duas variáveis faltantes por instância

Experimento	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
EBac.Co	0,9350	0,0522	0,8754	0,1042	0,9946	0,0028	0,9918	0,004	0,0766	0,0195
EBac.IZ1	0,6656	0,0921	0,6208	0,1769	0,7105	0,0335	0,7084	0,0339	0,5293	0,0298
EBac.IM1	0,9183	0,0461	0,8404	0,0912	0,9962	0,0019	0,9925	0,0037	0,0755	0,0208
EBac.kM1	0,9350	0,0525	0,8754	0,1042	0,9946	0,0030	0,9918	0,0044	0,0748	0,0184
EBac.kMd1	0,9350	0,0525	0,8754	0,1042	0,9946	0,0028	0,9918	0,0040	0,0757	0,0190
EBac.IZ2	0,6384	0,0653	0,6514	0,1388	0,6054	0,0244	0,6067	0,0229	0,6202	0,0170
EBac.IM2	0,8524	0,0637	0,7088	0,1276	0,996	0,0024	0,9893	0,0046	0,0864	0,0204
EBac.kM2	0,9367	0,0588	0,8784	0,1172	0,9951	0,0027	0,9923	0,0045	0,0749	0,0209
EBac.kMd2	0,9365	0,0586	0,8784	0,1172	0,9946	0,0026	0,9918	0,0042	0,0775	0,0204

Na Tabela 9 é possível notar que a ocorrência de dois atributos faltantes por instância manteve a tendência de queda da especificidade do classificador quando da imputação com zero. Uma comparação com os resultados do caso completo (EBac.Co) evidencia que quando a imputação com zero foi usada houve uma redução de 38% na especificidade. Por sua vez, o emprego da imputação com a média e com kNN (função com média) levou a um incremento da especificidade, de 0,14% e 0,05%, respectivamente. Apesar destas desigualdades numéricas, o único método para o qual teste de *post-hoc* encontrou uma diferença estatística, em relação ao caso completo, foi a imputação com zero (EBac.IZ2). Além disso, este ensaio (EBac.IZ2) obteve *p-values* inferiores a 0,05 quando comparado aos demais métodos de imputação.

A Tabela 9 também mostra uma queda na especificidade do classificador quando observado quando se compara os resultados dos testes EBac.IZ1 e EBac.IZ2 da imputação com zero. A queda da especificidade observada foi de 11% e está associada a uma diferença estatística significativa nos resultados de cada teste (*p-value* = 0,0001). Para os demais algoritmos, o teste T não indicou que o incremento no número de variáveis imputadas tenha causado uma diferença significativa na especificidade do classificador.

A sensibilidade dos modelos de classificação com duas variáveis faltantes também apresentou uma redução, em comparação ao caso completo. A maior redução ocorreu na imputação com zero (EBac.IZ2)- 22% - seguida pela imputação com média (EBac.IM2) - 16%. As diferenças da sensibilidade da imputação com zero e da imputação com média em relação à base completa se mostraram estatisticamente significantes; *p-values* de 0,007 e 0,005, respectivamente. Na imputação com kNN, agregação pela média/mediana (EBac.kM2 e EBac.kMd2), houve um aumento de cerca 0,3%. Assim, a imputação com kNN (ambas abordagens) produziu escores superiores ao do caso completo, mas não estatísticamente diferentes, por outro lado se mostraram estatisticamente diferentes das imputações com média e com zero, tendo *p-values* de 0,005 e 0,004, respectivamente.

Quando se compara, para um mesmo algoritmo, a sensibilidade nos casos em que houve imputação de uma ou duas variáveis Tabela 9, nota-se uma perda de 13% do desempenho na imputação com a média (EBac.IM1 e EBac.IM2). Na imputação com zero, queda foi de 3% (EBac.IZ1 e EBac.IZ2). Nos procedimentos baseados em kNN a sensibilidade dos classificadores se manteve. O *teste T* apresentou uma diferença estatística com a perda de desempenho da imputação com média, apresentando um *p-value* de 0,0162.

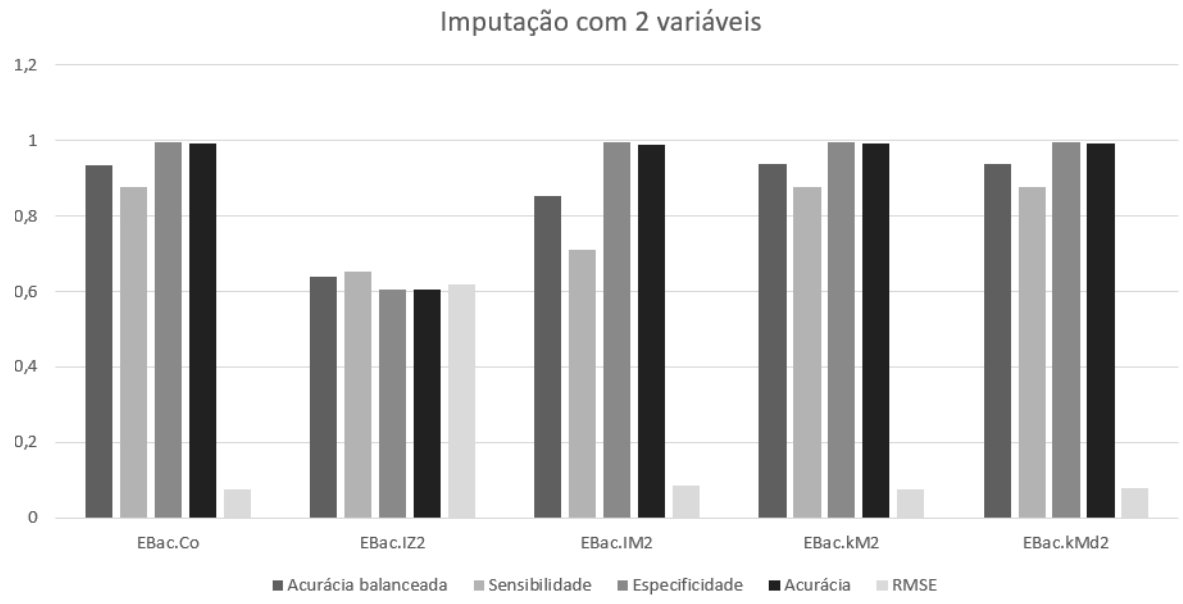
Assim, os dados da Tabela 9 mostram que, em relação a imputação com duas variáveis faltantes, houve uma redução dos escores de sensibilidade ou da especificidade dos classificadores em relação aos escores obtidos no teste EBac.Co. Tais resultados explicam a queda na acurácia balanceada dos classificadores. A imputação com zero e imputação com média apresentaram as maiores perdas de desempenho, 8% para o EBac.IM2 e 30% para a o EBac.IZ2. Para a imputação com a média a perda de desempenho está relacionada principalmente à redução na sensibilidade dos classificadores. Na imputação com zero a acurácia balanceada foi afetada pela queda da especificidade e da sensibilidade. Já aimputação com o kNN teve um aumento, porém inferior 0,2% na *acBal* dos classificadores.

Ainda com referência à Tabela 9, a análise dos resultados para a base de dados completa com aqueles observados quando da imputação de duas variáveis faltantes ao teste de *post-hoc* indicou que uma redução significativa da *acBal* na imputação com zero. A *acBal* na imputação com zero também foi significativamente menor do que na imputação com o kNN. Da mesma forma, a *acBal* na classificação de instâncias imputadas com a média foi significativamente menor do que aquela atingida na imputação como kNN (ambas abordagens) e na base completo. A aplicação do teste T de Student sobre os resultados dos pares de teste EBac.IZ1 vs EBac.IZ2, EBac.IM1 vs EBac.IM2, EBac.kM1 vs EBac.kM2, e EBac.kMd1 vs EBac.kMd2, na Tabela 9, indicou uma diferença estatística apenas entre os experimentos que envolveram a imputação com a média.

Em relação aos testes envolvendo a imputação de duas variáveis, o maior aumento do RMSE foi notado na imputação com zero (EBac.IZ2), cerca de 9 vezes o erro registrado no ensaio EBac.Co. Nos demais testes, o erro teve um aumento de no máximo 1%. O teste de *post-hoc* não revelou uma diferença significativa no RMSE entre EBac.Co e a imputação com kNN e com a média, considerando a imputação com duas variáveis faltantes. Para a imputação com zero, foi detectada uma diferença significativa do RMSE em comparação ao observado em EBac.Co, EBac.IM2, EBac.kM2 e EBac.kMd2. A aplicação do teste T sobre cada algoritmo, na imputação de uma e duas variáveis, indicou que o aumento de 10% de erro para a imputação com zero (EBac.IZ1 para EBac.IZ2) é significativo, com *p-value* de 0,0001.

Em resumo os resultados dos métodos de imputação com 2 variáveis faltantes podem ser representados pela Figura 7. Esses resultados permitem concluir que:

Figura 7: Resultados médios sobre a imputação com 2 variáveis faltantes



- A imputação com kNN não teve perda de desempenho em nenhum escore quando comparada ao caso completo;
- A imputação com zero foi o único escore que apresentou diferença em todos os escores na comparação com o caso completo;
- O incremento no número de variáveis faltantes fez com que os classificadores sofressem uma perda no escore da sensibilidade quando se aplicou a imputação com média, a imputação com zero foi relacionada a uma redução na especificidade;
- Houve uma queda da sensibilidade e acurácia balanceada quando da imputação com média em relação ao caso completo.

Tabela 10: Resultados da classificação para o gênero *Bacillus* com caso completo e contendo de 1 até 3 variáveis faltantes por instância

Experimento	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
EBac.Co	0,935	0,0522	0,8754	0,1042	0,9946	0,0028	0,9918	0,004	0,0766	0,0195
EBac.IZ1	0,6656	0,0921	0,6208	0,1769	0,7105	0,0335	0,7084	0,0339	0,5293	0,0298
EBac.IM1	0,9183	0,0461	0,8404	0,0912	0,9962	0,0019	0,9925	0,0037	0,0755	0,0208
EBac.kM1	0,9350	0,0525	0,8754	0,1042	0,9946	0,0030	0,9918	0,0044	0,0748	0,0184
EBac.kMd1	0,9350	0,0525	0,8754	0,1042	0,9946	0,0028	0,9918	0,0040	0,0757	0,0190
EBac.IZ2	0,6384	0,0653	0,6514	0,1388	0,6054	0,0244	0,6067	0,0229	0,6202	0,0170
EBac.IM2	0,8524	0,0637	0,7088	0,1276	0,996	0,0024	0,9893	0,0046	0,0864	0,0204
EBac.kM2	0,9367	0,0588	0,8784	0,1172	0,9951	0,0027	0,9923	0,0045	0,0749	0,0209
EBac.kMd2	0,9365	0,0586	0,8784	0,1172	0,9946	0,0026	0,9918	0,0042	0,0775	0,0204
EBac.IZ3	0,6234	0,0714	0,702	0,1380	0,5449	0,0342	0,5484	0,0343	0,6653	0,0263
EBac.IM3	0,8353	0,0505	0,6743	0,1019	0,9962	0,0027	0,9889	0,0037	0,0947	0,0147
EBac.kM3	0,9432	0,0446	0,8909	0,0882	0,9956	0,0029	0,9932	0,0039	0,0738	0,0224
EBac.kMd3	0,9379	0,0412	0,8818	0,080	0,9939	0,0039	0,9914	0,0052	0,0782	0,0226

A Tabela 10 mostra que a imputação com zero (EBac.IZ3) obteve o valor mais baixo para a especificidade, apenas 54% de acertos para classe $Y = 0$, quando do processamento de instâncias com três atributos faltantes. Isto é, no que se refere a este escore, houve uma perda média de 45% no desempenho em relação aquele registrado na base completa (EBac.Co). Mais uma vez o desempenho do classificador no experimento EBac.IZ3 se mostrou significativamente diferente daquele em EBac.Co. Mais ainda, os resultados de EBac.IZ3 também foram estatisticamente diferente de EBac.IM3 e kNN (EBac.kM3 e EBac.kMd3).

