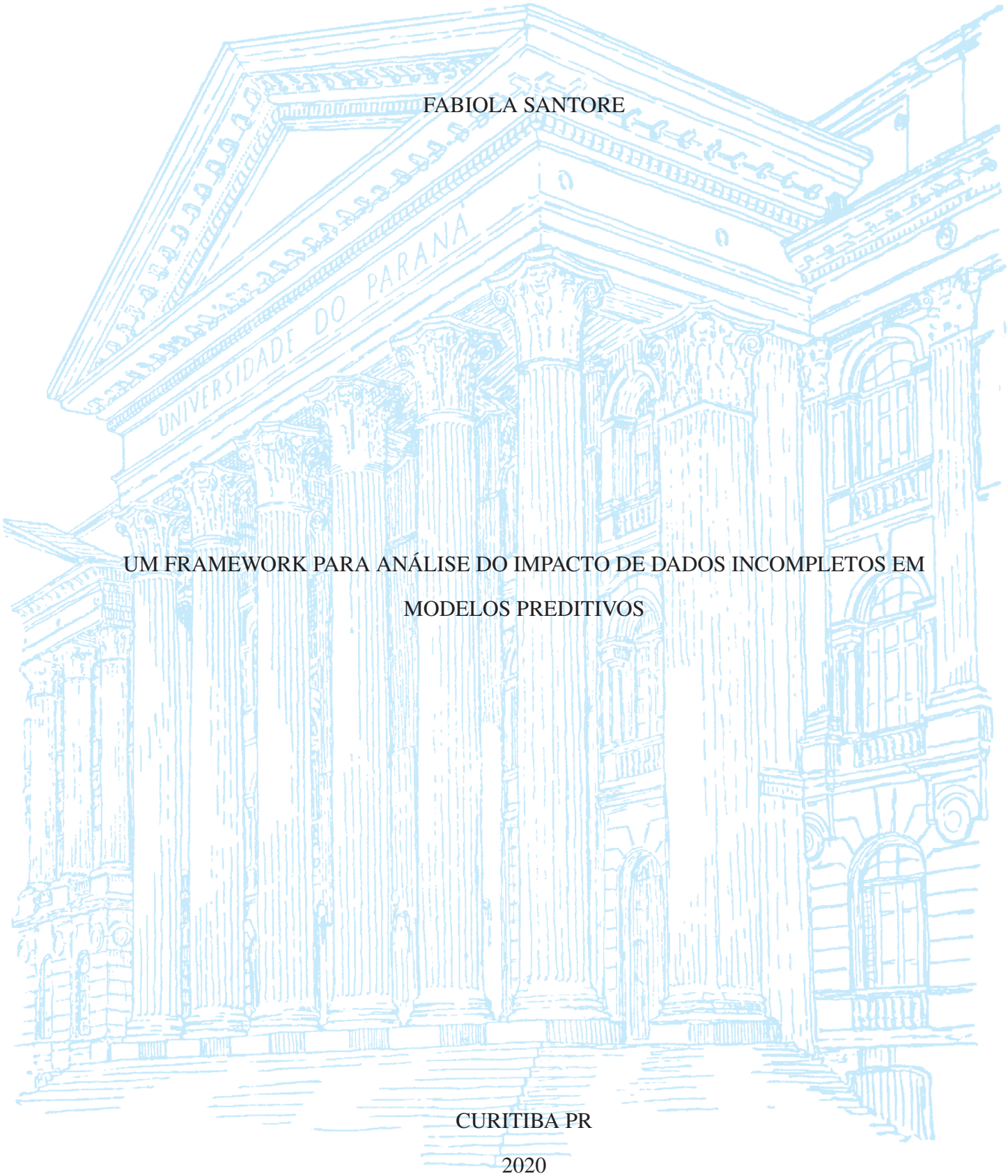
UNIVERSIDADE FEDERAL DO PARANÁ

FABIOLA SANTORE



UM FRAMEWORK PARA ANÁLISE DO IMPACTO DE DADOS INCOMPLETOS EM MODELOS PREDITIVOS

CURITIBA PR

2020

FABIOLA SANTORE

UM FRAMEWORK PARA ANÁLISE DO IMPACTO DE DADOS INCOMPLETOS EM MODELOS PREDITIVOS

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Informática no Programa de Pós-Graduação em Informática, setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: Prof. Dr. Eduardo Cunha de Almeida.

Coorientador: Prof. Dr. Wagner Hugo Bonat.

CURITIBA PR

2020

Catalogação na Fonte: Sistema de Bibliotecas, UFPR
Biblioteca de Ciência e Tecnologia

S237f Santore, Fabiola
Um framework para análise do impacto de dados incompletos em modelos preditivos [recurso eletrônico] / Fabiola Santore. – Curitiba, 2020.

Dissertação - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Informática, 2020.

Orientador: Eduardo Cunha de Almeida.
Coorientador: Wagner Hugo Bonat.

1. Conjunto de caracteres (Processamento de dados). I. Universidade Federal do Paraná. II. Almeida, Eduardo Cunha de. III. Bonat, Wagner Hugo. IV. Título.

CDD: 006

Bibliotecária: Vanusa Maciel CRB- 9/1928

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em INFORMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da dissertação de Mestrado de **FABIOLA SANTORE** intitulada: **Um Framework para Análise do Impacto de Dados Incompletos em Modelos Preditivos**, que após terem inquirido a aluna e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 20 de Agosto de 2020.

Assinatura Eletrônica
21/08/2020 09:56:16.0
EDUARDO CUNHA DE ALMEIDA
Presidente da Banca Examinadora (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
21/08/2020 10:24:21.0
LUIZ EDUARDO SOARES DE OLIVEIRA
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
21/08/2020 09:33:58.0
CESAR AUGUSTO TACONELI
Avaliador Externo (DEPARTAMENTO DE ESTATÍSTICA DA UNIVERSIDADE FEDERAL DO PARANÁ)

"Estou longe de ser perfeita e muito longe de saber tudo, mas sei o bastante para querer ser a melhor parte das pessoas que mais amo"

AGRADECIMENTOS

Agradeço sobre todas as coisas à Deus, pelo dom da vida, pelas lições que aprendi ao longo da minha caminhada, me fortalecendo a cada dia para seguir e realizar os meus sonhos.

Aos meus avós maternos, Elizário Cordeiro Marques e Sofia Marques (in memoriam) por toda base familiar, por me ensinarem a ser uma pessoa do bem, me apoiando até os últimos dias de suas vidas e além.

Ao meu pai, Sidiney Santore, por ser o meu melhor amigo e meu maior incentivador, por me ensinar a ser uma pessoa forte, por todo suporte financeiro e emocional, por acreditar no meu potencial e pelos melhores concelhos nos momentos mais difíceis.

À minha mãe, Marinéia Marques, por me ensinar a nunca desistir e me mostrar que a vida sempre nos oferta algo de bom, mesmo nos momentos ruins, basta sabermos olhar com os olhos do coração.

Às minhas irmãs, Bárbara Camila Flissak e Júlia Caroline Flissak, por todo apoio, carinho, conselhos, cuidados e principalmente o amor que sempre sentiram por mim. Eu sempre fui privilegiada, por poder compartilhar todos os pesos e dores com vocês, por sempre ter dois escudos à minha frente, minhas fortalezas, amo vocês incodicionalmente.

Às minhas sobrinhas e ao bebê que está à caminho, Angela Neubauer, Camila Flissak Graefling e João ou Maria, que são o maior amor da minha vida, sempre me ensinando a ter esperança e ver a vida com mais leveza, me mostrando que as coisas mais simples da vida, são as que mais importam.

Aos meus avós paternos, Santina e Domingos, e aos demais membros da minha família, que sempre rezaram por mim, me apoiaram e torceram pelas minhas conquistas

Ao meu orientador, Professor Doutor Eduardo Cunha de Almeida, por acreditar em meu potencial e me dar a oportunidade de estar entre seus orientados, mesmo em uma nova área de atuação. Por todo suporte e ensinamentos para a condução desse trabalho, além de toda a troca de experiência para o meu crescimento acadêmico e pessoal.

Ao meu co-orientador, Professor Doutor Wagner Hugo Bonat, por aceitar fazer parte desse projeto, me proporcionando suporte teórico e o equilíbrio necessário para encontrar a afinidade entre ambas as áreas, além de sempre ter as palavras necessárias para me manter firme na minha trajetória acadêmica.

Aos meu amigos, Jéssica, Arielle, Rayani, Cibele, Michele, Douglas, Carla e todos os outros que sempre entenderam minha ausência por conta dos estudos e sempre me proporcionaram os melhores momentos de distração, me dando o folego necessário pra eu continuar firme em minha caminhada.

Aos colegas do LBD e do C3SL, pela agradável companhia ao longo do meu mestrado, além das trocas acadêmicas.

E por fim, agradeço à banca avaliadora, Professor Doutor Cesar Augusto Taconeli e Professor Doutor Luiz Eduardo Soares de Oliveira,por terem aceito o convite, pela disponibilidade de leitura do meu trabalho e por todas as considerações valiosa.

RESUMO

A qualidade dos dados é fundamental no suporte a sistemas centrados em dados, rotinas de aprendizagem de máquinas e modelos preditivos. A pesquisa sobre a qualidade dos dados visa definir, identificar e reparar as inconsistências nos dados. Uma fonte comum de inconsistências são os dados incompletos, representado por valores ausentes, que são aqueles registros que não foram observados ou armazenados por alguma razão, mas para os quais existe um valor real no ambiente em que pertencem. Esse tipo de problema potencialmente esconde informações importantes sobre o conjunto de dados e impacta na aplicação em que será utilizado. A qualidade das variáveis de entrada e saída tem sido negligenciada na proposição de novos modelos preditivos, embora a popularidade da análise preditiva utilizando ferramentas de aprendizagem de máquina tenha aumentado. Como consequência, o efeito de dados incompletos em muitos modelos preditivos padrão é completamente desconhecido. Sendo assim, propomos um *framework* estocástico para avaliar o impacto de dados incompletos no desempenho dos modelos de preditivos. O *framework* permite o controle total de aspectos importantes da estrutura do conjunto de dados, tais como a quantidade e o tipo das variáveis de entrada, a correlação entre as variáveis de entrada e seu poder de previsão geral, e o tamanho da amostra. O mecanismo gerador de dados incompletos é aplicado a partir de uma distribuição multivariada Bernoulli, o que nos permite simular valores ausentes gerados a partir de diferentes variações do mecanismo MCAR (*Missing Completely at Random*). Embora o *framework* possa ser aplicado a diversos tipos de modelos preditivos, neste trabalho, nos concentramos no modelo de regressão logística e escolhemos a acurácia como medida preditiva. Os resultados da simulação mostram que os efeitos dos dados incompletos desaparecem para grandes tamanhos de amostra, como esperado. Por outro lado, à medida que o número de variáveis de entrada aumenta, a acurácia diminui principalmente para entradas binárias. Em relação ao mecanismo gerador de dados incompletos, as variações de MCAR têm diferentes impactos sobre a acurácia do modelo. Entretanto, o efeito depende de outras características do conjunto de dados, tais como tamanho da amostra e a quantidade de variáveis de entrada. Também discutimos alguns resultados interessantes sobre o impacto de dados incompletos sobre o poder preditivo das variáveis de entrada.

Palavras-chave: Dados Incompletos, Modelos Preditivos, Simulação de Dados, Regressão Logística, Análise Estatística, Qualidade de Dados

ABSTRACT

The quality of data is key in supporting data-centric systems, machine learning routines, and predictive models. Research on data quality aims to define, identify, and repair inconsistencies in the data. A common source of inconsistency is missing data, in which no data is stored for the variable in an observation, which potentially hides important information. The quality of the input and output variables have been neglected on the proposition of new predictive models, although the popularity of predictive analysis using machine learning tools has been increasing. As a consequence, the effect of missing data in many of the standard predictive models is completely unknown. We propose a stochastic framework to evaluate the impact of missing data on the performance of predictive models. The framework allows full control of important aspects of the data set structure such as the number and type of the input variables, the correlation between the input variables and their general predictive power, and sample size. The missing process is generated from a multivariate Bernoulli distribution, which allows us to simulate missing patterns corresponding to different levels of disturbance of the MCAR (Missing Completely at Random) mechanism. Although the framework may be applied to virtually all types of predictive models, in this article, we focus on the logistic regression model and choose the accuracy as the predictive measure. The simulation results show that the effects of missing data disappear for large sample sizes, as expected. On the other hand, as the number of input variables increases, the accuracy decreases mainly for binary inputs. With respect to mechanism that generate missing data, the levels of disturbance of MCAR has different impact on model accuracy. However, the effect depends on other characteristics of the data set, such as sample size and number of input variables. We also discuss some interesting results on the impact of incomplete data on the predictive power of input variables.