O experimento EBac.IZ3 também apresentou diferença estatística na especificidade para o experimento EBac.IZ2. O teste T de Student detectou que a queda de 6% na especificidade gerou um *p-value* de 0,0002. Para os demais procedimentos de imputação não se observou uma diferença no desempenho da classificação quando do tratamento de instâncias com duas ou três variáveis faltantes.

A sensibilidade para o testes com 3 variáveis faltantes por caso indicou que a substituição dos dados faltantes por zero ou por valores médios dos atributos não teve uma diferença de desempenho tão expressiva. Por outro lado, o desempenho dos classificadores sofreu uma perda de cerca de 17% em relação ao caso completo. Os resultados com o kNN mostram um aumento da sensibilidade para ambas as abordagens. O teste de Friedman com *post-hoc* mostrou que a diferença do sensibilidade do classificador quando da imputação com média foi estatisticamente diferente daquela obtida no caso completo (*p-value* de 0,01). O uso da imputação com média também produziu resultados estatisticamente diferentes dos obtidos por ambas as abordagens

do kNN. Na imputação com zero (EBac.IZ3), a queda de desempenho na sensibilidade em relação ao caso completo não foi suficiente para gerar uma diferença estatística relação ao completo (*p-value* de 0,07). Já quando a imputação com zero foi comparada ao kNN houve uma diferença identificada, com *p-value* de pelo menos 0,02.

A aplicação do *teste T* para verificar a diferença da sensibilidade nos testes com duas e três variáveis faltantes na Tabela 10 mostraram que a perda de desempenho de 3% e 5% para as imputação com média (EBac.IM2 e EBac.IM3) e zero (EBac.IZ2 e EBac.IZ3), respectivamente, não foram significativas do ponto de vista estatístico. O mesmo ocorreu para o kNN.

Os resultados dos experimentos EBac.kM3 e EBac.kMd3, na Tabela 10, mostram que a acurácia balanceada dos classificadores, quando da imputação com kNN, se manteve similar àquela observada no processamento da base de dados completa. Na imputação com média (EBac.IM3) e na imputação com zero (EBac.IZ3) foram identificadas, respectivamente, quedas de 10% e 31% nos valores deste escore em relação ao caso completo. O experimento EBac.IZ3 teve um desempenho estatísticamente diferente para o caso completo e do kNN. O *teste T* de *Student* não indicou diferenças significativas entre os experimentos que consideraram 2 e 3 variáveis faltantes.

O teste de Friedman com o *post-hoc* que comparou os resultados da *acBal* no caso completo e os experimentos com 3 variáveis faltantes, e indicou que a imputação com média(EBac.IM3) apresentou uma diferença estatística com relação ao experimentos EBac.kM3.

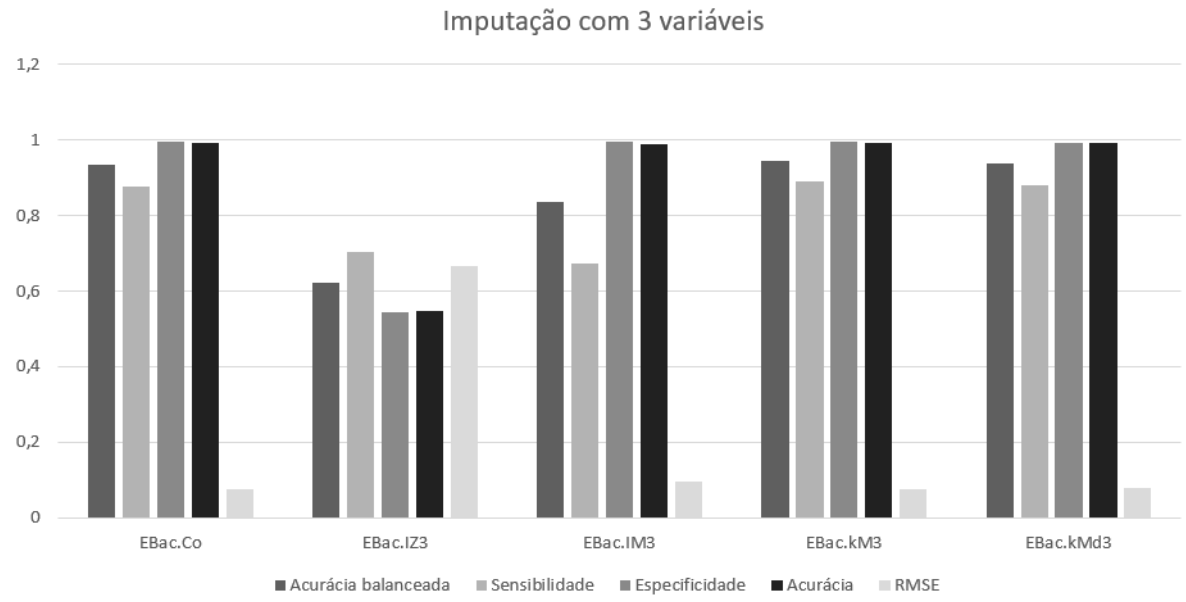
Ainda na Tabela 10 é possível notar que a imputação com zero (EBac.IZ3) foi o método que apresentou a maior perda de desempenho em relação ao RMSE, com um aumento de quase 60% em relação ao caso completo. A imputação com média também levou a um aumento deste escore, porém muito menor, cerca 2%. No que se refere à imputação com o kNN, o desempenho do classificador sofreu alterações inferiores a 0,01 para este escore.

O teste de Friedman indicou que houve diferença significativa entre os resultados do RMSE no experimento EBac.IZ3 e o EBac.Co. Também foi observada uma diferença entre os escores em EBac.IZ3 e com aqueles obtidos pelos classificadores quando da imputação de três valores com o kNN. A imputação com média apresentou diferença estatística apenas para kNN com agregação pela média.

A comparação de desempenho realizada pelo *teste T* em relação aos experimentos da Tabela 10 indicou que o aumento do erro em 4% do experimento EBac.IZ2 para EBac.IZ3 está associado a uma diferença estatística com um *p-value* de 0,0002. Não foram detectadas diferenças significativas para os demais testes.

Os resultados da classificação quando do emprego dos métodos de imputação em bases de testes com três variáveis faltantes por instância são ilustrados na Figura 8. A análise dos resultados permite concluir que:

Figura 8: Resultados médios sobre a imputação com 3 variáveis faltantes



- A classificação de dados submetidos à imputação com kNN não apresentou diferença significativa em relação ao caso completo em nenhum escore;
- A classificação de dados após a imputação com a média foi diferente de maneira significativa em relação ao caso completo na sensibilidade e acurácia balanceada;
- A imputação com zero fez com que o desempenho dos classificadores fosse significativamente diferente do caso completo em todos os escores;
- A imputação com zero produziu uma na diferença significativa na especificidade e no RMSE dos classificadores com o aumento do número de variáveis faltantes.

4.1.3 Resultados dos experimentos com imputação de um número variável de valores faltantes

A Tabela 11 reporta os resultados dos testes de classificação em bases de testes que continham um número variável de variáveis não observadas (1 até 8 variáveis faltantes em cada caso), conforme a Tabela 5.

Os testes de classificação registraram uma queda (47%) na especificidade para a imputação com zero em comparação àquela observada na base completa. Para a imputação com

Tabela 11: Experimentos para a Base Bacillus com caso completo e variando de 1 até 8 dados faltantes

Experimento	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
EBac.Co	0,9350	0,0522	0,8754	0,1042	0,9946	0,0028	0,9918	0,0040	0,0766	0,0195
EBac.IZNv	0,5468	0,0627	0,5782	0,1267	0,5154	0,0118	0,5169	0,0118	0,6898	0,0079
EBac.IMNv	0,7655	0,0878	0,5342	0,1762	0,9969	0,0024	0,9862	0,0054	0,1050	0,0221
EBac.kMNv	0,9095	0,0819	0,8234	0,1650	0,9955	0,0027	0,9916	0,0045	0,0755	0,0177
EBac.kMdNv	0,9146	0,0890	0,8339	0,1787	0,9953	0,0021	0,9916	0,0048	0,0749	0,0213

valores médios e com o kNN, o classificador obteve uma especificidade próxima a do caso completo, tendo obtido valores acima do mesmo nestes casos. A redução da especificidade no experimento EBac.IZNv em relação ao caso completo acarretou em uma diferença estatística para o teste de Friedman com o teste de *post-hoc*. O *p-value* obtido foi de 0,01. Além disso, de acordo como mesmo teste, o resultados do experimento EBac.IZNv foram estatisticamente diferentes daqueles obtidos da imputação com média e kNN para a especificidade.

Todos os métodos de imputação levaram a uma redução na sensibilidade dos classificadores quando confrontados com a base completa. A imputação com a média teve o pior desempenho médio, 53,42%, seguida pela imputação com zero (perda de 29% de sensibilidade). O melhor desempenho do classificador, com dados imputados, foi obtido pela imputação com kNN com função de agregação de mediana com 83,39% (4% abaixo de EBac.Co). Os experimentos EBac.IZNv e EBac.IMNv não apresentaram diferença estatística entre si quando submetidos ao teste de Friedman com *post-hoc*, porém divergiram estatisticamente do caso completo, e da imputação com kNN. Os métodos de imputação com kNN não apresentaram diferença estatística em relação ao caso completo.

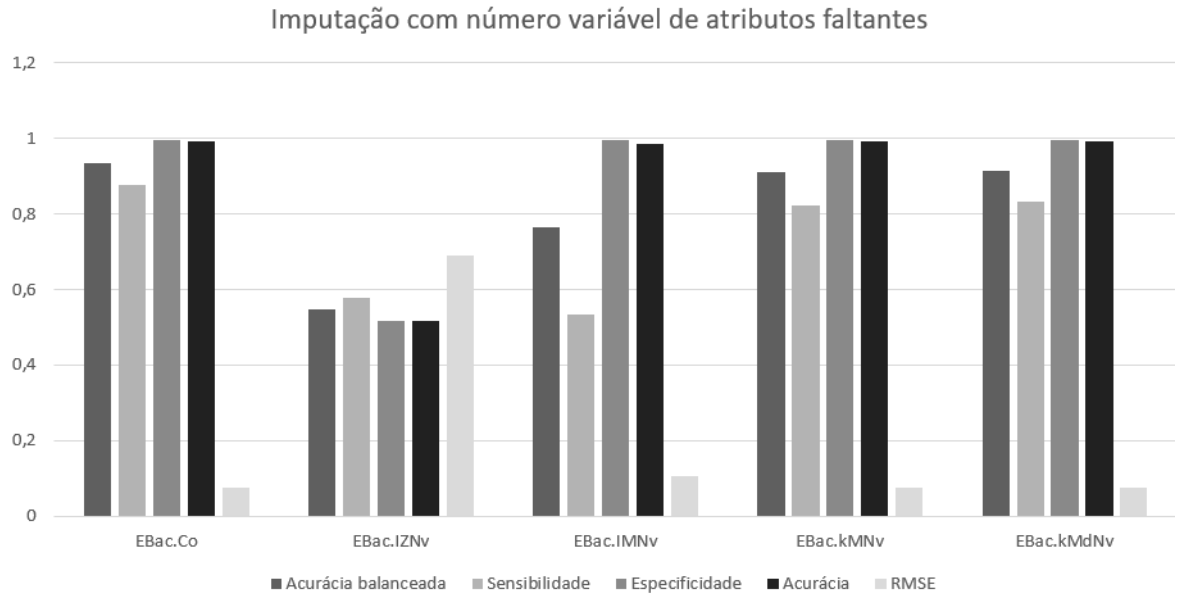
Os resultados de sensibilidade e especificidade se refletiram na acurácia balanceada para todos os algoritmos de imputação; queda de aproximadamente 38% na imputação com zero, 16% na imputação com a média e 2% para o kNN (ambas as versões). Quando os experimentos da Tabela 11 foram submetidos ao teste de Friedman com *post-hoc* foi detectado que a perda de desempenho do classificador na imputação com zero e na imputação com média foi significativa. O mesmo ocorreu quando os experimentos EBac.IZNv e EBac.IMNv foram comparados com a imputação com kNN. Todavia, a acurácia balanceada dos classificadores não apresentou diferença significativa quando do confronto dos resultados registrados nos experimentos com o kNN e com aqueles obtidos nos testes com a base completa.