Keywords: Missing Data, Predictive Model, Data Simulation, Logistic Regression, Statistical Analysis, Data Quality

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ACRÔNIMOS

CINPUT	Correlação entre as Variáveis de Entrada
CO	Conjuntos de Dados
DE	Delineamento Experimental
DI	Dados Incompletos
MAR	*Missing at Random*
MCAR	*Missing Completely at Random*
MNAR	*Missing Not at Random*
NINPUT	Quantidade de Variáveis de Entrada
PMD	Proporção de Valores Ausentes
PPINPUT	Poder Preditivo das Variáveis de Entrada
QD	Qualidade de Dados
SS	Tamanho Amostral
TINPUT	Tipo das Variáveis de Entrada
TMD	Tipo do Mecanismo Gerador de Dados Incompletos
UFPR	Universidade Federal do Paraná

SUMÁRIO

1 INTRODUÇÃO

Nas últimas décadas, os avanços tecnológicos impulsionaram o uso massivo dos diversos tipos de dispositivos eletrônicos. Por exemplo, dispositivos móveis e pessoais, televisões digitais, câmeras de vigilância, pedágios, caixas eletrônicos, sensores de diversos tipos, dispositivos médicos, aplicativos conectados à nuvem, entre tantos outros. Deste modo, a era tecnológica moderna tornou-se um meio facilitador para a produção exponencial de dados.

De acordo com um relatório da *International Data Corporation* (IDC) [16], em 2011, o volume de dados do mundo cresceu em nove vezes em cinco anos, com estimativa de alcance em volume de 35 ZB em 2020. No entanto, estudos atuais [22, 51] apontam que esse volume já está beirando 44 ZB, sendo que em 2013 o volume total do universo digital era de 4,4 ZB. Consequentemente, esse volume de dados apresenta grandes desafios no quesito de coleta, aquisição, armazenamento, gerenciamento e manipulação. Além disso, o grande volume de dados aumenta a complexidade da extração de conhecimento. Sendo assim, o acesso a uma grande quantidade de dados contém valor apenas se os mesmos forem estruturados, ou seja, organizados de forma coerente de modo a otimizar o seu uso e, após sua estruturação, analisados corretamente.

Com os dados estruturados, o processo de extração de conhecimento é denominado análise de dados. A análise de dados usualmente é utilizada no suporte a sistemas centrados em dados, rotinas de aprendizado de máquinas e modelos preditivos, entre tantas outras abordagens. Devido à grande disponibilidade de informações ou dados, juntamente com a melhora do poder computacional, a análise de dados tem sido explorada cada vez mais pelas organizações [42]. Os *insights* fornecidos pela análise de dados tornaram-se um meio de justificar, guiar e prescrever ações, tanto em estratégias futuras, quanto na otimização das operações do dia-a-dia, orientando de forma eficaz uma tomada de decisão e refletindo diretamente na melhora da eficiência, desenvolvimento e vantagem competitiva na indústria [30].

Tanto na indústria quanto na comunidade científica, o potencial da análise de dados é reconhecida e apontada como um diferencial na economia atual [17, 25]. Entretanto, apesar da importância da extração de conhecimento a partir de dados, na maioria das vezes os dados apresentam diversos erros. Como por exemplo: valores ausentes, registros duplicados, domínios incorretos, entre outros. O problema de valores incorretos ocorre frequentemente na maioria dos conjuntos de dados atuais, impactando na veracidade dessa informação e causando divergência no registro armazenado em relação ao ambiente em que representa.

Os erros nos dados estão atrelados diretamente com a qualidade dos mesmos. A qualidade dos dados é um requisito fundamental para a aplicação de qualquer método de análise de dados. Os problemas de qualidade de dados podem ocorrer em qualquer parte do processo entre a geração, aquisição e armazenamento dos dados e devido ao cenário digital, esse problema se torna mais complexo, pois grande parte dos dados são provenientes de diferentes fontes e não estão estruturados.

A área de pesquisa da Qualidade de Dados (QD) visa garantir uma representação mais consistente de informação em relação ao ambiente real que esta foi extraída, de modo a maximizar o valor dessa informação, tornando os dados mais significativos para a análise e interpretação do usuário. Essa área ganhou uma atenção considerável, uma vez que as taxas de erro dos dados podem variar de 20% a 80% e acaba afetando as principais empresas do mundo [6, 28].

Muitas medidas têm sido tomadas para melhorar a qualidade dos dados. Nesse aspecto, muitos algoritmos computacionais já demonstraram robustez em relação a alguns tipos de erros

ou já possuem implementado seus próprios mecanismos de limpeza de dados [55, 6, 21, 40, 10]. Entretanto, ainda faltam métodos eficazes para realizar detecção e reparo automático de diferentes tipos de erros nos dados, além de suportar diferentes fontes e grandes conjuntos de dados [9].

O atual cenário digital torna o processo de limpeza de dados ainda mais desafiador. A variedade de dados e a complexidade dos tipos de erros dificulta a relação custo-eficácia, sendo ainda um processo oneroso com alto desperdício de recursos, podendo chegar a custar bilhões de dólares em perdas para as organizações [19, 24]. Além disso, afeta muitos cientistas de dados, que gastam entre 30% a 80% do tempo diagnosticando problemas de qualidade e estruturação de dados, em vez de usar esse tempo para extrair conhecimentos significativos[27].

1.1 MOTIVAÇÃO

Uma fonte comum de inconsistência de dados que impacta a QD é a ocorrência de Dados Incompletos (DI) representados por valores ausentes, em que nenhuma informação é armazenada no registro em questão, potencialmente escondendo informações importantes. Nesse contexto, mesmo com o aumento da popularidade da análise preditiva utilizando métodos de aprendizagem de máquina, a qualidade das variáveis de entrada e saída tem sido negligenciada na proposta de novos modelos preditivos.

Os estudos que avaliam o desempenho de modelos preditivos na presença de dados incompletos utilizam conjuntos de dados reais que já possuem valores ausentes ou que são inseridos artificialmente. Em ambos os casos, a análise é aplicada após a utilização de alguma técnica de tratamento de dados incompletos, seguida pelo ajuste de um modelo de previsão. O desempenho preditivo é normalmente comparado com os resultados do modelo sem valores ausentes. Entretanto, esse processo demonstra certas limitações, que dificultam a verificação da eficácia desses métodos utilizados para corrigir o problema de dados incompletos e seu impacto nas análises de dados. Assim, na maioria das vezes, não fica claro o real valor do método utilizado e a possibilidade de aplicação futura em diferentes domínios.

Outra limitação compartilhada por essas abordagens é que elas utilizam um baixo número de conjuntos de dados e, consequentemente, não utilizam métodos estatísticos para analisar os resultados. Além disso, mesmo fazendo o uso de conjuntos de dados reais, não é possível controlar os aspectos relativos aos dados incompletos (por exemplo, mecanismo, proporção) de forma extensiva e, consequentemente, suas conclusões não levam em conta a incerteza associada a estes aspectos, refletindo muitas vezes apenas os cenários específico que foram aplicados. Como consequência, o efeito dos dados incompletos em muitos dos modelos preditivos padrão é completamente desconhecido.

Portanto, existe uma lacuna na literatura em relação a pesquisas mais sistemáticas que controlam diversos fatores influenciáveis, em busca de resultados em um contexto mais genérico. Nesse ponto, os resultados auxiliariam também na identificação da necessidade do uso ou não da limpeza de dados, com a perspectiva de otimização de custos e recursos envolvidos [18, 4, 54, 36].

1.2 OBJETIVO

A análise do impacto gerado por dados incompletos, quando aplicados a modelos preditivos, é primordial para o desenvolvimento de novas ferramentas de tratamento, eficazes e abrangentes. Sendo assim, neste trabalho desenvolvemos um *framework* estocástico com o objetivo de avaliar o impacto dos dados incompletos no desempenho dos modelos preditivos. O *framework* é baseado na simulação dos conjuntos de dados, controlando aspectos das variáveis

de entrada e dos dados incompletos. Para atingir esse objetivo, nosso *framework* possui quatro etapas:

i) Geração dos conjunto de dados;

ii) Inserção artificial de valores ausentes;

iii) Ajuste de um modelo preditivo e mensuração de medidas de desempenho;

iv) Avaliação dos resultados usando uma ferramenta estatística.

O *framework* nos permite projetar uma ampla gama de cenários de simulação para que possamos avaliar o impacto de fatores importantes. Os fatores controlados são divididos entre características gerais de um conjunto de dados e características específicas relacionadas ao problema de dados incompletos. Entre os fatores gerais, está o tamanho da amostra, quantidade, tipo, poder preditivo e a correlação entre as variáveis de entrada. Em relação aos fatores específicos, voltados para a compreensão do impacto de dados incompletos, será controlado a proporção e o mecanismo gerador de valores ausentes.

1.3 ESTRUTURA DA DISSERTAÇÃO

O trabalho é organizado da seguinte forma: o Capítulo 2 descreve a fundamentação teórica necessária para embasar o desenvolvimento do *framework* proposto. Discutimos o estado da arte no Capítulo 3. A estruturação da metodologia utilizada para o desenvolvimento do *framework* e o delineamento dos experimentos é descrito e explicado no Capítulo 4. Os resultados do estudo de simulação e a discussão são apresentados no Capítulo 5. Por fim, o Capítulo 6 apresenta a conclusão do trabalho e as direções abertas de pesquisa para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Esse capítulo apresenta os conceitos teóricos necessários para o desenvolvimento do nosso *framework*, em relação à qualidade de dados, o problema de dados incompletos, modelos preditivos e métodos utilizados para o planejamento dos experimentos.

2.1 QUALIDADE DE DADOS

A definição de Qualidade dos Dados (QD) é especificada como sendo a precisão, uniformidade e completude alcançada em relação à representação dos aspectos do mundo que os dados descrevem, para seu uso adequado em uma abordagem analítica e em um contexto específico [14, 36, 29]. Algumas das dimensões de QD mais comuns entre os pesquisadores de gerenciamento de dados são:

- **Acurácia:** quando a representação dos dados é próxima o bastante dos aspectos do mundo que eles descrevem;
- **Consistência:** refere-se à violação de regras semânticas definidas sobre itens de dados e geralmente expressadas como restrições de integridade;
- **Unicidade:** é satisfeita quando os dados não contêm quaisquer duplicações;
- **Conformidade:** se os dados estiverem em conformidade com as convenções e padrões apropriados em relação ao seu domínio;
- **Completude:** quando não há valores ausentes, ou seja, valores omissos em um registro específico de um conjunto de dados.

Em geral, os erros nos dados acontecem entre o processo de aquisição e armazenamento, que geram valores divergentes de um determinado registro em relação à sua representação do mundo real, o que muitas vezes acaba invalidando os registros armazenados [1]. Grande parte deste problema de divergência é um reflexo do cenário digital atual e da falta de estruturação dos dados, em que os tipos e formatos de dados mudam frequentemente, seguido de problemas relacionados à integração por diversas fontes, erros de digitação, falta de padrão de dados, anomalias de atualização, entre outros [1, 11, 17].

A Figura 2.1 ilustra resultados fictícios de um estudo sobre o tabagismo na adolescência com atributos referente ao nome, idade, gênero, estado e região de residência e a quantidade de cigarros consumidos diariamente pelos participantes da pesquisa [47]. Os seis registros ilustrados pela Figura 2.1 destacam alguns tipos de erros de dados referenciados pelas dimensões de QD. A expectativa de vida humana é muito menor que 180 anos, o que indica que o registro t_4 está errado. O mesmo problema de domínio incorreto ocorre no registro t_5. Os registros t_3 e t_6 referem-se à mesma pessoa, mas o registro t_6 tem um erro ortográfico no atributo $Nome$. Se quaisquer dos registros concordarem com os valores de Estado, logo elas devem possuir o mesmo valor para Região, o que não ocorre entre os registros t_1 e t_4. Problemas com valores ausentes são encontrados nos registros t_2 e t_5.

A utilização de dados incorretos ou inconsistentes em uma abordagem analítica pode distorcer significativamente os resultados. Consequentemente, piora o desempenho da técnica utilizada, fornecendo interpretação errônea e afeta decisões tomadas com base nestes dados, ou

	Nome	Idade	Gênero	Estado	Região	Cigarros
t_1	Joana Silva	16	Feminino	Paraná	Norte	2
t_2	Paulo Mendes	17	Masculino	São Paulo	Sudeste	-
t_3	Luana Almeida	17	Feminino	Pará	Norte	4
t_4	Fernanda Souza	180	Feminino	Paraná	Sul	3
t_5	José Ferraz	05-09-1998	Masculino	Pernambuco	Nordeste	-
t_6	Luana Almmeida	17	Feminino	Pará	Norte	4

Domínio incorreto
Registro duplicado
Dependência de atributos
Valores ausentes

Figura 2.1: Tipos de erros de dados.

seja, degrada os principais benefícios. Entretanto, a mensuração deste impacto está relacionada ao contexto e à abordagem que está sendo aplicada aos dados [58, 11, 59]. Portanto, as consequências podem variar desde nenhum impacto negativo até resultados totalmente imprecisos, inválidos e inúteis.

Ao longo dos anos, várias comunidades científicas, como a de estatística, aprendizado de máquinas e banco de dados, trabalharam para desenvolver e melhorar as abordagens de limpeza de dados. A limpeza de dados é o processo utilizado para definir, identificar e reparar erros a partir de dados originais a fim de garantir e melhorar de forma confiável e eficiente a qualidade dos dados, quando possível utilizando um procedimento automático [46, 12, 48].

Um dos problemas mais comuns em QD é a ocorrência de dados incompletos [47, 39]. Esse problema faz parte da dimensão de **completude** e consiste naqueles atributos de um conjunto de dados com valores ausentes, sendo representado na Figura 2.1 pelos registros t_2 e t_5.

2.2 DADOS INCOMPLETOS

Um dos problemas mais comuns e relevantes da qualidade dos dados é a ocorrência de dados incompletos, sendo caracterizado por valores ausentes. Os valores ausentes são aqueles registros que não foram observados ou não armazenados no conjunto de dados por alguma razão, mas para as quais existe um valor real de representação no ambiente em que eles pertencem.

O problema de dados incompletos ocorre por diferentes razões, tais como perda ou erro em um procedimento de entrada manual, decorrentes de uma variedade de fontes, defeitos de equipamentos, da rede ou do sistema de banco de dados, coleta insuficiente de informações, recusa dos respondentes em responder certas questões, morte de pacientes, ou até mesmo devido a não recuperação de registros que sofrem de outros tipos de erros de dados, sendo a única alternativa descartá-los, entre outros [47, 39].

O conjunto de dados estruturado geralmente contém alguma proporção variável de dados incompletos e de acordo com Pyle, Acuna e Strike et al [43, 52, 2] uma taxa de menos de 1% dos valores ausentes é geralmente considerada trivial e pode simplesmente ser removida do conjunto de dados sem ter um efeito significativo no resultado da análise final. Entre 1 e 5 % é considerado manipulável. Entretanto, de 5% a 15% já requer métodos mais sofisticados para lidar com eles, e mais de 15% podem ter severos impactos sobre qualquer tipo de utilidade e é preciso considerar cuidadosamente o tratamento dessa quantidade de dados incompletos.

A presença de valores ausentes em um conjunto de dados pode estar disposto em duas configurações diferentes. A configuração univariada refere-se a quando apenas um atributo do

conjunto de dados sofre com o problema de valores ausentes. A configuração multivariada refere-se a quando mais de um atributo tem valores ausentes. Os dois tipos de configurações são mostrados na Figura 2.2.

(a) Univariada. (b) Multivariada.

Figura 2.2: Tipos de configurações de dados incompletos.

Litle e Rubin [32] apresentam uma teoria que determinou três tipos diferentes de mecanismos geradores de valores ausentes:

i) **MCAR** - *Missing Completely at Random*: Não existe nenhum padrão identificável nos dados incompletos;

ii) **MAR** - *Missing at Random*: Quando os valores ausentes dependem dos valores das observações que estão presentes;

iii) **MNAR** - *Missing Not at Random*: Quando os valores ausentes dependem das próprias observações ausentes do conjunto de dados.

A Tabela 2.1 ilustra o mesmo estudo sobre tabagismo na adolescência relatado na Seção 2.1. Porém, refere-se apenas a idade e a quantidade de cigarros consumidos diariamente por vinte participantes e um exemplo prático dos três tipos de mecanismos de dados incompletos descritos acima (MCAR, MAR e MNAR).

No caso MCAR, o dado é incompleto de forma aleatória, sem nenhum tipo de padrão. No exemplo de dados incompletos MAR, podemos ver que existe um padrão e, a ausência de valores possui uma relação direta com o atributo idade dos participantes, em que os mais novos foram os que não relataram a quantidade de cigarro consumida. No terceiro caso, os dados ausentes MNAR, o padrão depende do próprio atributo de quantidade de cigarros consumidos, em que os valores ausentes são justamente naqueles participantes que possuem um consumo alto. Porém, o padrão somente é percebido, pois é comparado com os valores completos.

Em um primeiro momento, o caso MNAR é muito semelhante ao MCAR, pois ambos não possuem um padrão aparente, por conta disso o tipo MNAR é o mais difícil de identificar na prática, pois o próprio pesquisador deve possuir conhecimento prévio ou algum tipo de evidência sobre esse caso [3].

Em uma descrição mais formal de dados incompletos [47], assumimos um conjunto de dados $\mathbf{X}$ representado por uma matriz n x p, onde $t = 1, ..., n$ registros e $j = 1, ..., p$ atributos, representados na Figura 2.3. Os elementos de $\mathbf{X}$ são denotados por $x_{t,j}$, cada atributo individual em $\mathbf{X}$ é denotado por x_j e cada registro é referenciado como $\mathbf{x}_t = [x_{t,1}, x_{t,2}, ..., x_{t,p}]$. Em modelagem preditiva, cada registro é representado por um valor na variável resposta $\mathbf{Y}$.

Tabela 2.1: Mecanismos geradores de dados incompletos.

Idade	Consumo Diário de Cigarros			
	Completo	MCAR	MAR	MNAR
15	2	2	-	2
15	9	-	-	-
15	4	-	-	4
16	2	2	-	2
16	2	2	-	2
16	7	7	-	-
16	3	3	-	3
17	9	-	9	-
17	6	6	6	-
17	4	-	4	4
17	5	5	5	5
17	5	5	5	5
18	7	-	7	-
18	6	6	6	-
18	7	-	7	-
19	3	3	3	3
19	8	-	8	-
19	3	-	3	3
20	9	9	9	-
20	2	2	2	2
n	20	12	13	11
Média	5.15	4.33	5.69	3.18

Em um cenário de dados incompletos, cada registro em um conjunto de dados possui um indicador de valores ausentes $\mathbf{m}_t = [m_{t,1}, m_{t,2}, ..., m_{t,p}]$, que indica quais atributos estão ausentes em cada registro $\mathbf{x}_t$, definindo assim um indicador geral de dados incompletos $\mathbf{M}$, sendo uma matriz binária, definida como:

$$M = \begin{cases} m_{t,j} = 1, \text{ se } x_{t,j} \text{ é observado} \\ m_{t,j} = 0, \text{ se } x_{t,j} \text{ é ausente} \end{cases} \tag{2.1}$$

Litle e Rubin [32] apresentaram uma teoria estabelecendo que a distribuição de probabilidade de $\mathbf{M}$ depende de $\mathbf{X} = (\mathbf{X}_{obs}, \mathbf{X}_{aus})$, em que $\mathbf{X}_{obs}$ contém todos os elementos $x_{t,j}$ onde $m_{t,j} = 0$ e $\mathbf{X}_{aus}$ contém todos os elementos $x_{t,j}$ onde $m_{t,j} = 1$. Essa teoria determinou três tipos diferentes de mecanismos que os valores ausentes podem ser distribuídos, sendo representados pela distribuição condicional de $\mathbf{M}$ dado $\mathbf{X}$, sendo $f(\mathbf{M}|\mathbf{X}, \phi)$, em que ϕ é um conjunto de parâmetros desconhecidos que ajudam a descrever a relação entre $\mathbf{M}$ e os dados [32]:

- Valores ausentes completamente aleatórios (MCAR - *Missing Completely at Random*): quando a probabilidade de um registro possuir um valor ausente para um determinado atributo não depende de nenhum outro valor do conjunto de dados, ou seja, não existe nenhum padrão identificável nos dados incompletos. Isso significa que os valores

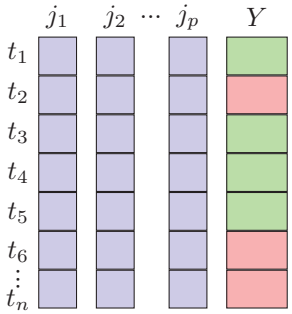


Figura 2.3: Conjunto de dados.

faltantes são uniformemente distribuídos, assim o impacto depende da quantidade de valores não observados.

$$f(\mathbf{M}|\mathbf{X}, \phi) = f(\mathbf{M}|\phi) \tag{2.2}$$

- Valores ausentes aleatórios (MAR - *Missing at Random*): quando a probabilidade de um registro possuir um valor ausente para um determinado atributo depende de algum outro valor do conjunto de dados, ou seja, quando a distribuição dos valores ausentes dependem dos valores das observações que estão presentes.

$$f(\mathbf{M}|\mathbf{X}, \phi) = f(\mathbf{M}|\mathbf{X}_{obs}, \phi) \tag{2.3}$$

- Valores ausentes não aleatórios (MNAR - *Missing Not at Random*): quando a probabilidade de um registro possuir um valor ausente para um determinado atributo depende das próprias observações ausentes do conjunto de dados. Este caso é similar ao MAR, porém mais complicado, pois as razões dos valores ausentes não são conhecidas e geralmente não são possíveis identificá-las.

$$f(\mathbf{M}|\mathbf{X}, \phi) = f(\mathbf{M}|\mathbf{X}_{aus}, \phi) \tag{2.4}$$

Ao longo dos anos, diversas técnicas para lidar com esse tipo de problema foram propostas na literatura. Uma abordagem convencional para lidar com dados incompletos consiste em remover os registros ou atributos que possuem valores ausentes, para criar um novo conjunto de dados totalmente completo. Apesar de sua simplicidade conceitual, informações úteis podem ser omitidas com este método.

Outra abordagem comum para resolver esse problema é a substituição dos valores ausentes de um conjunto de dados, por alguns valores plausíveis. Essa abordagem é conhecida como imputação de dados, sendo realizada por diversos métodos. A técnica consiste em estimar o valor ausente ou a distribuição desses dados, para gerar predições através de um determinado modelo [50, 57, 41, 44, 49, 2, 31].

No entanto, ainda a grande maioria dos métodos assumem que os dados são aleatoriamente incompletos [14, 39, 33, 35], o que em dados reais essa condição quase nunca é atendida [3]. Portanto, os mecanismos geradores de dados incompletos condicionam a validade das conclusões derivadas da técnica utilizada para lidar com os mesmos, dependendo assim do pressuposto de como os valores se tornaram ausentes em primeiro lugar [20, 47]. Além disso, a escolha de como lidar com os dados incompletos também deve ser baseada na porcentagem de dados que faltam e no tamanho da amostra.

A utilização de uma abordagem inadequada pode resultar em consequências negativas, assim como o não reparo do conjunto de dados. Os métodos de imputação, mesmo os mais rigorosos, podem introduzir distorções e gerar conhecimento errôneo. Por exemplo, o estudo sobre o tabagismo na adolescência, um método de imputação seria trocar os valores ausentes da Tabela 2.1 pelo valor da média amostral calculada em cada tipo de dados incompletos, o que pode gerar algumas distorções no comportamento do dado em relação ao mundo que ele de fato representa.

2.3 MODELOS PREDITIVOS APLICADOS A DADOS INCOMPLETOS

No desenvolvimento de um modelo preditivo, mais especificamente em um problema de classificação, um conjunto de dados de treinamento é fornecido como entrada e para cada caso desse conjunto existe uma classe pré-estabelecida. O modelo de classificação consiste em usar as informações contidas nos dados de treinamento para generalizar os padrões mapeados e inferir ou prever a classe de novos casos que não foram rotulados [33]. Consequentemente, a qualidade de um modelo preditivo está diretamente ligada com a qualidade do conjunto de dados utilizados na modelagem e no poder que o modelo possui em generalizar os padrões, ou seja, a redução do viés indutivo na predição dos dados anteriormente não observados [5].