O resultados do RMSE também mostram a média do erro médio quadrático do classificador na imputação zero é nove vezes maior do que aquele observado na classificação de dado completos. Este incremento do erro está associado há uma diferença significativa entre a

classificação com dados completos e com dados submetidos para imputação com zero. Isto é evidenciado por um *p-value* de 0,02 no teste de Friedmann com *post-hoc* . Na imputação com a média, o RMSE do classificador aumentou 1,3 vezes. Para as duas versões do kNN, o RMSE permaneceu próximo ao do caso completo. O teste de Friedman com o *post-hoc* indicou uma diferença estatística entre o experimento da imputação com zero quando confrontado com o desempenho obtido na base completa e quando se usou a imputação com o kNN. O kNN não apresentou-se significativamente diferente do caso completo.

Os resultados aqui apresentados podem ser resumidos pela Figura 9, bem como pode-se concluir que:

Figura 9: Resultados médios sobre a imputação com 1 até 8 variáveis faltantes



- O desempenho dos classificadores não apresentou diferença estatística para nenhum dos escores quando da imputação com o kNN;
- A especificidade dos classificadores foi maior quando da imputação com a média porém essa diferença numérica não foi estatisticamente significativa;
- A imputação com zero sempre levou a uma redução do desempenho do classificador.

4.2 RESULTADO DE CLASSIFICAÇÃO PARA A ESPÉCIE *STAPHYLOCOCCUS AUREUS*

4.2.1 Resultados do experimento para o caso completo

A Tabela 12 mostra os resultados na identificação de objetos que pertenciam ou não à espécie *S. aureus* a partir de dados completos. Foi obtida uma acurácia de 99%, com um RMSE de 1% (desvio de 3%).

Tabela 12: Resultado para a espécie *Staphylococcus aureus* da classificação para o caso completo

Método	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
ESt.Co	0,9775	0,0373	0,9993	0,0020	0,9557	0,0751	0,9975	0,0032	0,0316	0,0406

Uma vez que *Base S. aureus* também é desbalanceada (96% das amostras são da espécie *S. aureus*), classe $Y = 1$, é apropriado realizar a análise do desempenho do classificador por meio dos escores da acurácia balanceada, RMSE, sensibilidade, e especificidade. De ser notado que a sensibilidade do modelo que foi próxima de 100%. Dado o desbalanceamento da base de dados, isto está em acordo com a literatura que informa que bases desbalanceadas tendem a produzir modelos mais acurados para a classe majoritária (PHUNG; BOUZERDOUM; NGUYEN, 2009). Mesmo assim, o valor da especificidade, superior a 98%, mostra que o modelo foi capaz de detectar indivíduos que não pertenciam à espécie *Base S. aureus*, o que implicou em uma acurácia balanceada superior a 97%.

4.2.2 Resultados dos experimentos com a imputação de um número constante de valores faltantes em cada caso

As Tabelas 13, 14 e 16 listam os resultados dos experimentos em que a base de teste continha uma, duas ou três variáveis não observadas por instância. Novamente deve ser notado que em cada um dos casos processados nestes testes a performance do regressor logístico foi mensurada admitindo que todos as instâncias tinham o mesmo número de variáveis faltantes. Estes experimentos estão descritos na Tabela 6.

Na Tabela 13 é possível observar que a acurácia balanceada nos experimentos relacionados ao kNN (ESt.kM1 e ESt.kMd1) e a imputação com valores médios (ESt.IM1) foram aproximadamente iguais ao obtido no caso completo (ESt.Co). O único caso que apresentou uma queda significativa em relação ao teste ESt.Co foi a imputação com zero (ESt.IZ1), um decréscimo de 26% na acurácia balanceada com um *p-value* 7.10^{-7} para o teste de Friedman

Tabela 13: Resultados da classificação para a espécie *Staphylococcus aureus* com caso completo e com 1 variável faltante por instância

Experimento	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
ESt.Co	0,9775	0,0373	0,9993	0,002	0,9557	0,0751	0,9975	0,0032	0,0316	0,0406
ESt.IZ1	0,7150	0,0756	0,6885	0,0481	0,7414	0,1271	0,6899	0,0481	0,5546	0,0420
ESt.IM1	0,9650	0,0483	0,9993	0,0020	0,9307	0,0971	0,9968	0,0032	0,0384	0,0403
ESt.kM1	0,9775	0,0373	0,9993	0,0751	0,9557	0,0751	0,9975	0,0032	0,0317	0,0405
ESt.kMd1	0,9650	0,0483	0,9993	0,0020	0,9307	0,0971	0,9968	0,0032	0,0394	0,0411

com *post-hoc*. Mais ainda, a *acBal* na classificação de dados usando a imputação com zero teve escores inferiores àqueles vistos com os demais métodos imputação. Além disto, a diferença observada se mostrou significativa segundo o teste de Friedman com *post-hoc*. O kNN não apresentou diferença estatística em relação ao caso completo.

A especificidade apresentou um comportamento semelhante ao da acurácia balanceada. A imputação com zero (ESt.IZ1) ocasionou, novamente, a maior queda de desempenho em relação a classificação com dados completos, 21%, aproximadamente. Da mesma forma ao que aconteceu para a acurácia balanceada o teste de Friedman indicou a diferença estatística que foi apresentada na acurácia balanceada foi igualmente apresentada neste escore.

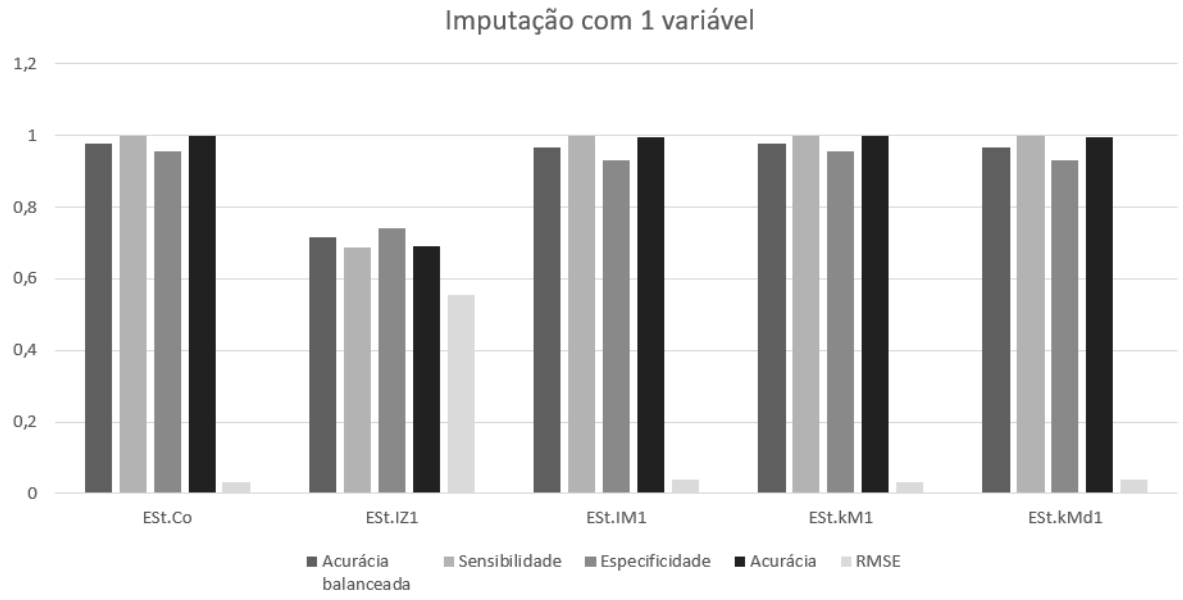
Para a sensibilidade o único experimento que apresentou perda de desempenho foi a imputação com zero. Uma diminuição de 31% no escore em relação ao observado no caso completo. A aplicação do teste de hipótese resultou em diferenças entre a imputação com zero quando confrontada com o caso completo os demais métodos de imputação. Nas demais comparações não foram encontradas diferenças estatísticas.

O RMSE teve um comportamento similar ao apresentado pela sensibilidade, onde apenas a imputação com zero teve uma perda de desempenho significativa em relação ao caso completo, aumento de dezoito vezes, já os demais algoritmos apresentaram diferenças inferiores a 0,01 em relação ao caso completo. A aplicação do teste de hipótese indicou diferenças significativas entre os resultados, sendo que o *post-hoc* apresentou *p-values* inferiores a 0,05 para todos os casos que o experimento ESt.IZ1 foi considerado.

Os resultados acima descritos podem ser resumidos pela Figura 10, bem como concluise que os resultados da Tabela 13, em relação ao desempenho do regressor logístico na presença de um atributo faltante por instância apresentam as seguintes interpretações:

- A imputação com kNN não apresentou diferença estatística do caso completo; a função

Figura 10: Resultados médios sobre a imputação com 1 variável faltante



com mediana expôs valores em sua maioria iguais ao caso completo;

- Nenhum dos escores avaliados apresentou diferença estatística em relação ao caso completo quando da aplicação da imputação com a média;
- A imputação com zero, teve uma queda em todos os escores analisados em relação ao caso completo; isto implicou em diferenças estatísticas para todos as comparações que este método foi considerado.