A presença de dados incompletos induz a perda de informações relevantes, degradando a qualidade dos mesmos. A qualidade dos dados é primordial na aplicação de modelos preditivos e ao ser utilizado dados incompletos, além de resultar na maior complexidade de manipulação, gera incertezas na análise preditiva, impactando diretamente o desempenho do modelo em relação à precisão. A qualidade do modelo ajustado, consequentemente, afeta a eficiência dos resultados obtidos, dando lugar ao viés decorrente da diferença entre os dados faltantes e os completos [15, 33, 38, 7, 34, 8], influenciando negativamente a interpretação e tomada de decisão a partir dos resultados obtidos [56, 41].

Nesse aspecto, a qualidade dos dados possui relação direta com as características gerais do conjunto de dados, em relação à quantidade de registros e o tipo (categórico, binário, ordinal, numérico), além da correlação entre eles [33, 23]. Além dos aspectos gerais, as características específicas do problema de dados incompletos também influenciam os resultados do modelo preditivo, como por exemplo os tipos de mecanismos geradores de valores ausentes e a quantidade deles no conjunto de dados inteiro [7, 18, 8].

A grande maioria dos algoritmos de modelos preditivos implementados são baseados na suposição de que os dados são completos [57]. Alguns deles já possuem seus próprios mecanismos para lidar com o problema de valores ausentes, mas geralmente possuem uma limitação em relação à quantidade e a maioria assume que os valores ausentes são distribuídos aleatoriamente (MCAR), o que não é uma realidade encontrada nos banco de dados atuais [33].

Consequentemente, para desenvolver um procedimento de análise do desempenho de um modelo preditor em relação a dados incompletos, o processo de definição da combinação de características de entrada possui extrema importância, pois qualquer alteração afeta de alguma forma a saída do modelo [7]. Sendo assim, é bastante complexo encontrar um conjunto de características ideais capaz de reduzir a incerteza e garantir a qualidade dos modelos.

2.4 ESTUDO POR SIMULAÇÃO

De maneira geral, um experimento é um processo de investigação de uma hipótese, que através da modificação das variáveis de entrada, observa-se tanto o efeito produzido, quanto à relação entre as variáveis de entrada e a variável de resposta do experimento. Nesse contexto, a

metodologia de Delineamento de Experimentos (DE) é empregada com o objetivo de estruturar e conduzir o experimento sobre condições controladas. Para isso, o DE apresenta diversos métodos eficientes e econômicos, além de orientar qual seria a aplicação mais apropriada para a análise estatística e interpretação dos resultados obtidos, com a finalidade de alcançar conclusões sólidas e adequadas sobre a hipótese levantada.

A etapa de planejamento de um experimento é a de maior importância, pois sendo precisamente desenvolvida garante a qualidade nos resultados obtidos, além de assegurar os pressupostos para a aplicação de uma análise estatística predeterminada. O ponto inicial do planejamento é o reconhecimento e definição do problema em que a hipótese testada será delimitada e, assim, pode-se determinar qual será a variável resposta. As possíveis condições controláveis de um experimento são denominadas fatores. Cada fator envolvido no experimento pode ter diferentes valores chamados de níveis. A combinação entre os fatores são os tratamentos. O desenho experimental se encarrega de especificar quais níveis dos fatores e suas possíveis combinações serão usadas no experimento, como também a quantidade de unidades experimentais que serão submetidas aos diferentes tratamentos . Por fim, é realizado a análise de dados a partir do resultados, usando métodos estatísticos, geralmente regressão linear e ANOVA, seguido por conclusões práticas e recomendações através da validação dos resultados.

Ao realizar um experimento, o pesquisador deve assegurar três princípios: a *aleatorização*, *replicação* e a *blocagem*. A aleatorização das execuções dos tratamentos serve para proteger o experimento contra variáveis de incômodo desconhecidas que podem distorcer os resultados do experimento[13].

Em um experimento para comparar diferentes tratamentos, cada tratamento deve ser aplicado em diferentes unidades experimentais, assim se caracterizando o princípio de replicação, que tem como objetivo aumentar o tamanho da amostra e a precisão do experimento. A replicação é necessária porque as respostas das diferentes unidades variam, mesmo que as unidades sejam tratadas de forma idêntica. Se cada tratamento for aplicado apenas a uma única unidade, os resultados serão ambíguos - não podendo distinguir se a diferença nas respostas de duas unidades é causada pelos diferentes tratamentos, ou simplesmente pelas diferenças inerentes entre as unidades.

A blocagem é o processo de controle sistemático da variabilidade resultante da presença de fatores conhecidos que perturbam o sistema, mas que não se têm interesse em estudá-los, assim colocam-se as unidades experimentais em grupos (blocos) que são similares entre si. A aplicação dos três princípios evita a subjetividade do tratamento, diminui o erro sistemático causado por fatores incontroláveis, evita qualquer viés que pode corromper os dados e melhora a precisão com a qual as comparações entre os fatores de interesse são feitas, além da precisão da variável resposta. Tudo isso garante a utilização de métodos estatísticos, com a realização de uma análise integra dos resultados, validando e fortalecendo a conclusão obtida.

Nesse contexto, para entender o impacto dos dados incompletos em um determinado modelo preditivo, todos os fatores que possam exercer qualquer influência auxiliam na identificação de padrões, no entendimento do comportamento e mensuração da importância de cada informação perdida para os modelos preditivos. Consequentemente, estudos por simulação possuem essa capacidade de controlar precisamente um grande número de fatores e assegurar algumas visões precisas sobre um determinado processo e seu desempenho. Como resultado, é obtido maior flexibilidade de aplicação e de entendimento de um assunto em geral.

3 TRABALHOS RELACIONADOS

Esse Capítulo apresenta uma análise detalhada sobre diversos trabalhos que abordam com diferentes visões o problema de dados incompletos. Para cada trabalho relacionado à nossa proposta, discutimos as principais características e objetivos. Além disso, pontuamos as vantagens e limitações. Buscamos enfatizar as brechas encontradas na literatura para uma melhor definição do problema que abordamos ao longo do nosso estudo, como também a motivação de como iremos prosseguir para o desenvolvimento do *framework*.

De modo geral, os estudos que avaliam o impacto dos dados incompletos quando aplicados a modelos preditivos, seguem uma abordagem padrão, exemplificada na Figura 3.1. A abordagem consiste na utilização de conjuntos de dados reais que já possuem valores ausentes, ou que são inseridos artificialmente. Em ambos os casos, o conjunto de dados incompleto recebe algum tratamento voltado para limpeza de dados, mais especificamente alguma técnica de imputação dos valores ausentes é aplicada. Após o tratamento dos dados, é realizado o ajuste do modelo preditor. Por fim, é realizado uma análise dos resultados, sendo normalmente a comparação entre os resultados do modelos preditivo aplicado aos dados incompletos versus ao aplicado com os dados tratados.

Figura 3.1: Processo geral comumente aplicado na literatura.

3.1 ESTUDOS QUE UTILIZAM VALORES AUSENTES ORIGINAIS

Acuna and Rodriguez [2] analisam o desempenho de dois modelos preditivos, aplicados a dados que já possuem valores ausentes e que foram reparados por quatro métodos de imputação. Foram utilizados 12 conjuntos de dados que foram coletados do repositório UCI[1] e as principais características estão sumarizadas na Tabela 3.1. A coluna CD representa os conjuntos de dados e n o tamanho amostral. A coluna Variáveis representa a quantidade de variáveis de entrada, equanto que o valor dentro dos parênteses representam a quantidade de variáveis relevantes.

Em [34] os autores realizam um dos trabalhos mais extenso da área de dados incompletos. Foi analisado vinte e três modelos preditivos e quatorze métodos de imputação. Para a aplicação foi utilizado vinte e um conjunto de dados, coletados do repositório UCI. Os conjuntos de dados continham originalmente valores ausentes e suas principais características estão sumarizadas na Tabela 3.2. Os autores dividem os modelos preditivos em três grupos e, a partir da aplicação do teste de Wilcoxon, encontram a melhor opção de técnica de imputação global para cada grupo. O trabalho conclui que o desempenho de cada grupo melhora com determinado tipo de imputação. No entando, a utilização de valores ausentes originais resulta em confundimento da real eficácia tanto dos métodos de imputação analisados, quanto do poder de generalização do modelo preditor.

[1] `archive.ics.uci.edu/ml/`

Tabela 3.1: Conjuntos de dados utilizados no estudo [2].

CD	n	Variáveis	CD	n	Variáveis
Iris	150	4(3)	*Crx*	690	15(9)
Hepatitis	155	19(10)	*Breastw*	699	9 (5)
Sonar	208	60(37)	*Diabetes*	768	8(5)
Heartc	303	13	*Vehicle*	846	18(10)
Bupa	345	6(3)	*German*	1000	20(13)
Ionosphere	351	34(21)	*Segment*	2310	19(11)

No trabalho desenvolvido por Jerez et al. [26], oito métodos de imputação foram utilizados. O conjunto de dados é referente a uma base médica contendo informações sobre pacientes com câncer. O tamanho amostral é de 3679, com 8 variáveis de entrada tanto do tipo categóricas, quanto contínuas. O desempenho dos métodos de imputação são analisado, após a aplicação de um modelo preditor. O diferencial desse estudo é a utilização do teste de Friedman para testar estatisticamente os efeitos globais dos métodos de imputação.

O estudo [48] utiliza apenas um conjunto de dados, com 25 variáveis de entrada, contendo variáveis contínuas e categóricas e tamanho amostral de 400. O conjunto de dados foi coletado do repositório UCI e contém valores ausentes originais. O estudo realiza a aplicação de quatro modelos preditivos, após a aplicação de dois métodos de imputação para a reparação dos valores ausentes. Foi realizado uma fase de seleção de variáveis, porém a grande maioria teve importância na explicação da resposta e foi mantida na modelagem. O objetivo é a mensuração do desempenho dos métodos de imputação, não sendo avaliado o impacto dos dados incompletos.

3.2 ESTUDOS QUE REALIZAM INSERÇÃO ARTIFICIAL DE VALORES AUSENTES

O trabalho [47] apresenta um revisão de literatura, dos métodos que foram utilizados para a inserção artificial de valores ausentes. O trabalho considera os três mecanismos geradores de dados incompletos (MCAR, MAR e MNAR) e pontua as vantagens e desvantagens de cada técnica, sendo de grande importância para o entendimento das abordagens aplicadas na literatura.

Batista and Monard [5] optaram por inserir valores ausentes artificialmente nos conjuntos de dados, com o objetivo de obterem controle total sobre os aspectos dos dados incompletos. O estudo analisa os benefícios da utilização do modelo KNN como método de imputação,

Tabela 3.2: Conjuntos de dados utilizados no estudo [34].

CD	n	Variáveis	CD	n	Variáveis
Cleveland	303	14	*Wisconsin*	699	10
Credit	689	16	*Breast*	286	10
Autos	205	26	*Primary tumor*	339	18
Dermatology	365	35	*House-votes-84*	434	17
Water-treatment	526	39	*Sponge*	76	46
Bands	540	40	*Horse-colic*	368	24
Audiology	226	71	*Lung-cancer*	32	57
Hepatitis	155	20	*Mushroom*	8124	23
Post-operative	90	9	*Echocardiogram*	132	12
Soybean	307	36	*Mammographic*	961	6
Ozone	2534	73			

comparando com outros três métodos. Foi utilizado 4 conjuntos de dados, coletados do repositório UCI. Os tamanhos amostrais dos conjuntos de dados variam entre 345 e 1473 registros. São utilizadas variáveis de entrada tanto contínuas, quanto categóricas, porém a quantidade não passa de um total de 9. A inserção artificial de valores ausentes é gerada através do mecanismo MCAR.

Tabela 3.3: Conjuntos de dados utilizados no estudo [45].

CD	n	Variáveis	CD	n	Variáveis
Abalone	4177	8	*Breast Cancer*	699	9
BMG	2295	40	*CalHouse*	20640	8
Car	1728	6	*Coding*	20000	15
Contraceptive	1473	9	*Credit*	690	15
Downsize	1277	15	*Etoys*	270	40
Expedia	500	40	*Move*	3029	10
PenDigits	10992	16	*Priceline*	447	40
QVC	500	40			

O estudo [45] utilizou 15 conjuntos de dados, coletados do reporitório UCI, com suas principais características sumarizadas na Tabela 3.3. Os autores inseriram artificialmente valores ausentes nos conjuntos de dados, através do mecanismos MCAR. O estudo teve como objetivo a análise do desempenho de três tipos de métodos de reparação de dados incompletos. Os resultados do estudo estão dispostos na Figura 3.2 e demonstra a variabilidade obtida, refletida pelas diferença entre os conjuntos de dados utilizados.

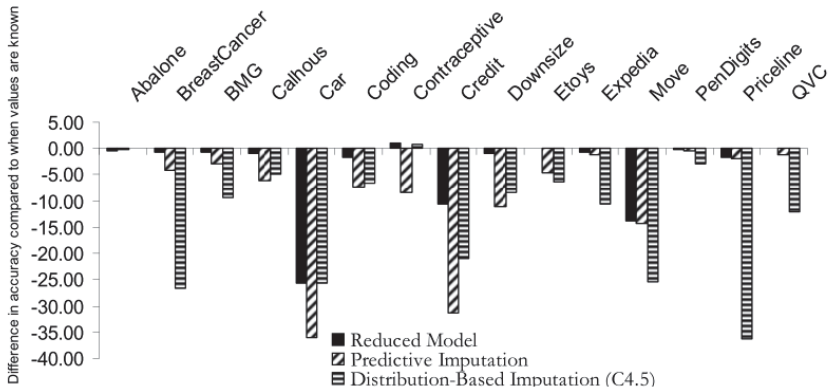


Figure 1: Relative differences in accuracy (%) between prediction with each missing data treatment and prediction when all feature values are known. Small negative values indicate that the treatment yields only a small reduction in accuracy.

Figura 3.2: Resultados apresentados pelo estudo [45].

O estudo realizado por Poulos and Valle [41] analisa o desempenho de três modelos preditivos, aplicados a conjuntos de dados com valores ausentes apenas em variáveis do tipo categórica. Os dois conjuntos de dados utilizados foram coletados do repositório UCI. O primeiro conjunto de dados contém o tamanho amostral de 48842, com 14 variáveis de entrada. O segundo conjunto de dados contém o tamanho amostral de 435, com 16 variáveis de entrada. Os autores inserem valores ausentes artificialmente, utilizando o mecanismo gerador MCAR. Seis métodos de imputação foram testados.

O estudo [18], os autores inserem artificialmente diferentes porcentagens de valores ausentes, utilizando os três mecanismos geradores de dados incompletos. Na aplicação, dez conjuntos de dados reais são utilizadas no ajuste de quatorze modelos preditivos. Os autores analisam o comportamento resultante dos modelos preditivos, após a realização da imputação dos valores ausentes. Esse é o padrão mais frequente aplicado pela literatura. Seguindo a mesma linha, os estudos [20, 33, 7, 54] apresentam resultados do desempenho dos métodos de imputação.

No trabalho desenvolvido em [50], são utilizados 6 conjuntos de dados, com valores ausentes inseridos artificialmente, através dos três tipos de mecanismos geradores de dados incompletos. O estudo analisa se a tolerância de um algoritmo preditivo que atua bem com dados incompletos, melhora ao ser aplicada uma técnica de imputação. No mesmo contexto de modelos robustos a dados incompletos, a tese em [35] trata-se do desenvolvimento de modelos preditivos que possuam seus próprios processos para lidar com dados incompletos.

3.3 ESTUDOS QUE REALIZAM SIMULAÇÃO DOS CONJUNTOS DE DADOS

Poucos estudos na área de dados incompletos optam por utilizar a simulação dos conjuntos de dados. Em [44], esse método é utilizado com o benefício de controlar alguns aspectos importantes, como o número de variáveis de entrada, tamanho amostral e diferentes níveis de correlação entre as variáveis de entrada. O conjunto de dados simulado contém 10 variáveis de entrada e tamanho amostral de 100. A simulação foi realizada através de reamostragem com 100 réplicas. Os valores ausentes foram gerados atificialmente, através do mecanismo MCAR. Foram utilizado 5 métodos de imputação. O ponto negativo desse estudo é que foi simulado apenas para um cenário experimental, referente às características do conjunto de dados.

O estudo [37] realiza a aplicação em 6 conjuntos de dados reais e 1 simulado. O conjunto de dados simulado contém 4 variáveis de entrada e tamanho amostral de 210. A simulação é feita através da geração de números aleatórios entre o intervalo 0 e 1. O ponto negativo do estudo é que ele analisa o impacto do ruído em geral, não aplicando no contexto de dados incompletos. Da mesma forma, a estruturação da metodologia utilizada não foi descrita minuciosamente e pouco pode ser aproveitado para a simulação de dados.

3.4 DISCUSSÃO

O uso de conjuntos de dados reais é altamente pertinente para a demonstração da efetividade dos resultados. Entretanto, a grande questão na abordagem padrão da literatura, é que com o uso de conjuntos de dados reais, dificilmente se tem o controle de diversas características. Consequentemente, essa falta de controle pode influenciar no poder de generalização de um modelo preditivo e nos resultados obtidos. Além disso, com a abordagem padrão, os efeitos nos resultados ficam confundidos, entre a eficácia do método de imputação aplicado ou a influência de alguma característica específica dos conjuntos de dados. Consequentemente, o efeito dos dados incompletos é desconhecido e o real valor de uma técnica de imputação não fica clara na maioria das análises preditivas.

Outra limitação compartilhada por estas abordagens da literatura é que elas utilizam um baixo número de conjuntos de dados e não garantem os pressupostos necessários para a utilização de métodos estatísticos, na confirmação dos resultados obtidos. Muitos deles, também não controlam os aspectos relativos aos dados incompletos (por exemplo, proporção e mecanismos gerados) e, consequentemente, suas conclusões não levam em conta a incerteza associada a estes aspectos.

A caracterização do problema e a definição de todos os aspectos influenciáveis são etapas fundamentais para a condução de um experimento de forma precisa. Esse processo tem como objetivo alcançar todos os requisitos necessários para a aplicação eficaz de métodos estatístico e assim confirmar os resultados obtidos. Isso também garante que o conjunto de dados utilizado seja representativo e que os resultados possuam capacidade de refletir quais os principais fatores de influência para a obtenção de resultados mais precisos e com maior poder de generalização. Essa é a principal lacuna que encontramos na literatura e que levamos como motivação no desenvolvimento do nosso estudo.

Na Tabela 3.4, realizamos uma sumarização dos trabalhos relacionados que são pontuados na primeira coluna. As abreviações CD e DI significam, respectivamente, Conjuntos de Dados e Dados Incompletos. A coluna CD descreve a quantidade e se o conjunto de dados é real ou sintético. A coluna DI descreve se os valores ausentes foram inseridos artificialmente ou são utilizados os originais. Os tipos de mecanismos geradores de dados incompletos são descritos na coluna Tipo DI, seguido da coluna que descreve o objetivo da abordagem utilizada.

Tabela 3.4: Análise comparativa entre os trabalhos relacionados.

Referência	CD	DI	Tipo DI	Abordagem
Acuna and Rodriguez [2]	Real (12)	Real	*	Impacto da imputação
Batista and Monard [5]	Real (4)	Artificial	MCAR	Impacto da imputação
Blomberg and Ruiz [7]	Real (10)	Artificial	MCAR	Impacto DI
Garciarena and Santana [18]	Real (10)	Artificial	3 tipos	Impacto imputação
Gill et al. [20]	Real (1)	Artificial	MAR	Impacto DI
Jerez et al. [26]	Real (1)	Real	*	Impacto da imputação
Liu et al. [33]	Real (10)	Artificial	MCAR	Impacto da Imputação
Luengo et al. [34]	Real (21)	Real	*	Impacto da imputação
Marlin [35]	Real/Sintética	Aritificial	3 tipos	Algoritmos robustos
Nettleton et al. [37]	Sintético/ Real	Artificial	Perturbação	Impacto da perturbação
Poulos and Valle [41]	Real (2)	Artificial	MCAR	Impacto da imputação
Richman et al. [44]	Sintético	Artificial	MCAR	Impacto da imputação
Saar-Tsechansky and Provost [45]	Real (15)	Artificial	MCAR	Impacto da imputação
Santos et al. [47]	-	Artificial	3 tipos	Tipos de DI
Sessa and Syed [48]	Real (1)	Real	*	Impacto da imputação
Song et al. [50]	Real (6)	Artificial	3 tipos	Impacto da imputação
Twala [54]	Real (21)	Artificial	3 tipos	Algoritmos robustos

* O autor não deixa claro qual tipo de mecanismo gerador de dados incompletos foi utilizado

4 FRAMEWORK PARA ANÁLISE DO IMPACTO DE DADOS INCOMPLETOS EM MODELOS PREDITIVOS

Nesse Capítulo, apresentamos uma visão geral do nosso *framework*, explicando cada parte do processo que foi realizado. Na sequência, apresentamos o detalhamento da metodologia aplicada para alcançarmos os objetivos do estudo.

Nosso estudo teve como objetivo o desenvolvimento de um *framework* para a análise do impacto de dados incompletos em modelos preditos. Para isso, foi realizado um extenso estudo por simulação, seguindo a metodologia DE, com a finalidade de manter o maior número de fatores influenciáveis sob controle e obter resultados precisos sobre o tema abordado. Nosso *framework* é dividido em quatro fases, demonstrado pela Figura A.10), que são: simulação dos conjuntos de dados, inserção artificial de valores ausentes, aplicação do modelo preditivo e a análise estatística dos resultados obtidos. Nossa abordagem está em nítido contraste ao processo padrão utilizado na literatura, que foi demonstrado na Figura 3.1.

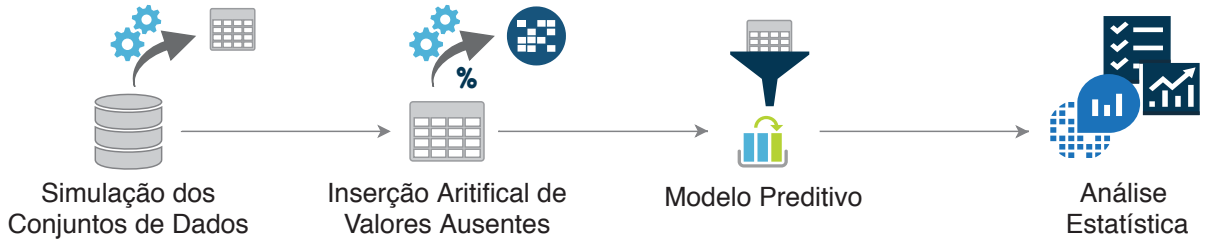


Figura 4.1: Processo geral realizado para o desenvolvimento do *framework*.

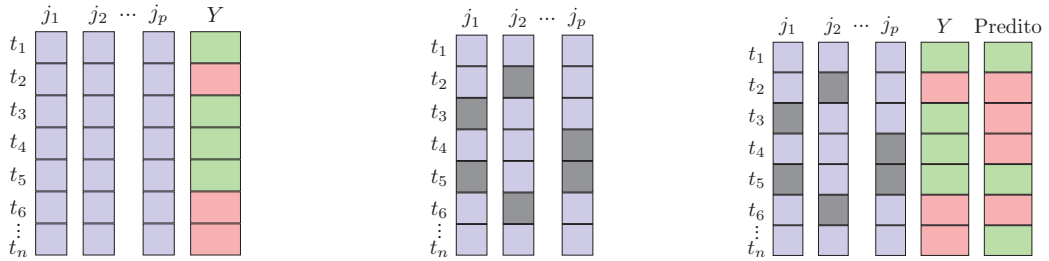
Na primeira fase, o objetivo é gerar diversos conjuntos de dados artificiais, sob condições controladas. Por exemplo, tamanho amostral, quantidade e tipo de variável de entrada, além da correlação entre elas e o poder preditivo. A segunda fase irá inserir artificialmente valores ausentes nos conjuntos de dados gerados na primeira fase, a partir dos três tipos de mecanismos geradores de dados incompletos, mantendo também sob controle a quantidade de valores ausentes.

Nas duas primeiras fases, nosso estudo busca controlar o maior número de características influenciáveis para uma modelagem preditiva. Esse controle irá garantir que ao realizarmos a terceira fase, e ao aplicarmos um determinado modelo preditivo, os resultados serão mais precisos em relação ao nosso objetivo de compreender o real impacto dos mecanismos de dados incompletos. As informações resultantes serão mais abrangentes e garantem os pressupostos para a realização da análise estatística. Para fins de reprodução, os códigos utilizados estão disponíveis em um repositório público[1].

4.1 EXPERIMENTOS

Para a execução dos experimentos, nós utilizamos uma máquina com processador Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 4-Core de 64 bits, com suporte a 2 *threads* por core. A máquina possui 8GB de memória RAM, com 1000 GB de armazenamento em disco rígido sobre o sistema operacional Ubuntu, versão 18.04.4 LTS (Bionic Beaver). Todos os

[1] `https://github.com/fsantore/master_degree_code`



(a) Simulação dos conjuntos de dados. (b) Inserção artificial de valores ausentes. (c) Aplicação do modelo preditor.