Tabela 14: Resultados da classificação para a espécie *Staphylococcus aureus* com caso completo, 1 e 2 variáveis faltantes por instância

Experimento	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
ESt.Co	0,9775	0,0373	0,9993	0,0020	0,9557	0,0751	0,9975	0,0032	0,0316	0,0406
ESt.IZ1	0,7150	0,0756	0,6885	0,0481	0,7414	0,1271	0,6899	0,0481	0,5546	0,0420
ESt.IM1	0,9650	0,0483	0,9993	0,0020	0,9307	0,0971	0,9968	0,0032	0,0384	0,0403
ESt.kM1	0,9775	0,0373	0,9993	0,0751	0,9557	0,0751	0,9975	0,0032	0,0317	0,0405
ESt.kMd1	0,9650	0,0483	0,9993	0,0020	0,9307	0,0971	0,9968	0,0032	0,0394	0,0411
ESt.IZ2	0,5233	0,1115	0,5479	0,0685	0,4986	0,2180	0,5466	0,0647	0,6715	0,0470
ESt.IM2	0,9237	0,0897	0,9993	0,0020	0,8480	0,1801	0,9937	0,0058	0,0633	0,0491
ESt.kM2	0,9567	0,0488	0,9993	0,0020	0,9140	0,0982	0,9962	0,0032	0,0472	0,0406
ESt.kMd2	0,9518	0,0491	0,9992	0,0021	0,9044	0,0991	0,9958	0,0031	0,0525	0,0394

A Tabela 14 mostra que a especificidade dos classificadores caiu, em comparação com aquela observada no processamento de dados completos, quando do processamento de dados

com dois valores faltantes, independentemente do método de imputação usado. A maior queda de desempenho neste escore ocorreu para a imputação com zero (ESt.IZ2), cerca de 46%. A imputação com média (ESt.IM2) teve uma especificidade 10,8% menor enquanto o kNN (ESt.kM2 e ESt.kMd2) apenas 4% (90% de acerto para $Y = 0$). Estas observações são evidenciadas pelo fato de que o resultado da especificidade no experimento ESt.IZ2 foi estatisticamente diferente dos demais experimentos relacionados a imputação com kNN, porém quando o ESt.IZ2 foi comparado ao ESt.IM2 um *p-value* de 0,08 foi retornado. Ao confrontar os experimentos com a mesma técnica de imputação, variando apenas o número de casos faltantes da especificidade na Tabelas 14, bases com uma ou duas variáveis faltantes com o *teste T*, apenas para a classificação de dados usando a imputação com zero apresentou como significativa (*p-value* de 0,0070).

No que se refere à sensibilidade e à acurácia balanceada, é possível observar que o desempenho do classificador não foi afetado pelo emprego dos métodos baseados no kNN (ESt.kM2 e ESt.kMd2) e na imputação com a média (ESt.IM2) no tratamento das bases com duas variáveis faltantes por instância. Essa conclusão foi confirmada pelo teste de Friedman com *post-hoc* sobre os dados da Tabela 14, onde os escores não apresentaram diferença estatística em relação ao caso completo. Por sua vez, a classificação usando a imputação com zero (ESt.IZ2) resultou em uma redução de 31% na sensibilidade e 34% na *acBal*, gerando uma diferença estatística para ambos os casos no teste de *post-hoc*. Um dos fatores que pode ter contribuído para estes resultados, pode advir do desbalanceamento da base dados, 96% da *Base S. aureus* pertence à categoria *Staphylococcus aureus*. Além disto, deve ser observado que muitos dos atributos descritores apresentam uma pequena variabilidade da relação m/z para esta espécie, como pode ser observado na Tabela 15.

A aplicação do *teste T* na Tabelas 14, relacionando os mesmos métodos de imputação variando apenas o número de variáveis faltantes, indicou que esse aumento não gerou uma diferença estatística entre as técnicas de imputação com média e kNN para a sensibilidade e *acBal*. Todavia, a imputação com zero apresentou quedas de 19% para *acBal* e 14% para sensibilidade, gerando assim um *p-value* de 0,0003 para *acBal* e 0,0001 para sensibilidade, isso é, diferença estatística em ambos os escores.

No que se refere a utilização de duas variáveis faltantes por instância, o RMSE dos classificadores cresceu de 0,03 no processamento de dados completos para aproximadamente 0,05 quando se utilizou a imputação baseada em kNN e para 0,06 quando se usou a imputação com a média. Na imputação com zero o erro subiu para 0,67. Quando se compara os resultados da Tabela 14 contendo uma e duas variáveis faltantes, notou-se um aumento de 0,12 no RMSE para a imputação com zero. Para os demais métodos o incremento foi inferior a 0,02. Para o teste de Friedman com *post-hoc*, o kNN não apresentou diferença para o caso completo,

Tabela 15: Estatística descritiva sobre as proteínas da base *S. aureus* para Y = 1

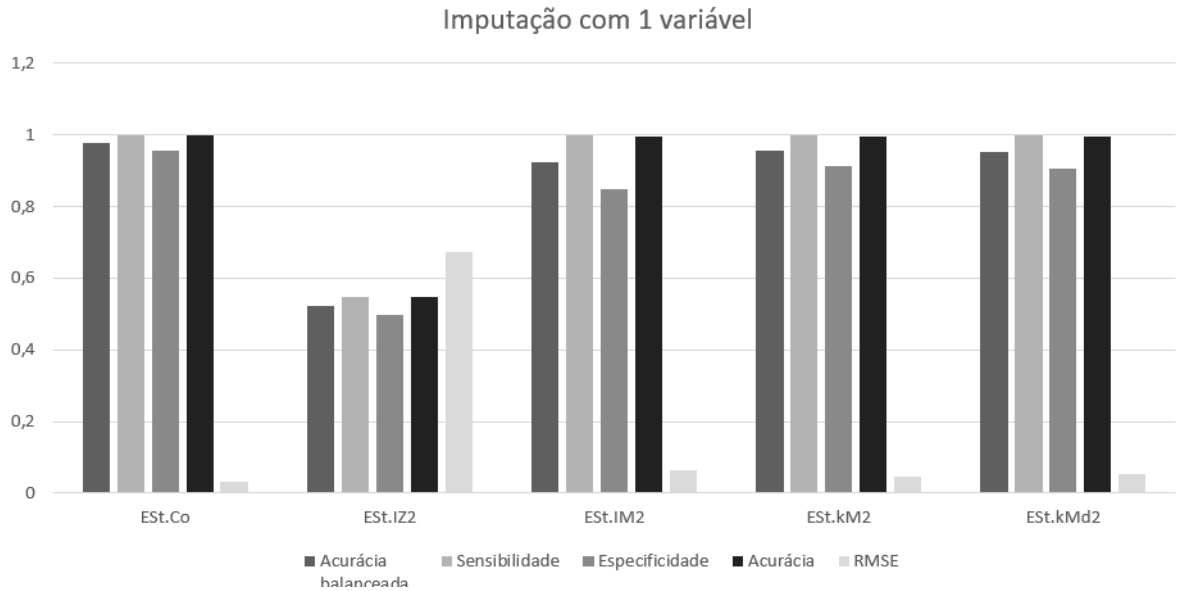
Proteína	L1	L3	L4	L6	L7a	L7.L12	L9
Média	24576,49	23602,67	22333,75	19655,10	9314,71	12580,86	16453,94
DP	12,41	17,20	4,02	4,15	21,83	5,12	3,42
Proteína	L10	L11	L14	L16	L17	L18	L19
Média	17578,86	14788,34	13135,31	16256,94	13615,69	13097,46	13240,67
DP	1,78	2,40	1,75	2,25	46,78	16,45	6,61
Proteína	L20	L21	L22	L23	L24	L25	L27
Média	13555,04	11333,26	12832,87	10605,40	11536,52	23653,35	10314,74
DP	1,51	2,52	103,02	1,06	0,90	73,02	1,14
Proteína	L28	L29	L30	L31	L32	L33	L35
Média	6845,68	8090,28	6422,54	9722,95	6353,80	5923,36	7564,97
DP	16,70	0,14	1,29	0,55	3,56	29,09	61,10
Proteína	L36	S2	S3	S4	S6	S8	S9
Média	4305,39	28976,24	23968,72	22882,40	11724,30	14698,48	14698,64
DP	0,71	51,13	1,99	2,96	0,90	92,47	2,25
Proteína	S13	S14	S16	S17	S19	S20	S21
Média	13587,70	9248,70	10103,79	10038,72	10484,06	8890,26	6840,97
DP	0,94	1554,06	2,02	10,69	0,70	0,00	1,35

divergindo apenas da imputação com zero, a imputação com média também não apresentou-se estatisticamente do caso completo, e a imputação com zero expôs um *p-value* de 0,0001 quando comparado ao caso completo. Em relação ao *teste T* que comparou os resultados entre 1 e 2 variáveis ausentes, o kNN e a média não apresentaram diferenças significativas com este aumento do número atributos faltantes, porém a imputação com zero apresentou-se uma diferença significativa com um *p-value* de 0,0001 .

Os resultados acima descritos são ilustrados na Figura 11, bem como conclui-se que os resultados da Tabela 14, em relação ao desempenho do regressor logístico na presença de dois atributos faltantes por instância apresentam as seguintes interpretações:

- O desempenho da classificação com dados usando o kNN não foi diferente daquele obtido na base de dados completa para todos os escores;
- A imputação com média apresentou uma queda de desempenho para os escores *acBal*, especificidade e RMSE, em relação ao caso completo, porém sem apresentar diferença estatística;
- Mais uma vez a imputação com zero apresentou queda de desempenho expressiva, resultado em diferenças estatísticas em relação ao caso completo.

Figura 11: Resultados médios sobre a imputação com 2 variáveis faltantes



Para a Tabela 16, a especificidade, que neste caso representa a classe minoritária, demonstrou que o classificador acertou apenas metade dos registros disponíveis, com um erro de 45% na imputação com zero em relação ao caso completo, os demais métodos de imputação tiveram uma queda menor, 11% para a imputação com média e entre 3% e 4% para o kNN, para o caso completo. Para o teste de Friedman com o *post-hoc* a imputação com zero apresentou-se estatisticamente diferente tanto da imputação com média e kNN como do caso completo. Em comparação com a especificidade da Tabela 14 não foram apresentadas diferenças numéricas expressivas, apesar do DP da imputação com média e imputação zero serem próximos de 0,20, porém para o *teste T* isso não foi suficiente para apresentar diferenças estatísticas para quaisquer comparações.

A sensibilidade na Tabela 16 apresentou queda de desempenho apenas para a imputação com zero, sendo os demais resultados numericamente idênticos ao caso completo. Assim, o teste de hipótese aplicado apontou como estatisticamente diferente os pareamentos em que a imputação com zero esteve presente. Como o único resultado a apresentar variação foi a imputação com zero, apenas nesta se fez necessária a aplicação do *teste T* que prontamente negou qualquer diferença estatística entre as sensibilidades das Tabelas 14 e 16 com um *p-value* de 0,4733.