Figura 4.2: Etapas do *framework*.

experimentos foram realizados exclusivamente nesse ambiente, utilizando o software estatístico R.

4.1.1 Simulação dos Conjuntos de Dados

O desempenho de um modelo preditivo depende dos aspectos gerais do conjunto de dados utilizado, logo, pode influenciar os resultados da análise do impacto dos dados incompletos. Assim, propomos controlar o tamanho da amostra, número e tipo das variáveis de entrada e a correlação entre essas variáveis. Estas características nomeamos de fatores e os valores que eles podem assumir de níveis. Dado $\mathbf{X}$ uma matriz de $n \times p$ variáveis de entrada, demonstrada na Figura 4.2(a).

Fixamos o tamanho da amostra em $n = 500, 1000$ e 10000, e para as variáveis de entrada, fixamos $p = 10, 50$ e 200. Para explorar o impacto do tipo de variáveis de entrada, projetamos três cenários: i) as variáveis de entrada são binários gerados a partir de uma distribuição Bernoulli com probabilidade de sucesso igual a 0.5, ii) as variáveis de entrada são inteiros gerados de uma distribuição Poisson com parâmetro igual a 10 e iii) as variáveis de entrada são contínuas geradas a partir de uma distribuição Gaussiana padrão. As três distribuições de probabilidade utilizadas para gerar os tipos de variáveis são exemplificadas pela Figura 4.3.

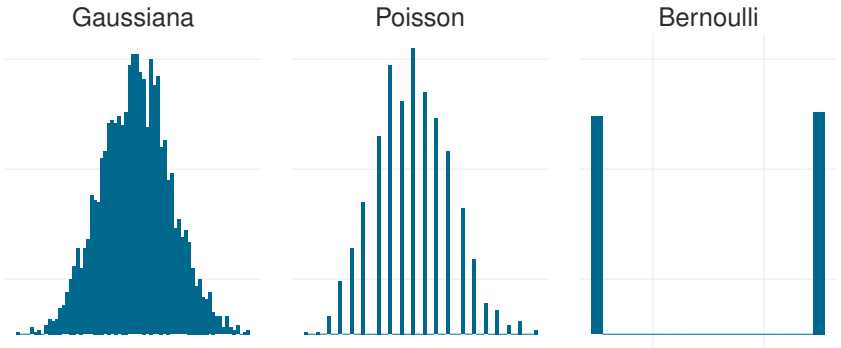
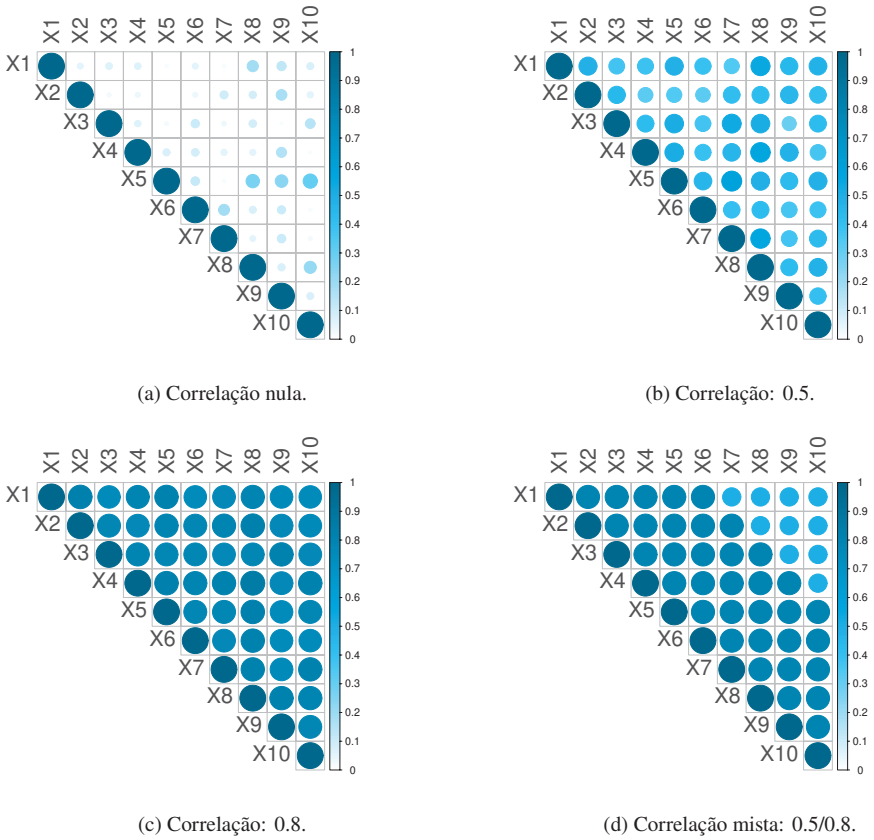


Figura 4.3: Distribuições amostrais de probabilidade.

Para simular variáveis de entrada não Gaussianas correlacionadas, utilizamos o algoritmo NORTA (*Normal to Anything*) que é implementado no pacote SIMCORMULTRES para o software estatístico R [53]. Para um nível crescente de redundância entre as variáveis de entrada, consideramos três níveis de correlação $\rho = 0, 0.5$ e 0.8, demonstrados na Figura 4.4 (a,b,c). A Figura 4.4 exemplifica a correlação para um conjunto de dados com dez variáveis de entrada, sendo estentido, igualmente, para outras quantidades de variáveis de entrada.



(a) Correlação nula.

(b) Correlação: 0.5.

(c) Correlação: 0.8.

(d) Correlação mista: 0.5/0.8.

Figura 4.4: Níveis de correlação.

Para simular a variável resposta, selecionamos a estrutura do modelo de regressão logística por causa de sua popularidade em problemas de classificação. Entretanto, outros modelos preditivos poderiam ser avaliados de forma semelhante. Neste contexto, dado um conjunto de variáveis de entrada representadas numa matriz $\mathbf{X}$ representado pela Figura 4.2(a), a variável resposta Y_i é simulada com base no modelo logístico,

$$Y_i \sim \mathrm{B}(p_i) \tag{4.1}$$

$$p_i = \mathrm{E}(Y_i|X_{ij}) = \mathrm{P}(Y_i = 1) = \frac{exp(X_{ij}^{\top}\boldsymbol{\beta})}{1 + exp(X_{ij}^{\top}\boldsymbol{\beta})}. \tag{4.2}$$

O vetor $p \times 1$ dos coeficientes de regressão $\boldsymbol{\beta}$ é simulado a partir de uma distribuição geométrica, cujo parâmetro α controla o poder preditivo das variáveis de entrada, ou seja, $\beta_j \sim \Delta G(\alpha)$ para $j = 1, \ldots, p$. Dada a estrutura da distribuição geométrica, a primeira variável de entrada é a mais importante e a importância decresce exponencialmente, como é demonstrado através da Figura 4.5.

Além disso, incluímos um parâmetro de ajuste extra Δ para controlar a acurácia esperada do modelo preditivo. O Algoritmo 1 mostra como selecionar Δ, que é fundamental para manter a comparabilidade dos resultados entre os diferentes cenários de simulação. Nesse processo de calibração, selecionamos um Δ para cada cenário de simulação de modo que a análise de todo o conjunto de dados resulte em 90% de acurácia.

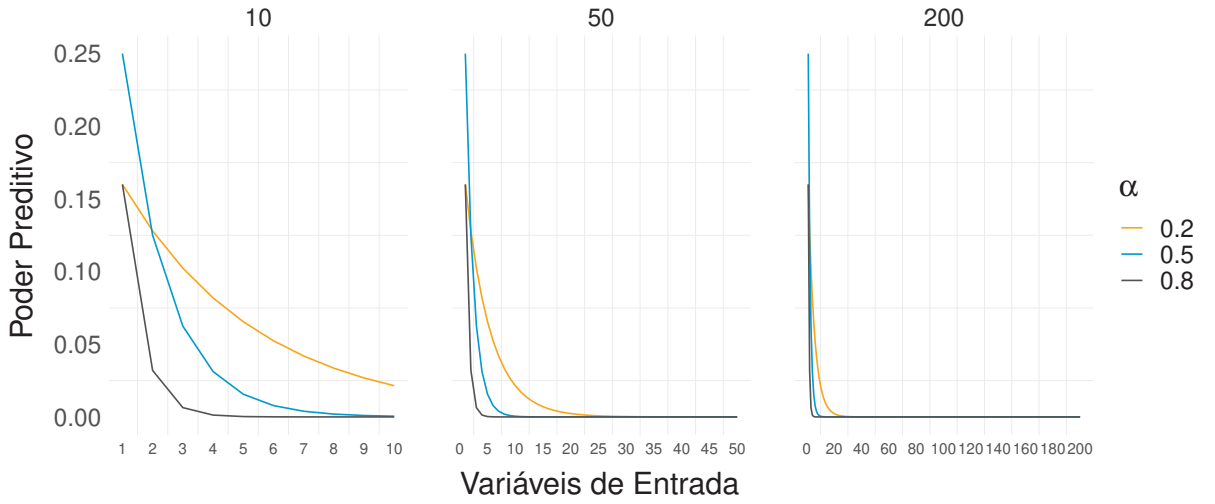


Figura 4.5: Poder preditivo.

Embora, Δ seja selecionado para cada cenário de simulação, podendo ser representado visualmente pela variação da altura da curva na Figura 4.5. O parâmetro α foi fixado em 0.2, 0.5 e 0.8, representando um cenário onde temos uma distribuição indo de uniforme até uma assimétrica para os coeficientes de regressão, sendo visualmente a inclinação da curva na Figura 4.5.

4.1.2 Inserção Artificial de Dados Incompletos

A segunda fase do *framework* controla o mecanismo gerador e a quantidade de dados incompletos inseridos nos conjuntos de dados simulados na primeira fase. Para gerar valores ausentes, utilizamos a configuração multivariada (Figura 2.2(b)) e diferentes variações do mecanismo MCAR exemplificado na Seção 2.2 .Para a simulação da matriz $\mathbf{M}$, utilizamos diferentes especificações na matriz de correlação de uma distribuição Bernoulli multivariada.

A simulação do mecanismo MCAR é feita através da simulação da matriz $\mathbf{M}$ segundo distribuições Bernoulli independentes, em que a probabilidade de sucesso determina a quantidade de valores ausentes no conjunto de dados. A simulação das demais variações do mecanismos MCAR é mais complexa e são denominados MCAR2 e MCAR3. Neste estudo, adotamos uma estratégia baseada em uma distribuição Bernoulli multivariada, que, por sua vez, foi especificada por um vetor $p \times 1$ de probabilidades e uma matriz de covariância de $p \times p$.

No mecanismo MCAR2, a distribuição dos valores ausentes é controlada pelos valores das variáveis de entrada observadas. Como consequência, as colunas da matriz $\mathbf{M}$ são correlacionadas, já que são geradas com base no mesmo conjunto de $\mathbf{X}$. Assim, simulamos $\mathbf{M}$ a partir de uma distribuição multivariada Bernoulli fixando o parâmetro de correlação em $\rho = 0.5$ e 0.8, demonstrado pela Figura 4.4 (b,c). Note que, $\rho = 0$ corresponde ao mecanismo MCAR padrão.

A simulação do MCAR3 é a variação mais complexa, pois considera que o padrão de dados incompletos depende de variáveis de entrada não observadas. Para isso, introduzimos uma suposição extra de que, embora os padrões ausentes dependam de variáveis não observadas, estas variáveis não observadas ou latentes induzem um padrão especial na matriz de covariância da distribuição multivariada Bernoulli. A ideia é semelhante à análise de fatores, em que o padrão de covariância é interpretado como efeito de variáveis latentes.