Os resultados da Tabela 16 mostram que a acurácia balanceada manteve seu desempenho para o algoritmo de imputação com kNN (ESt.kM3, ESt.kMd3) em relação a classificação com dados completos. Em relação a imputação com média (ESt.IM3) e imputação com zero (ESt.IZ3), houve uma queda de 1% e 49% para estes algoritmos, respectivamente, em compara-

Tabela 16: Resultados da classificação para a espécie *Staphylococcus aureus* com caso completo e contendo de 1 até 3 variáveis faltantes por instância

Experimento	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
ESt.Co	0,9775	0,0373	0,9993	0,0020	0,9557	0,0751	0,9975	0,0032	0,0316	0,0406
ESt.IZ1	0,7150	0,0756	0,6885	0,0481	0,7414	0,1271	0,6899	0,0481	0,5546	0,0420
ESt.IM1	0,9650	0,0483	0,9993	0,002	0,9307	0,0971	0,9968	0,0032	0,0384	0,0403
ESt.kM1	0,9775	0,0373	0,9993	0,0751	0,9557	0,0751	0,9975	0,0032	0,0317	0,0405
ESt.kMd1	0,9650	0,0483	0,9993	0,0020	0,9307	0,0971	0,9968	0,0032	0,0394	0,0411
ESt.IZ2	0,5233	0,1115	0,5479	0,0685	0,4986	0,2180	0,5466	0,0647	0,6715	0,0470
ESt.IM2	0,9237	0,0897	0,9993	0,0020	0,8480	0,1801	0,9937	0,0058	0,0633	0,0491
ESt.kM2	0,9567	0,0488	0,9993	0,0020	0,9140	0,0982	0,9962	0,0032	0,0472	0,0406
ESt.kMd2	0,9518	0,0491	0,9992	0,0021	0,9044	0,0991	0,9958	0,0031	0,0525	0,0394
ESt.IZ3	0,5173	0,0889	0,5267	0,0607	0,5080	0,1818	0,5248	0,0551	0,6877	0,0409
ESt.IM3	0,9228	0,0760	0,9993	0,0020	0,8464	0,1518	0,9943	0,0045	0,0572	0,0418
ESt.kM3	0,9550	0,0691	0,9993	0,0020	0,9107	0,1387	0,9962	0,0043	0,0399	0,0439
ESt.kMd3	0,9611	0,0495	0,9992	0,0021	0,9230	0,0997	0,9965	0,0032	0,0402	0,0395

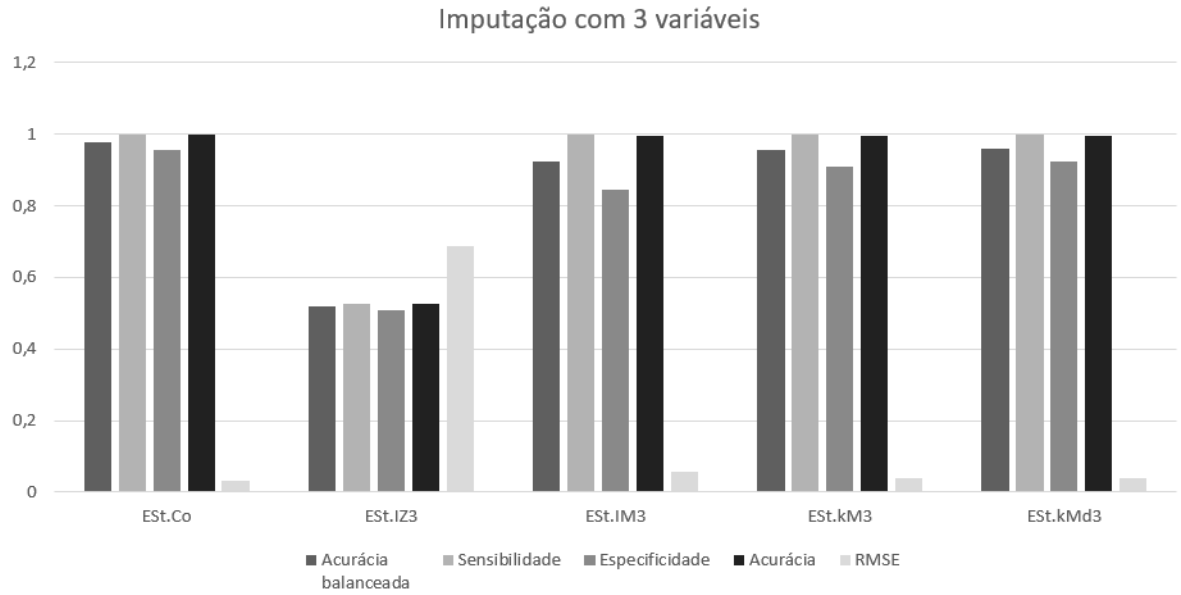
ção ao experimento ESt.Co. Apesar do algoritmo de imputação com média ter decaído, não foi suficiente para gerar uma diferença estatística quando comparado com o teste de Friedman, por outro lado a imputação com zero apresentou-se estatisticamente diferente em todas as comparações que foi considerada. O *teste T* não apresentou uma diferença de desempenho significativa para o aumento de 2 para 3 variáveis faltantes por caso.

O RMSE se mostrou com uma queda de desempenho para todos os experimentos em comparação ao experimento ESt.Co, com quedas de 66%, para a imputação com zero, 5% para a imputação com média, e 3% para os experimentos com kNN. Aplicando o teste de Friedman foram identificadas diferenças estatísticas entre os experimentos, o *post-hoc* apontou para os pareamentos ESt.Co e ESt.IM3, ESt.Co e ESt.IZ3, ESt.kM3 e ESt.IZ3, ESt.kP3 e ESt.IZ3 como responsáveis.

Os resultados acima descritos podem ser resumidos pela Figura 12, bem como concluise que os resultados da Tabela 16, em relação ao desempenho do regressor logístico na presença de três atributos faltantes por instância apresentam as seguintes interpretações:

- A classificação dos dados usando a imputação com kNN para *acBal* teve um resultado próximo ao obtido nos experimentos com duas variáveis faltantes; isto mostra que, o kNN não foi tão afetado pelo aumento do número de variáveis faltantes;
- A imputação com zero levou a quedas no desempenho dos classificadores em todos os

Figura 12: Resultados médios sobre a imputação com 3 variáveis faltantes



escores, implicando em diferenças estatísticas em relação ao caso completo;

- A imputação com a média produziu uma redução dos escores de classificação em relação ao caso completo; em particular, detectou-se uma diferença estatística para o RMSE.

4.2.3 Resultados dos experimentos com imputação de um número variável de valores

A Tabela 17 mostra os resultados de classificação após a imputação de dados faltantes em bases de testes em que número de atributos não observados variava de um até oito. Os resultados mostram que houve uma diminuição da especificidade dos classificadores para todos os métodos de imputação numa comparação com valores observados no teste ESt.Co. O melhor desempenho na classificação das instâncias que não pertenciam a classe meta foi atingido no emprego do kNN com a função de agregação pela mediana na imputação dos dados (2% de decréscimo). Para a imputação com kNN usando agregação com a média, a imputação com a média e a imputação com zero, a perda de desempenho foi de 6%, 5% e 54%, respectivamente.

O teste de Friedman demonstrou que o kNN não apresentou diferença estatística em relação ao caso completo, porém o kNN com função de agregação de mediana foi significativamente diferente da imputação com média. Por outro lado, a imputação com zero juntamente com a imputação com média apresentaram *p-values* abaixo de 0,05 quando comparados ao caso completo.

A sensibilidade apresentou resultados similares ao caso completo na maioria dos métodos, tendo tido uma queda de aproximadamente 55% para a imputação com zero, e os demais

Tabela 17: Experimentos para a espécie *Staphylococcus aureus* com caso completo e variando de 1 até 8 dados faltantes

Experimentos	Acurácia balanceada		Sensibilidade		Especificidade		Acurácia		RMSE	
	Média	DP	Média	DP	Média	DP	Média	DP	Média	DP
ESt.Co	0,9775	0,0373	0,9993	0,0020	0,9557	0,0751	0,9975	0,0032	0,0316	0,0406
ESt.IZNv	0,4471	0,0532	0,4488	0,0626	0,4454	0,1384	0,4476	0,0564	0,7422	0,0379
ESt.IMNv	0,9508	0,0650	0,9993	0,0020	0,9023	0,1306	0,9950	0,0049	0,0538	0,0480
ESt.kMNv	0,9475	0,1247	0,9993	0,0020	0,8957	0,2498	0,9956	0,0077	0,0394	0,0557
ESt.kMdNv	0,9675	0,0641	0,9993	0,0020	0,9357	0,1287	0,9968	0,0043	0,0348	0,0459

algoritmos com desempenho numericamente igual ao caso completo. Dessa forma, o teste de Friedman indicou a diferença estatística nos casos que a imputação com zero se fez presente.

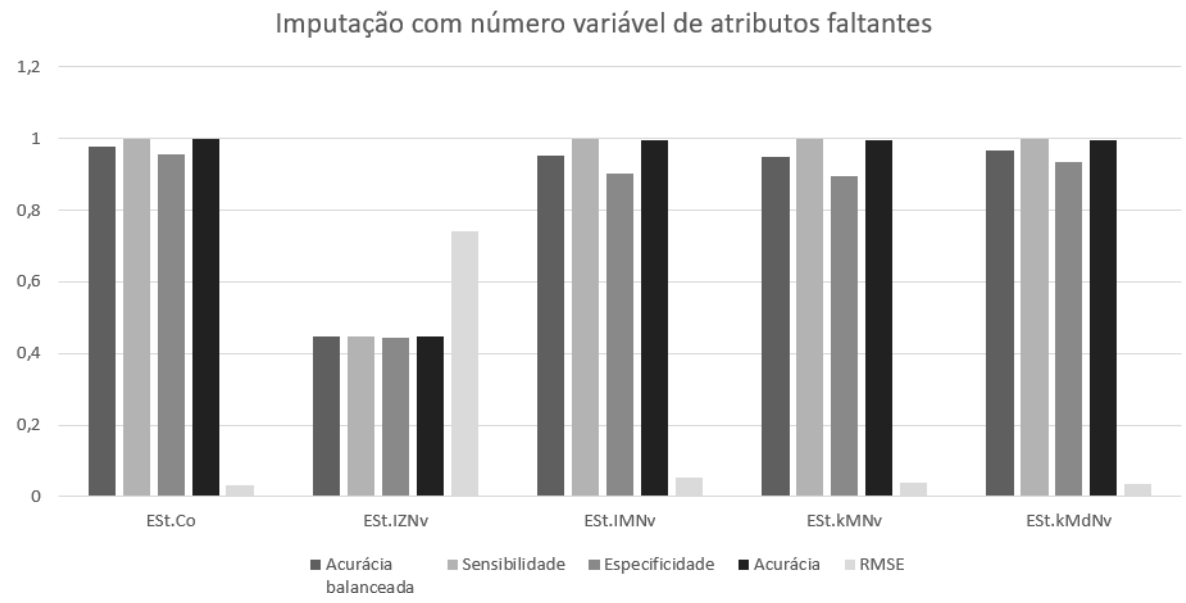
Para a acurácia balanceada, o kNN teve uma perda entre 1% e 3%, funções de mediana e média respectivamente, em relação ao caso completo. A imputação com média sofreu um decréscimo de 2%, e a imputação com zero chegou próximo aos 53% de perca, relação ao caso completo. Apesar de todos os algoritmos terem apresentado queda em relação ao caso completo, apenas a imputação com média e imputação com zero foram significativamente diferentes para o teste de *post-hoc*, o kNN só apresentou diferença em relação a imputação com zero.

O RMSE teve um aumento expressivo para a imputação com zero para este caso variável, chegou a 71% de aumento, vinte e três vezes maior que o apresentado no caso completo. A imputação com a média também teve um aumento, porém de apenas 2%, já o kNN teve um aumento mais discreto, não chegando a 1% em relação ao caso completo. O teste de Friedman apresentou a existência de diferenças estatísticas, sendo que o *post-hoc* apontou estas como sendo entre o caso completo e a imputação com média e imputação com zero, além da imputação com zero com kNN.

Os resultados acima descritos podem ser resumidos pela Figura 13, bem como concluise que os resultados da Tabela 17, em relação ao desempenho do regressor logístico na presença de um número variável de atributos faltantes por instância apresentam as seguintes interpretações:

- O kNN não apresentou quedas significativas em relação ao caso completo;
- A imputação com zero foi inferior as demais métodos de imputação em todos os escores analisados;
- A imputação com média teve uma queda de 2% na *acBal* em relação ao caso completo, tendo obtido um valor superior ao kNN com função de agregação de média.

Figura 13: Resultados médios sobre a imputação com um número variável de atributos faltantes



4.3 DISCUSSÃO

Esta seção analisa os resultados obtidos nos testes em que as bases de dados tinham um número variável de atributos faltantes; Seções 4.1.3 e 4.2.3. A escolha destes resultados para análise porque não é possível prever o número de atributos faltantes em situações reais.

4.3.1 Desempenho da regressão logística com os dados imputados

Antes de examinar a influência da imputação com kNN na identificação de bactérias é útil comparar o desempenho dos classificadores gerados durante os testes deste trabalho com aquele apresentado por sistemas similares. Neste sentido, os trabalhos de Tomachewski *et al.* (2018) e Tomachewski (2017) permitem proceder uma avaliação qualitativa dos resultados da seção anterior com aqueles obtidos por um classificador que também foi aprendido a partir da *Base PUKYU*. Nesta avaliação é importante observar que, como visto na Seção 2.5, aqueles autores abordaram a tarefa de identificar a espécie e o gênero de bactérias como um problema de classificação politômico em que procedimento de imputação atribuía uma constante aos valores faltantes. Assim, o escopo do problema abordado naquele trabalho difere do atual neste aspecto.

O desempenho do classificador desenvolvido em Tomachewski *et al.* (2018) e Tomachewski (2017) foi avaliado apenas pela acurácia e os valores obtidos foram de 94,83% na identificação de para espécies e 98,69% no reconhecimento de gêneros. Os classificadores logísticos gerados no presente trabalho tiveram um acurácia média superior a 99% tanto no processamento de dados completos quanto no tratamento dos dados imputados. A acurácia balanceada

na identificação da espécie *S. aureus*, nos testes em que se aplicou a imputação com o kNN foi superior à acurácia obtida por aqueles autores na identificação de espécies. Entre os fatores que concorreram para isto está a escolha do kNN como algoritmo de imputação. Isto porque, como mostram Liao *et al.* (2014), a imputação com o kNN tende a ter um desempenho superior àquele apresentado por métodos que atribuem constantes aos valores faltantes.

Também deve ser notado que os atributos da base de dados *S. aureus* apresentam pouca variabilidade em instâncias relacionadas à classe alvo. Como pode ser visto nos ensaios, isto também concorreu para uma redução do erros de predição. Tanto para a imputação com o kNN quanto para a imputação com a média. Este fator pode ter contribuído para que os classificadores da *S. aureus* tivessem uma acurácia balanceada superior à 95% quando do emprego daqueles algoritmos.

No presente trabalho, os classificadores para identificação de instâncias do gênero *Bacillus* obtiveram uma acurácia superior a 99% e uma a acurácia balanceada superior a 91%, com o kNN na imputação dos dados. Assim, no que se refere a acurácia o desempenho foi maior do que a aquela reportada por Tomachewski (2017). A acurácia balanceada foi quatro porcento menor do que a acurácia obtida naquele trabalho Contudo, como observado, aqueles trabalhos não explicitam os resultados da acurácia balanceada e da matriz de confusão, isto dificulta uma análise mais detalhada dos resultados da acurácia balanceada, além do fato do escopo ser diferente.

Finalmente, é importante destacar que o uso do kNN para imputação produziu resultados de classificação próximos aos obtidos quando do processamento da base de dados completa. Os classificadores obtiveram um desempenho médio superior a 90%, tanto para acurácia quanto para acurácia balanceada, quando do uso kNN na imputação dos dados. Estes valores foram superiores àqueles atingidos com a imputação com a média e a imputação zero. Estes resultados estão em consonância com os de Jonsson e Wohlin (2004) que afirmam que os métodos não paramétricos são, frequentemente, mais eficientes do o uso de constantes.

4.3.2 Análise do erro de imputação

O principal fator que contribui para a perda de desempenho dos classificadores logísticos é o erro no cálculo da probabilidade posterior de cada hipótese, $P(Y|\mathbf{x})$. Nos experimentos realizados este erro foi estimado pelo SE (Erro Quadrático, do inglês *Squared Error*). No caso de dados incompletos, o incremento/decremento no valor do SE depende do número de variáveis faltantes, do coeficientes destas variáveis e do erro de imputação.

As Tabelas 18 e 19 mostram o erro médio relativo ($\overline{E}_r$) e o erro ponderado relativo

(E_{pr}) da imputação de variáveis nos testes de identificação de bactérias do gênero *Bacillus* e da espécie *S. aureus*, respectivamente. Estas tabelas se referem aos testes em que o número de atributos não observados era variável (Seções 4.1.3 e 4.2.3). Nestas tabelas, os dados do erro relativo para a imputação com zero não é listado pois, o erro relativo é sempre igual a 1 (100%) para este método. O erro ponderado relativo é definido como:

$$E_{pr} = \frac{\sum_i^j (\beta_i x_i' - \beta_i x_i)}{\sum_i^j \beta_i x_i}$$

em que x_i e x_i' são os valores real e imputado do *i*-ésimo atributo. Assim, o E_{pr} mensura a diferença no valor do modelo linear inerente ao classificador logístico, para um determinado caso e influência no erro quadrático e no erro quadrático médio.

Tabela 18: Erros de imputação para Base *Bacillus* para o caso variável

Método	$\overline{E_r}$		$\overline{E}_{pr}$	
	Média	DP	Média	DP
IM	0,0402	0,0015	0,3541	0,3694
kNN Média	0,0191	0,0020	0,1097	0,0721
kNN Mediana	0,0175	0,0020	0,0683	0,0235

Tabela 19: Erros de imputação para Base *S. aureus* para o caso variável

Método	$\overline{E_r}$		$\overline{E}_{pr}$	
	Média	DP	Média	DP
IM	0,0050	0,0009	0,0038	0,0019
kNN Média	0,0037	0,0010	0,0011	0,0018
kNN Mediana	0,0035	0,0011	0,0009	0,0013

Os resultados mostram que o erro relativo de imputação com kNN teve uma média inferior a de 2%. Tanto nos testes para identificação de *Bacillus* quanto nos testes para *S. aureus*. Para ambas as bases houve uma diferença estatística quando aplicado o *teste T* bicaudal entre o kNN (ambas abordagens) e a imputação com a média. A análise da média do erro ponderado relativo evidencia que o kNN teve, em média, um erro de aproximadamente 11% para o gênero *Bacillus* e 1% para a espécie *S. aureus*. Em particular, a média do erro ponderado da imputação com kNN usando função de agregação foi de aproximadamente 7%. Como pode ser observado, estes valores são 3 a 5 vezes menores que os erros obtidos na imputação com a média.

Para a *Base S. aureus*, os erros $\overline{E}_r$ e $\overline{E}_{pr}$ foram inferiores a 1%. Concorreu para este resultado o fato de que os atributos tem pouca variabilidade na classe majoritária (ver Tabela 15). Isto fez com que a imputação com o kNN e com média atribuíssem valores próximos aos observados no caso completo e assim houvesse uma redução dos erros.

As Tabelas 20 e 21 mostram, para as bases de dados *Bacillus* e *S. aureus*, as correlações entre: (a) o Número de Variáveis Faltantes (NVF) e o erro relativo da imputação e (b) o NVF e o erro quadrático na predição. Como pode ser observado a correlação entre o NVF e os erros do kNN são desprezíveis (MUKAKA, 2012). Logo, é possível concluir que o NVF não afetou significativamente o erro da imputação ou o erro da predição. Adicionalmente, as correlações associadas ao kNN foram significativamente menores do que aquelas registradas pela imputação com a média, com exceção do kNN com função de agregação de mediana . As correlações para a imputação com zero também foram nulas.

Tabela 20: Correlações de erros de imputação para Base *Bacillus* - número variável de casos faltantes

Algoritmo de imputação	Correlações	
	NVF x Erro Médio	NVF x SE
kNN Média	0,0893	-0,0082
kNN Mediana	0,0549	-0,0086
IM	0,1374	0,0324

Tabela 21: Correlações de erros de imputação para Base *S. aureus*- número variável de casos faltantes

Algoritmo de imputação	Correlações	
	NVF x Erro Médio	NVF x SE
kNN Média	0,1462	-0,0068
kNN Mediana	0,1154	0,0009
IM	0,1398	0,0378

A Tabela 22 lista as correlações entre o número de variáveis faltantes (*Num var*) e erro médio, e número de variáveis faltantes e erro quadrático para cada uma das classes dos experimentos: EBac.IMNv, EBac.kMNv e EBac.kMdNv com a base *Bacillus*. Nesta tabela é possível observar que, na maioria dos casos, a correlação entre o número de variáveis e o erro de imputação é fraca (MUKAKA, 2012) quando do processamento de casos pertencentes à classe $Y = 1$ e desprezível quando $Y = 0$, independente do método de imputação. O único caso em que a correlação não é desprezível é aquele em que os dados pertenciam à classe minoritária e o tratamento dos dados incompletos foi feito pela a imputação com a média.

No que diz respeito à relação entre NVF e o SE, a correlação foi desprezível quando do uso do kNN. Entretanto, isso não prevaleceu na imputação com a média para a classe minoritária ($Y = 1$). Os resultados experimentais mostram uma correlação alta (de 0,70 até 0,90) entre NVF e SE quando aqueles algoritmos foram executados. Assim, os dados da tabela sugerem que o incremento do número de variáveis faltantes não levou o kNN imputar valores que

Tabela 22: Correlações de erros de imputação para Base *Bacillus* por classe - número variável de casos faltantes

Algoritmo de imputação	Classe	Correlações	
		NVF x Erro Médio	NVF x SE
IM	0	0,1405	-0,0215
	1	0,3117	0,7070
kNN Média	0	0,0967	0,0002
	1	0,1929	0,0117
kNN Mediana	0	0,0612	0,0018
	1	0,1423	0,0155

aumentassem o SE, o que poderia reduzir a capacidade preditiva do classificador. Por sua vez, os dados também sugerem que NVF tem uma forte influência sobre o SE quando do uso da imputação com a média.

Tabela 23: Correlações de erros de imputação para Base *S. aureus* por classe - número variável de casos faltantes

Algoritmo de imputação	Classe	Correlações	
		NVF x Erro Médio	NVF x SE
IM	0	0,2501	0,1040
	1	0,1891	0,1201
kNN Média	0	0,1242	-0,1698
	1	0,1804	0,0269
kNN Mediana	0	0,2000	-0,1087
	1	0,1405	0,0266

A Tabela 23 apresenta as correlações entre o NVF e o erro médio para a Base *S. aureus* e as correlações entre o número de variáveis faltantes e o erro quadrático. Esses valores foram obtidos dos resultados produzidos nos testes ESt.IMNv, ESt.kMNv e ESt.kMdNv. Naquela tabela, é possível observar que em todos os casos as correlações ficaram abaixo de 0,30. Isto evidencia que o número de variáveis faltantes não teve efeito sobre aqueles erros.

5 CONCLUSÃO

Este trabalho apresentou um estudo sobre o efeito da imputação de dados no desempenho de classificadores logísticos para identificação de bactérias a partir de dados extraídos de espectros de massa do tipo MALDI-TOF. Foram realizados experimentos que permitiram avaliar como a acurácia, a acurácia balanceada, a sensibilidade, a especificidade e o RMSE do classificador foram afetados pelo uso do kNN na imputação de um número variável de atributos faltantes. A motivação para a utilização de um método não paramétrico de imputação foi a não detecção de relacionamentos entre os atributos descritores.

O desempenho dos classificadores logísticos foram, em termos de acurácia, próximos aos obtidos por Tomachewski (2017) para um problema da classificação politômica. Na base *Bacillus*, a sensibilidade foi afetada negativamente pelo fato das instâncias referentes à classe alvo pertencerem à classe minoritária. De maneira análoga, na base *S. aureus*, observou-se um fenômeno similar no que diz respeito à especificidade. Assim, influenciada pelo desbalanceamento dos dados, a acurácia balanceada foi menor que a acurácia.