Especificamos a matriz de covariância da distribuição multivariada de Bernoulli como uma matriz de bloco diagonal, onde cada bloco é atribuído ao efeito de uma variável latente. Para simplificar, dividimos a matriz de covariância em dois blocos de tamanho igual. No primeiro

Algoritmo 1: Algoritmo de Calibração

Entrada: poder preditivo das variáveis de entrada (α);
Saída: O tamanho do efeito para garantir 90% de acurácia, dado condições da simulação;
início
  para $u = 0$ ***até*** *10* **faça**
    gera X usando a função *rnorta*, dado condições da simulação;
    para $k = 0$ ***até*** *15* **faça**
      repita
        $\Delta \leftarrow$ Procura-se a raiz da função, entre o intervalo (0,30), que resulta em 90% de acurácia ;
        gera o vetor $\beta_j \sim G(\alpha)$ para $j = 1, \ldots, p$;
        $\beta' \leftarrow \Delta * \beta_j / sum(\beta_j)$;
        gera $Y_i \sim B(p_i)$ visto na Equação 4.2;
        $mod \leftarrow$ ajusta o modelo de regressão logística;
        acc $\leftarrow$ acurácia do mod;
      até $acc = 0.9$;
      $ef \leftarrow$ armazena cada valor de Δ quando $acc = 90\%$;
    fim
    $efeitos \leftarrow$ armazena cada vetor de ef;
  fim
  $TamanhoEfeito \leftarrow mean(efeitos)$;
  return($TamanhoEfeito$);
fim

bloco, usamos a correlação $\rho = 0.5$, e no segundo bloco usamos $\rho = 0.8$, demonstrado pela Figura 4.4 (d). Finalmente, combinamos exaustivamente todos os fatores e seus níveis para compor 2187 cenários de simulação.

4.1.3 Aplicação e Mensuração do Desempenho do Modelo Preditivo

Na primeira fase, foi realizado a simulação dos conjuntos de dados. Na segunda fase, foi realizado a inserção artificial de valores ausentes nos conjuntos de dados simulados. Por fim, a terceira e quarta fase, respectivamente, foi realizado o ajuste do modelo com a mensuração do desempenho e análise estatística dos resultados.

Para o ajuste do modelo, utilizamos o software estatístico `R` e a função glm() para aplicar o modelo de regressão logística. Para cada cenário de simulação (2187), replicamos o tratamento para 150 conjuntos de dados. Cada conjunto de dados foi dividido em conjunto de treino (70%) e teste (30%), realizando o ajuste do modelo na população de treino e validando com a população de teste. Para mensurar a performance do modelo ajustado, utilizamos como métrica principal a acurácia. No entanto, levamos em consideração a mensuração do F1 score, como medida de sinalização do balancemaneto dos conjuntos de dados simulados.

Para realizar a análise estatística, utilizamos o modelo de regressão linear múltipla, sendo específicado na forma matricial como:

$$\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon} \tag{4.3}$$

Em que $\mathbf{y}$ é o vetor de variáveis respostas, representado pela acurácia ou F1 Score do modelo. Os fatores controlados na simulação são representados pelo vetor $\mathbf{X}$ e o erro aleatório (devido ao acaso) representado pelo vetor $\boldsymbol{\varepsilon}$. O vetor $\boldsymbol{\beta}$ representa os parâmetros que foram estimados e que nos permitiu quantificar o efeito principal de cada fator controlado, na acurácia do modelo de regressão logística. Como medida de qualidade do ajuste do modelo de regressão linear múltipla, ou seja, para mensurar o quão próximos os dados estão da linha de regressão ajustada, utilizamos o coeficiente de determinação (R^2).

5 RESULTADOS

Neste Capítulo, apresentamos os resultados do nosso estudo de simulação. A Figura 5.1 e a Figura 5.2 apresentam a visão geral de nossos principais resultados[1]. Para uma melhor visualização dos resultados, optamos por apresentar no gráfico os resultados usando barras que representam o primeiro e o terceiro quantil. Assim, avaliamos os resultados levando em conta a incerteza associada a eles em cada cenário.

No gráfico, o eixo y esquerdo representa a acurácia. A linha vermelha pontilhada na horizontal representa a acurácia esperada de 90%, que foi definida no processo de calibração na Subseção 4.1.1. A quantidade de variáveis de entrada é apresentada no eixo y direito e o tipo das variáveis de entrada representado pelas cores das barras. Em relação à correlação entre as variáveis de entrada, mostramos apenas o cenário mais desafiador, ou seja, a correlação de 0.8.

O eixo x inferior do gráfico representa o poder preditivo, enquanto que o eixo x superior representa o tamanho amostral, juntamente com os mecanismos geradores de dados incompletos, em que apresentamos um cenário para cada um deles, ou seja, completo (COM), MCAR, MCAR2 com correlação de 0.5 e MCAR3 com correlações de 0.5 e 0.8. Na Figura 5.1, mostramos os casos em que simulamos 30% de valores ausentes.

Em geral, os resultados na Figura 5.1 mostram que, para amostras grandes, os efeitos dos dados incompletos desaparecem para todos os mecanismos geradores de dados incompletos. Além disso, a precisão da acurácia aumenta, ou seja, as barras ficam mais estreitas. Esse é um resultado esperado, dado que quanto maior a amostra, mais informações se têm sobre e, consequentemente, maior precisão e confiabilidade é esperado dos resultados.

Por outro lado, quando o número de variáveis de entrada aumenta, a acurácia diminui rapidamente e, consequentemente, o poder de generalização do modelo é fraco. A combinação de pequenas amostras e um grande número de variáveis de entrada é o pior cenário em termos de acurácia, sendo também os mais afetado pelos mecanismos MCAR2 e MCAR3.

Em relação ao tipo de variáveis de entrada, notamos um aumento na acurácia em relação a Bernoulli para Poisson e Gaussiana, respectivamente. Da mesma forma, a acurácia tende a diminuir dos mecanismos MCAR para MCAR2 e MCAR3. Em geral, o mecanismo MCAR3 mostra os piores resultados em termos de acurácia em todos os cenários considerados. No entanto, quando combinado com grandes amostras e um baixo número de variáveis de entrada, a acurácia mal está abaixo do nível de 90%.

O poder preditivo das variáveis de entrada é um fator importante para determinar o impacto dos dados incompletos no desempenho preditivo do modelo. Em geral, quando o poder preditivo é concentrado em algumas variáveis de entrada contínuas (Gaussianas) ($\alpha = 0.8$), a acurácia é menos afetada pelos dados incompletos do que nos outros cenários. Por outro lado, para entradas binárias, uma distribuição mais uniforme do poder preditivo implica em um efeito menor dos mecanismos geradores de dados incompletos.

A Figura 5.2 apresenta os mesmo parâmetros da Figura 5.1, porém a quantidade de valores ausentes simulados é de 10%. O mesmo padrão de comportamento foi encontrado, entretanto, com a menor quantidade de valores ausentes as barras ficam mais estreitas, apontando o aumento na precisão da acurácia e, consequentemente, a maior confiabilidade nos resultados.

Finalmente, para medir o impacto dos principais fatores considerados em nosso estudo de simulação, ajustamos os resultados a um modelo de regressão linear múltipla, descrito na Subseção 4.1.3. A variável de resposta é a acurácia e os fatores são: o número de variáveis de

[1]Os demais cenários da simulação estão disponíveis no Apêndice A

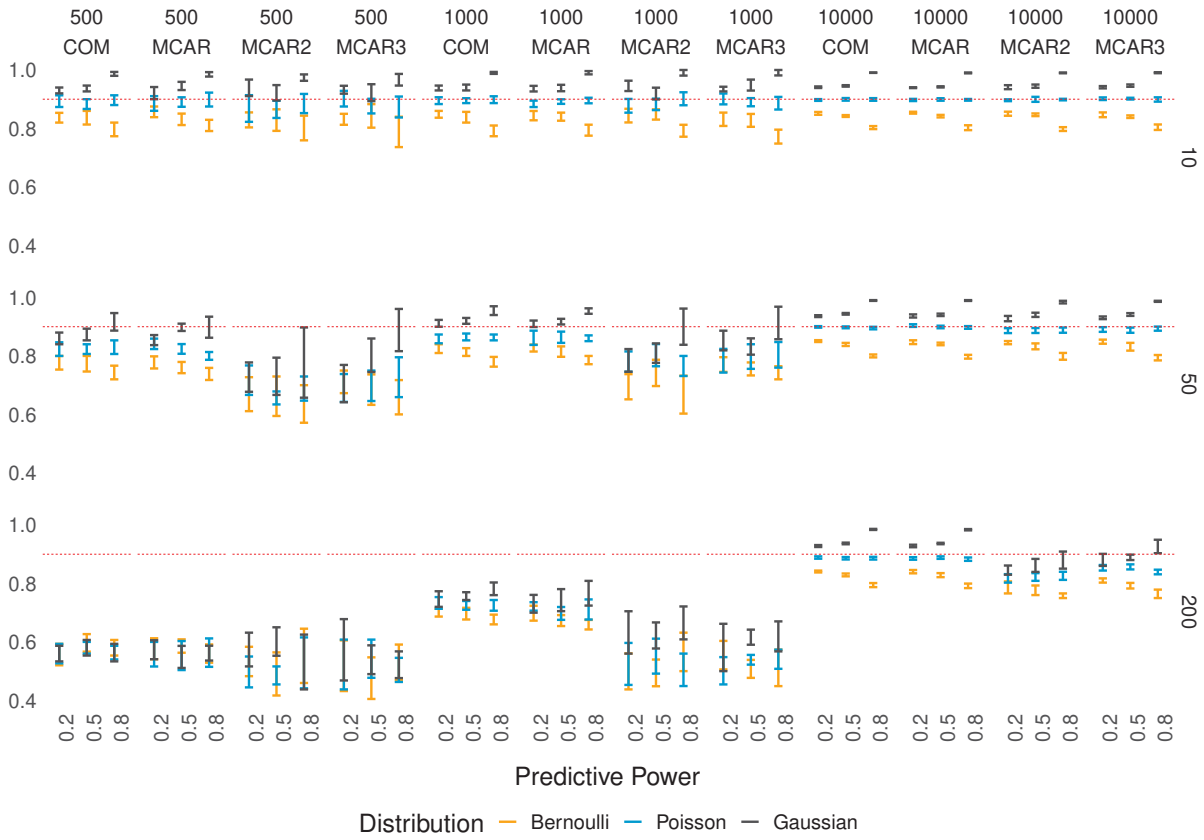


Figura 5.1: Intervalo interquartil da acurácia (CINPUT: 0.8; PMD: 30%; TMD MCAR2: 0.5).

entrada (NINPUT) com níveis (10, 50, e 200); tamanho da amostra (SS) com níveis 500, 1000, e 10000; correlação entre as variáveis de entrada (CINPUT) com níveis 0, 0.5, e 0.8; tipo das variáveis de entrada (TINPUT) com níveis Bernoulli, Gaussiana e Poisson; poder preditivo das variáveis de entrada (PPINPUT) com níveis 0.2, 0.5 e 0.8; tipo de dados incompletos (TMD) com níveis MCAR, MCAR2 e MCAR3 e proporção de dados incompletos (PMD) com níveis 0.7 e 0.9.

É importante enfatizar que ajustamos o modelo utilizando apenas os efeitos principais. O modelo poderia ser ajustado usando os efeitos interativos, porém o modelo apenas com os efeitos principais explica 67.91% da variabilidade da acurácia. Esse valor representa o coeficiente de determinação (R^2), utlizado como medida de qualidade de ajuste e o qual consideramos satisfatória apenas com o ajuste dos efeitos principais.

Tabela 5.1: Estimativas dos parâmetros e erros padrão, do modelo utilizando a acurácia como variável resposta.