Os experimentos de imputação foram executados utilizando um procedimento de validação cruzada denominado VCSDI. Os resultados dos experimentos de classificação com os dados imputados pelo kNN foram comparados com os resultados obtidos na classificação com dados completos. A análise destes resultados mostrou que a imputação com o kNN não implicou uma perda significativa no desempenho dos classificadores em relação ao caso completo quando da identificação da espécie *S. aureus* e do gênero *Bacillus*. Este algoritmo de imputação se mostrou eficiente em todos os cenários apresentados. Adicionalmente, a quantidade de atributos faltantes não aumentou significativamente o erro da imputação e da probabilidade posterior calculada pelo classificador. Este resultado fornece evidências de que o algoritmo kNN foi capaz de imputar valores cujos erros não afetaram a capacidade de predição do modelo.

O desempenho da classificação com dados imputados pelo o kNN também foi comparado com o registrado quando do uso de outros dois métodos de imputação não paramétrica: a imputação com a média e a imputação zero. O kNN propiciou, na maioria dos testes, um desempenho do superior aos demais classificadores.

Como trabalhos futuros pretende-se abordar aspectos do problema que, como observados nos experimentos, podem influir o desempenho dos classificadores ou da imputação de dados. Assim, objetiva-se estudar o efeito do uso de técnicas de balanceamento de dados na base de dados a ter dados imputados. Também pretende-se empregar métodos de classificação e imputação que possibilitem considerar situações em que uma proteína é observável em algumas

espécies mas não em outras.

Adicionalmente, há o interesse de se desenvolver classificadores e testar a imputação com o kNN para outras espécies e gêneros de bactérias usando a base *PUKYU* e ou outras bases de dados biológicas. A implementação de modelos de regressão logística politômicos e o teste do classificador logístico no tratamento de dados obtidos em laboratório também é considerada.

REFERÊNCIAS

ALASALMI, T. *et al.* Classification uncertainty of multiple imputed data. In: IEEE. *Computational Intelligence, 2015 IEEE Symposium Series on*. [S.l.], 2015. p. 151–158.

ALTMAN, E. I. Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The journal of finance*, Wiley Online Library, v. 23, n. 4, p. 589–609, 1968.

AMBLER, G.; OMAR, R. Z.; ROYSTON, P. A comparison of imputation techniques for handling missing predictor values in a risk model with a binary outcome. *Statistical methods in medical research*, Sage Publications Sage UK: London, England, v. 16, n. 3, p. 277–298, 2007.

ARGUDÍN, M. Á.; MENDOZA, M. C.; RODICIO, M. R. Food poisoning and staphylococcus aureus enterotoxins. *Toxins*, Molecular Diversity Preservation International, v. 2, n. 7, p. 1751–1773, 2010.

ARSHAM, H. Kuiper's p-value as a measuring tool and decision procedure for the goodness-of-fit test. *Journal of Applied Statistics*, Taylor & Francis, v. 15, n. 2, p. 131–135, 1988.

BATISTA, G. E.; MONARD, M. C. A study of k-nearest neighbour as an imputation method. *HIS*, v. 87, n. 251-260, p. 48, 2002.

BEKKAR, M.; DJEMAA, H. K.; ALITOUCHE, T. A. Evaluation measures for models assessment over imbalanced data sets. *Iournal Of Information Engineering and Applications*, v. 3, n. 10, 2013.

BELLMAN, R. E. *Adaptive control processes: a guided tour*. [S.l.]: Princeton university press, 2015.

BERETTA, L.; SANTANIELLO, A. Nearest neighbor imputation algorithms: a critical evaluation. *BMC medical informatics and decision making*, BioMed Central, v. 16, n. 3, p. 74, 2016.

BORTZ, J.; LIENERT, G. A.; KLAUS, B. *Verteilungsfreie Methoden in der Biostatistik*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. (Springer-Lehrbuch). ISBN 978-3-540-74706-2. Disponível em: <dx.doi.org/10.1007/978-3-540-74707-9>.

BRAIN, D.; WEBB, G. On the effect of data set size on bias and variance in classification learning. In: *Proceedings of the Fourth Australian Knowledge Acquisition Workshop, University of New South Wales*. [S.l.: s.n.], 1999. p. 117–128.

BRUYNE, K. D. *et al.* Bacterial species identification from maldi-tof mass spectra through data analysis and machine learning. *Systematic and applied microbiology*, Elsevier, v. 34, n. 1, p. 20–29, 2011.

DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern classification*. [S.l.]: John Wiley & Sons, 2012.

ECKEL-PASSOW, J. E. *et al.* An insight into high-resolution mass-spectrometry data. *Biostatistics*, Oxford University Press, v. 10, n. 3, p. 481–500, 2009.

ELMONGUI, H. G.; MOKBEL, M. F.; AREF, W. G. Continuous aggregate nearest neighbor queries. *GeoInformatica*, v. 17, n. 1, p. 63–95, Jan 2013. ISSN 1573-7624. Disponível em: <https://doi.org/10.1007/s10707-011-0149-0>.

FACELI, K. *et al.* Inteligência artificial: Uma abordagem de aprendizado de máquina. *Rio de Janeiro: LTC*, v. 2, p. 192, 2011.

FRIEDMAN, M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the american statistical association*, Taylor & Francis, v. 32, n. 200, p. 675–701, 1937.

GOULART, V. A. M.; RESENDE, R. R. Maldi-tof: uma ferramenta revolucionária para as análises clínicas e pesquisa do câncer. *Nanocell News*, v. 1, n. 3, 11 2013.

GOWER, J. C. A general coefficient of similarity and some of its properties. *Biometrics*, JSTOR, p. 857–871, 1971.

GROSS, J. H. *Mass spectrometry: a textbook*. [S.l.]: Springer Science & Business Media, 2011.

HAIR, J. F. *et al. Análise multivariada de dados*. [S.l.]: Bookman Editora, 2009.

HASSANAT, A. B. *et al.* Solving the problem of the k parameter in the knn classifier using an ensemble learning approach. *arXiv preprint arXiv:1409.0919*, 2014.

HASTIE, T. *et al. Imputing missing data for gene expression arrays*. [S.l.]: Stanford University Statistics Department Technical report, 1999.

HAWKINS, D. M. The problem of overfitting. *Journal of chemical information and computer sciences*, ACS Publications, v. 44, n. 1, p. 1–12, 2004.

HORTAL, J.; LOBO, J. M.; JIMÉNEZ-VALVERDE, A. Limitations of biodiversity databases: case study on seed-plant diversity in tenerife, canary islands. *Conservation Biology*, Wiley Online Library, v. 21, n. 3, p. 853–863, 2007.

HOSMER, D.; LEMESHOW, S.; STURDIVANT, R. *Applied Logistic Regression*. 3rd. ed. Wiley, 2013. ISBN 9780470582473. Disponível em: <http://gen.lib.rus.ec/book/index.php?md5=D42CA76DA4F0EA3B4E5724428673BDAB>.

HRYDZIUSZKO, O.; VIANT, M. R. Missing values in mass spectrometry based metabolomics: an undervalued step in the data processing pipeline. *Metabolomics*, Springer, v. 8, n. 1, p. 161–174, 2012.

HU, L.-Y. *et al.* The distance function effect on k-nearest neighbor classification for medical datasets. *SpringerPlus*, Nature Publishing Group, v. 5, n. 1, p. 1304, 2016.

HURLBERT, A. H.; JETZ, W. Species richness, hotspots, and the scale dependence of range maps in ecology and conservation. *Proceedings of the National Academy of Sciences*, National Acad Sciences, v. 104, n. 33, p. 13384–13389, 2007.

HYNDMAN, R. J.; ATHANASOPOULOS, G. *Forecasting: principles and practice*. [S.l.]: OTexts, 2014.

JAPKOWICZ, N.; SHAH, M. *Evaluating learning algorithms: a classification perspective*. [S.l.]: Cambridge University Press, 2011.

JONSSON, P.; WOHLIN, C. An evaluation of k-nearest neighbour imputation using likert data. In: IEEE. *Software Metrics, 2004. Proceedings. 10th International Symposium on*. [S.l.], 2004. p. 108–118.

KANG, H. The prevention and handling of the missing data. *Korean journal of anesthesiology*, v. 64, n. 5, p. 402–406, 2013.

KANG, W. *et al.* Factors associated with negative t-spot. tb results among smear-negative tuberculosis patients in china. *Scientific reports*, Nature Publishing Group, v. 8, n. 1, p. 4236, 2018.

KARABULUT, E. M.; ÖZEL, S. A.; IBRIKCI, T. A comparative study on the effect of feature selection on classification accuracy. *Procedia Technology*, Elsevier, v. 1, p. 323–327, 2012.

KIM, K. *et al.* Detection of pancreatic cancer biomarkers using mass spectrometry. *Cancer informatics*, SAGE Publications Sage UK: London, England, v. 13, p. CIN–S16341, 2014.

KIM, Y. J. *et al.* Stratified sampling design based on data mining. *Healthcare informatics research*, v. 19, n. 3, p. 186–195, 2013.

KOMAREK, P. Logistic regression for data mining and high-dimensional classification. *Robotics Institute*, p. 222, 2004.

KOTU, V.; DESHPANDE, B. *Predictive analytics and data mining: concepts and practice with rapidminer*. [S.l.]: Morgan Kaufmann, 2014.

KRAUTENBACHER, N.; THEIS, F. J.; FUCHS, C. Correcting classifiers for sample selection bias in two-phase case-control studies. *Computational and mathematical methods in medicine*, Hindawi, v. 2017, 2017.

LACHISH, S.; MURRAY, K. A. The certainty of uncertainty: Potential sources of bias and imprecision in disease ecology studies. *Frontiers in veterinary science*, Frontiers Media SA, v. 5, 2018.

LEE, J. *et al.* Svm classification model of similar bacteria species using negative marker: Based on matrix-assisted laser desorption/ionization time-of-flight mass spectrometry. In: IEEE. *Bioinformatics and Bioengineering (BIBE), 2017 IEEE 17th International Conference on*. [S.l.], 2017. p. 145–150.

LIAO, S. G. *et al.* Missing value imputation in high-dimensional phenomic data: imputable or not, and how? *BMC bioinformatics*, BioMed Central, v. 15, n. 1, p. 346, 2014.

LITTLE, R. J.; RUBIN, D. B. *Statistical analysis with missing data*. [S.l.]: John Wiley & Sons, 2014.

LU, L. *et al.* Disease status determination: Exploring imputation and selection techniques. *The Canadian Journal of Statistics/La Revue Canadienne de Statistique*, JSTOR, p. 197–201, 2011.

MANKIEWICZ, R. *The story of mathematics*. [S.l.]: Cassell, 2000.

MEEYAI, S. Logistic regression with missing data: A comparison of handling methods, and effects of percent missing values. *Journal of Traffic and Logistics Engineering Vol*, v. 4, n. 2, 2016.

MICHELIN, D. *et al.* Avaliação da atividade antimicrobiana de extratos vegetais. *Rev Bras Farmacogn*, v. 15, n. 4, p. 316–20, 2005.

MIRANDA, J. *et al.* Atividade antibacteriana de extratos de folhas de montrichardia linifera (arruda) schott (araceae). *Rev. Bras. Pl. Med*, v. 17, n. 4 supl III, p. 1142–1149, 2015.

MUKAKA, M. M. A guide to appropriate use of correlation coefficient in medical research. *Malawi Medical Journal*, Medical Association of Malawi, v. 24, n. 3, p. 69–71, 2012.

PHUNG, S. L.; BOUZERDOUM, A.; NGUYEN, G. H. Learning pattern classification tasks with imbalanced data sets. 2009.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2014. Disponível em: <http://www.r-project.org>.

REMUS, J. J. *et al.* Comparison of a distance-based likelihood ratio test and k-nearest neighbor classification methods. In: IEEE. *Machine Learning for Signal Processing, 2008. MLSP 2008. IEEE Workshop on*. [S.l.], 2008. p. 362–367.

RUBIN, D. B. Inference and missing data. *Biometrika*, Oxford University Press, v. 63, n. 3, p. 581–592, 1976.

RUETE, A. Displaying bias in sampling effort of data accessed from biodiversity databases using ignorance maps. *Biodiversity data journal*, Pensoft Publishers, n. 3, 2015.

RUSSEL, S.; NORVIG, P. Inteligência artificial. *Editora Campus*, p. 26, 2004.

SAAR-TSECHANSKY, M.; PROVOST, F. Handling missing values when applying classification models. *Journal of machine learning research*, v. 8, n. Jul, p. 1623–1657, 2007.

SASAKI, Y. *et al.* The truth of the f-measure. *Teach Tutor mater*, v. 1, n. 5, 2007.

SCHMITT, P.; MANDEL, J.; GUEDJ, M. A comparison of six methods for missing data imputation. *Journal of Biometrics & Biostatistics*, OMICS Publishing Group, v. 6, n. 1, p. 1, 2015.

SCHULMAN, L.; TOIVONEN, T.; RUOKOLAINEN, K. Analysing botanical collecting effort in amazonia and correcting for it in species range estimation. *Journal of Biogeography*, Wiley Online Library, v. 34, n. 8, p. 1388–1399, 2007.

SHALIZI, C. *Advanced data analysis from an elementary point of view*. [S.l.]: Cambridge University Press Cambridge, 2013.

SILVEIRA, A. B. d. Isolamento e caracterização de linhagens de bacillus e paenibacillus promotores de crescimento vegetal em lavouras de arroz e trigo do rio grande do sul. 2008.

SOUTO, M. C. D.; JASKOWIAK, P. A.; COSTA, I. G. Impact of missing data imputation methods on gene expression clustering and classification. *BMC bioinformatics*, BioMed Central, v. 16, n. 1, p. 64, 2015.

SOUZA, A. P. O. *et al.* Atividade antimicrobiana dos sumos de alecrim, aroeira, guiné e mastruz sobre staphylococcus aureus e escherichia coli. *Scientia Plena*, v. 11, n. 7, p. 9, 2015.

SOUZA, R. d.; AMBROSINI, A.; PASSAGLIA, L. M. Plant growth-promoting bacteria as inoculants in agricultural soils. *Genetics and molecular biology*, SciELO Brasil, v. 38, n. 4, p. 401–419, 2015.

STRIMBU, K.; TAVEL, J. A. What are biomarkers? *Current Opinion in HIV and AIDS*, NIH Public Access, v. 5, n. 6, p. 463, 2010.

SUAREZ, S. *et al.* Ribosomal proteins as biomarkers for bacterial identification by mass spectrometry in the clinical microbiology laboratory. *Journal of microbiological methods*, Elsevier, v. 94, n. 3, p. 390–396, 2013.

TAMURA, H.; HOTTA, Y.; SATO, H. Novel accurate bacterial discrimination by maldi-time-of-flight ms based on ribosomal proteins coding in s10-spc-alpha operon at strain level s10-germs. *Journal of The American Society for Mass Spectrometry*, Springer, v. 24, n. 8, p. 1185–1193, 2013.

TAN, P.-N. *et al. Introduction to data mining*. [S.l.]: Pearson Education India, 2006.

THEODORIDIS, S.; KOUTROUMBAS, K. *Pattern Recognition, Fourth Edition*. 4th. ed. [S.l.]: Academic Press, 2008. ISBN 1597492728, 9781597492720.

TODD, J. F. Recommendations for nomenclature and symbolism for mass spectroscopy (including an appendix of terms used in vacuum technology).(recommendations 1991). *Pure and applied chemistry*, De Gruyter, v. 63, n. 10, p. 1541–1566, 1991.

TOMACHEWSKI, D. *Utilização de aprendizagem de máquina para classificação de bactérias através de proteínas Ribossomais*. 1–72 p. Dissertação (Mestrado) — Universidade Estadual de Ponta Grossa, 2017.

TOMACHEWSKI, D.; GALVÃO, C. W.; ETTO, R. M. *PUKYU - Banco de Dados de Massa Molecular de Proteínas Ribossomais*. [S.l.], 2018.

TOMACHEWSKI, D. *et al.* Ribopeaks: a web tool for bacterial classification through m/z data from ribosomal proteins. *Bioinformatics*, 2018.

VENABLES, W. N.; RIPLEY, B. D. *Modern applied statistics with S-PLUS*. [S.l.]: Springer Science & Business Media, 2013.

VIJAYKUMAR, V. Classifying bacterial species using computer vision and machine learning. *International Journal of Computer Applications*, Foundation of Computer Science, v. 151, n. 8, 2016.

WALJEE, A. K. *et al.* Comparison of imputation methods for missing laboratory data in medicine. *BMJ open*, British Medical Journal Publishing Group, v. 3, n. 8, p. e002847, 2013.

WEINBERGER, K. Q.; SAUL, L. K. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, v. 10, n. Feb, p. 207–244, 2009.

WILLIAMS, D. *et al.* On classification with incomplete data. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 29, n. 3, p. 427–436, 2007.

WITTEN, I. H. *et al. Data Mining: Practical machine learning tools and techniques*. [S.l.]: Morgan Kaufmann, 2016.

XIE, Y. *et al.* Rapid identification and classification of staphylococcus aureus by attenuated total reflectance fourier transform infrared spectroscopy. *Journal of food safety*, Wiley Online Library, v. 32, n. 2, p. 176–183, 2012.

YILDIRIM, P. Filter based feature selection methods for prediction of risks in hepatitis disease. *International Journal of Machine Learning and Computing*, IACSIT Press, v. 5, n. 4, p. 258, 2015.

ZHANG, Z. Missing data imputation: focusing on single imputation. *Annals of translational medicine*, AME Publications, v. 4, n. 1, 2016.

ZIEGLER, D. *et al.* Ribosomal protein biomarkers provide root nodule bacterial identification by maldi-tof ms. *Applied microbiology and biotechnology*, Springer, v. 99, n. 13, p. 5547–5562, 2015.

APÊNDICE A – QUANTIDADE DE DADOS FALTANTES DE CADA PROTEÍNA POR CLASSE

Dados faltantes para base Bacillus

Classe	**L1**	**L2**	**L3**	**L4**	**L5**
Bacillus	68	67	50	60	59
Não Bacillus	8741	8689	3860	4012	6610
Classe	**L6**	**L7**	**L7a**	**L7ae**	**L7.L12**
Bacillus	67	332	483	498	79
Não Bacillus	8003	29177	27947	26735	10337
Classe	**L9**	**L10**	**L11**	**L12**	**L13**
Bacillus	62	45	96	548	81
Não Bacillus	9306	7361	8807	30792	8699
Classe	**L14**	**L15**	**L16**	**L17**	**L18**
Bacillus	88	81	82	88	75
Não Bacillus	9536	8435	9710	9101	8740
Classe	**L19**	**L20**	**L21**	**L22**	**L23**
Bacillus	84	75	79	89	63
Não Bacillus	8714	9643	8879	8974	6927
Classe	**L24**	**L25**	**L27**	**L28**	**L29**
Bacillus	126	277	89	99	103
Não Bacillus	8696	11124	9520	10606	10071
Classe	**L30**	**L31**	**L32**	**L33**	**L34**
Bacillus	92	89	131	110	182
Não Bacillus	11683	8902	10905	11162	14664
Classe	**L35**	**L36**	**S1**	**S2**	**S3**
Bacillus	96	151	546	74	81
Não Bacillus	10759	14903	30552	8436	8696
Classe	**S4**	**S5**	**S6**	**S7**	**S8**
Bacillus	62	78	73	86	78
Não Bacillus	8567	9090	9243	9162	6662
Classe	**S9**	**S10**	**S11**	**S12**	**S13**
Bacillus	72	127	70	169	62
Não Bacillus	6865	10430	6005	10563	4747
Classe	**S14**	**S15**	**S16**	**S17**	**S18**
Bacillus	71	112	86	76	106
Não Bacillus	9589	8964	9402	4518	10385

Classe	**S19**	**S20**	**S21**	**S22**	**S31e**
Bacillus	97	74	122	552	552
Não Bacillus	9244	9773	15021	30930	31470
Total de registros Classe Bacillus: 552					
Total de registros Classe Não Bacillus: 31474					
Limiar: 250					
Proteínas com mais de 302 dados faltantes são eliminadas para a Classe Bacillus Proteínas com mais de 31224 dados faltantes são eliminadas para a Classe Não Bacillus					

Dados faltantes para Base S. aureus

Classe	**L1**	**L2**	**L3**	**L4**	**L5**
Staphylococcus aureus	151	159	11	23	72
Não Staphylococcus aureus	8658	8597	3899	4049	6597
Classe	**L6**	**L7**	**L7a**	**L7ae**	**L7.L12**
Staphylococcus aureus	147	4025	928	926	159
Não Staphylococcus aureus	7923	25484	27502	26307	10257
Classe	**L9**	**L10**	**L11**	**L12**	**L13**
Staphylococcus aureus	148	113	155	4063	147
Não Staphylococcus aureus	9220	7293	8748	27277	8633
Classe	**L14**	**L15**	**L16**	**L17**	**L18**
Staphylococcus aureus	151	151	151	145	149
Não Staphylococcus aureus	9473	8365	9641	9044	8666
Classe	**L19**	**L20**	**L21**	**L22**	**L23**
Staphylococcus aureus	150	151	152	147	111
Não Staphylococcus aureus	8648	9567	8806	8916	6879
Classe	**L24**	**L25**	**L27**	**L28**	**L29**
Staphylococcus aureus	153	164	154	148	157
Não Staphylococcus aureus	8669	11237	9455	10557	10017
Classe	**L30**	**L31**	**L32**	**L33**	**L34**
Staphylococcus aureus	153	158	161	147	190
Não Staphylococcus aureus	11622	8833	10875	11125	14656
Classe	**L35**	**L36**	**S1**	**S2**	**S3**
Staphylococcus aureus	147	160	4064	143	150
Não Staphylococcus aureus	10708	14894	27034	8367	8627
Classe	**S4**	**S5**	**S6**	**S7**	**S8**
Staphylococcus aureus	148	152	153	148	101
Não Staphylococcus aureus	8481	9016	9163	9100	6639

Classe	**S9**	**S10**	**S11**	**S12**	**S13**
Staphylococcus aureus	103	149	21	170	16
Não Staphylococcus aureus	6834	10408	6054	10562	4793
Classe	**S14**	**S15**	**S16**	**S17**	**S18**
Staphylococcus aureus	93	153	146	23	160
Não Staphylococcus aureus	9567	8923	9342	4571	10331
Classe	**S19**	**S20**	**S21**	**S22**	**S31e**
Staphylococcus aureus	152	149	157	4064	4064
Não Staphylococcus aureus	9189	9698	14986	27418	27958

Total de registros Classe Staphylococcus aureus: 4064
Total de registros Classe Não Staphylococcus aureus: 27962
Limiar: 100
Proteínas com mais de 3964 dados faltantes são eliminadas para Classe Staphylococcus aureus. Proteínas com mais de 27862 dados faltantes são eliminadas para a Classe Não Staphylococcus aureus.