Parâmetro	Estimativa	Parâmetro	Estimativa
Intercepto	79.61~(0.04)	TINPUT_Poisson	4.06~(0.03)
NINPUT_50	-4.52~(0.03)	TINPUT_Gauss	8.50~(0.03)
NINPUT_200	-19.53~(0.03)	PPINPUT_0.5	-0.07~(0.03)
SS_1000	6.31~(0.03)	PPINPUT_0.8	0.36~(0.03)
SS_10000	14.89~(0.03)	TMD_MCAR	-1.46~(0.04)
CINPUT_0.5	-0.08~(0.03)	TMD_MCAR2	-4.07~(0.04)
CINPUT_0.8	-0.23~(0.03)	TMD_MCAR3	-4.67~(0.04)
		PMD_0.9	2.27~(0.03)

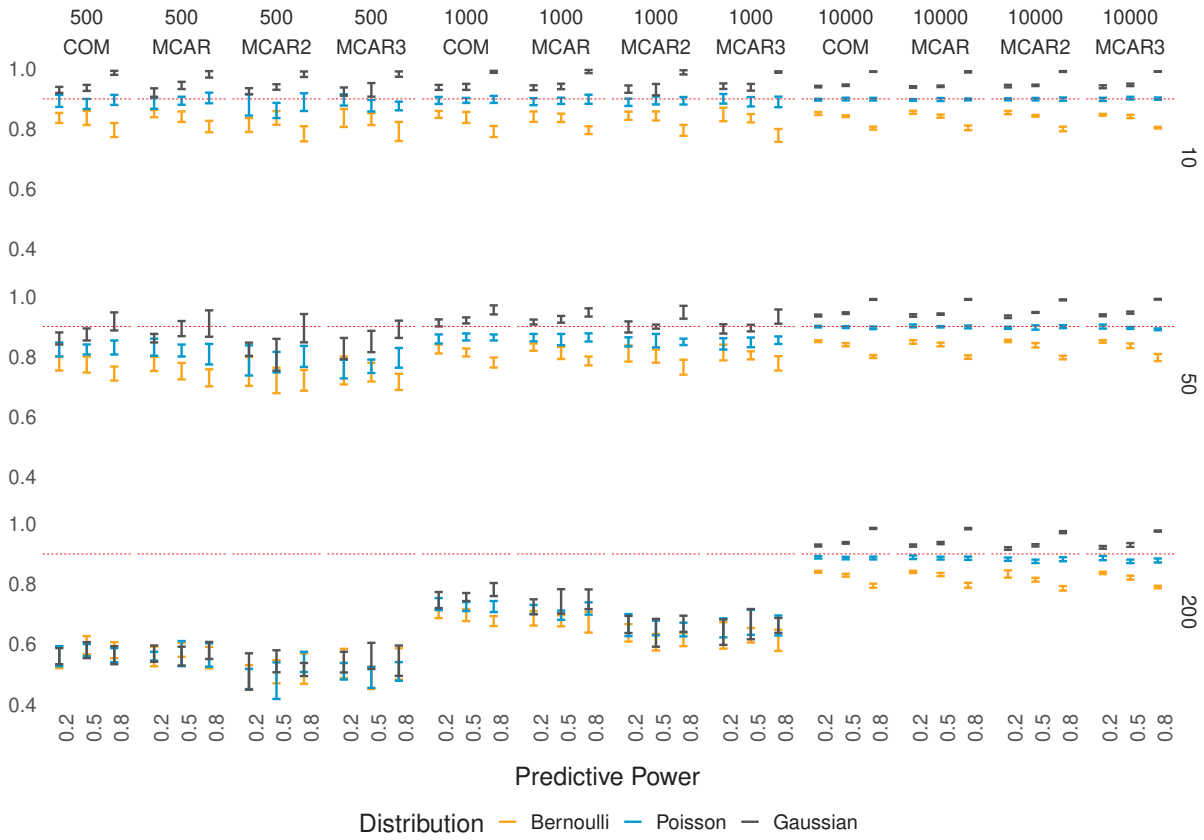


Figura 5.2: Intervalo interquartil da acurácia (CINPUT: 0.8; PMD: 10%; TMD MCAR2: 0.5).

Os resultados na Tabela 5.1 quantificam o efeito de cada fator sobre o valor esperado da acurácia. Assim, aumentando o número de variáveis de entrada de 10 para 50 e 200, esperamos uma diminuição média na acurácia de 4.52% e 19.53%, respectivamente. Da mesma forma, ao aumentar o tamanho da amostra de 500 para 1000 e 10000, esperamos um aumento médio na acurácia de 6.31% e 14.89%, respectivamente. A correlação entre as variáveis de entrada e a distribuição de seu poder preditivo apresenta menor impacto sobre a acurácia.

Entretanto, o tipo das variáveis de entrada é um fator importante para explicar a variabilidade da acurácia. A acurácia tende a aumentar em média 4.06% e 8.50% para inteiros e entradas contínuas em relação às entradas binárias. Finalmente, em relação ao tipo de mecanismo geradores de dados incompletos, nossos resultados mostram que do caso completo ao MCAR, MCAR2 e MCAR3, esperamos uma diminuição da acurácia de 1.46%, 4.07% e 4.67%, respectivamente.

Além da acurácia, também trazemos os resultados obtidos através da mensuração do F1 score. Optamos pela não representação gráfica dessa métrica, pois obtivemos resultados similares aos demonstrados pelos gráficos utilizando a acurácia. No entanto, também realizamos o ajuste do modelo de regressão linear múltipla para medir o impacto dos principais fatores considerados em nosso estudo de simulação, utilizando o F1 score. Os resultados são demonstrados pela Tabela 5.2 e quantificam o efeito de cada fator sobre o valor esperado da acurácia.

Da mesma forma que no modelo anterior utilizando a acurácia como variável resposta, ajustamos o modelo utilizando apenas os efeitos principais, que explica 65.52% da variabilidade do F1 score. Os resultados obtidos são bem próximos entre as duas métricas, a acurácia e o F1 Score, e a interpretação da Tabela 5.2, é realizada de forma análoga à Tabela 5.1.

Tabela 5.2: Estimativas dos parâmetros e erros padrão, do modelo utilizando o F1 score como variável resposta.

Parâmetro	Estimativa	Parâmetro	Estimativa
Intercepto	79.4 ~(0.05)	TINPUT_Poisson	4.5 ~(0.03)
NINPUT_50	-4.5~(0.03)	TINPUT_Gauss	8.6~(0.03)
NINPUT_200	-19.7~(0.03)	PPINPUT_0.5	-0.1 ~(0.03)
SS_1000	6.4~(0.03)	PPINPUT_0.8	0.3 ~(0.03)
SS_10000	15.1~(0.03)	TMD_MCAR	–1.5~(0.04)
CINPUT_0.5	-0.1 ~(0.03)	TMD_MCAR2	-4.2~(0.03)
CINPUT_0.8	-0.2~(0.03)	TMD_MCAR3	-4.8~(0.04)
		PMD_0.9	2.3~(0.03)

6 CONCLUSÃO

Neste trabalho, apresentamos um framework para avaliar o impacto dos dados incompletos em modelos preditivos. A estrutura foi baseada na geração estocástica de conjuntos de dados, em que controlamos fatores importantes como tamanho da amostra, quantidade, tipo e poder preditivo das variáveis de entrada, assim como a correlação entre elas. Consideramos também diferentes variações do mecanismo gerador de dados incompletos mais frequente: MCAR. Os resultados foram analisados usando ferramentas gráficas e um modelo de regressão linear múltipla que nos permite quantificar o impacto de cada fator sobre a acurácia esperada.

Os resultados mostraram que à medida que o tamanho da amostra aumenta, o impacto dos dados incompletos sobre a acurácia da predição, tende a desaparecer. Tal resultado é esperado e abre espaço para discutir a necessidade de tratar os dados incompletos neste cenário. Assim, uma contribuição adicional que emerge de nosso *framework* é uma ferramenta para quantificar quão grande deve ser nossa amostra, a fim de ignorar os dados incompletos com segurança.

Por outro lado, observamos que à medida que o número de variáveis de entrada aumenta, a acurácia tende a diminuir. Este ponto reforça que a seleção das variáveis de entrada é importante para aumentar o poder de previsão do modelo, principalmente em cenários com mais de 50 variáveis de entrada. A riqueza das variáveis de entrada é também um fator importante para explicar a acurácia do modelo. Portanto, recomendamos o uso de variáveis de entrada contínuas quando possível. Um resultado interessante é que a correlação entre as variáveis de entrada ou o mecanismo gerador de dados incompletos apresentou um baixo impacto sobre o desempenho preditivo. Este resultado sugere que o problema de multicolinearidade da estimativa dos parâmetros não é uma preocupação com a predição.

O conjunto de fatores controlados explicou a variabilidade da acurácia de $67,91\%$. Assim, argumentamos que nosso *framework* foi eficaz para avaliar o impacto dos dados incompletos e outros aspectos importantes do conjunto de dados no desempenho preditivo de um modelo de regressão logística. Por fim, sugerimos estender o *framework* para avaliar outros modelos de predição, tais como redes neurais, *random forest*, entre outros. Além disso, poderia ser adotado como um *framework* prático para avaliar a eficácia de novas abordagens para tratar os dados incompletos.

REFERÊNCIAS

[1] Ziawasch Abedjan, Xu Chu, Dong Deng, Raul Castro Fernandez, Ihab F Ilyas, Mourad Ouzzani, Paolo Papotti, Michael Stonebraker, and Nan Tang. Detecting data errors: Where are we and what needs to be done? *Proceedings of the VLDB Endowment*, 9(12):993–1004, 2016.

[2] Edgar Acuna and Caroline Rodriguez. The treatment of missing values and its effect on classifier accuracy. In *Classification, clustering, and data mining applications*, pages 639–647. Springer, 2004.

[3] Sergio Pío Alvarez, Adriana Marotta, and Libertad Tansini. Data density assessment using classification techniques. In *AMW*, 2017.

[4] Hélder Araújo and Joao Santos. Influence of data distribution in missing data imputation. In *Artificial Intelligence in Medicine: 16th Conference on Artificial Intelligence in Medicine, AIME 2017, Vienna, Austria, June 21-24, 2017, Proceedings*, volume 10259, page 285. Springer, 2017.

[5] Gustavo EAPA Batista and Maria Carolina Monard. An analysis of four missing data treatment methods for supervised learning. *Applied artificial intelligence*, 17(5-6):519–533, 2003.

[6] Laure Berti-Equille, Hazar Harmouch, Felix Naumann, Noël Novelli, and Saravanan Thirumuruganathan. Discovery of genuine functional dependencies from relational data with missing values. *Proceedings of the VLDB Endowment*, 11(8):880–892, 2018.

[7] Luciano C Blomberg and Duncan Dubugras A Ruiz. Evaluating the influence of missing data on classification algorithms in data mining applications. *SBSI 2013: Simpósio Brasileiro de Sistemas de Informação*, pages 734–743, 2013.

[8] Marvin L Brown and John F Kros. Data mining and the impact of missing data. *Industrial Management & Data Systems*, 103(8):611–621, 2003.

[9] Min Chen, Shiwen Mao, and Yunhao Liu. Big data: A survey. *Mobile networks and applications*, 19(2):171–209, 2014.

[10] Fei Chiang and Renée J Miller. Discovering data quality rules. *Proceedings of the VLDB Endowment*, 1(1):1166–1177, 2008.

[11] Richard D De Veaux, David J Hand, et al. How to lie with bad data. *Statistical Science*, 20 (3):231–238, 2005.

[12] Dong Deng, Raul Castro Fernandez, Ziawasch Abedjan, Sibo Wang, Michael Stonebraker, Ahmed K Elmagarmid, Ihab F Ilyas, Samuel Madden, Mourad Ouzzani, and Nan Tang. The data civilizer system. In *CIDR*, 2017.

[13] Benjamin Durakovic. Design of experiments application, concepts, examples: State of the art. *Periodicals of Engineering and Natural Sciences*, 5(3):421–439, dec 2017. ISSN 23034521. doi: 10.21533/pen.v5i3.145.

[14] Imane Ezzine and Laila Benhlima. A study of handling missing data methods for big data. In *2018 IEEE 5th International Congress on Information Science and Technology (CiSt)*, pages 498–501. IEEE, 2018.

[15] Wenfei Fan. Data quality: From theory to practice. *Acm Sigmod Record*, 44(3):7–18, 2015.

[16] By John Gantz and David Reinsel. Extracting Value from Chaos. Technical Report June, 2011. URL `http://idcdocserv.com/1142`.

[17] John Gantz and David Reinsel. The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. *IDC iView: IDC Analyze the future*, 2007 (2012):1–16, 2012.

[18] Unai Garciarena and Roberto Santana. An extensive analysis of the interaction between missing data types, imputation methods, and supervised classifiers. *Expert Systems with Applications*, 89:52–65, 2017.

[19] Demand Gen. Assessing the impact of dirty data on sales marketing performance. `https://www.zoominfo.com/business/mktg/ebooks/dirtydataebok.pdf`, 2017.

[20] M Kashif Gill, Tirusew Asefa, Yasir Kaheil, and Mac McKee. Effect of missing data on performance of learning algorithms for hydrologic predictions: Implications to an imputation technique. *Water resources research*, 43(7), 2007.

[21] Sergio Greco, Cristian Molinaro, and Irina Trubitsyna. Computing approximate certain answers over incomplete databases. In *AMW*, 2017.

[22] Deepak Gupta and Rinkle Rani. A study of big data evolution and research challenges. *Journal of Information Science*, 45(3), 2019. ISSN 17416485. doi: 10.1177/0165551518789880.

[23] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.

[24] Anders Haug, Frederik Zachariassen, and Dennis Van Liempd. The costs of poor data quality. *Journal of Industrial Engineering and Management (JIEM)*, 4(2):168–193, 2011.

[25] Lorin Hitt. Strength in numbers: How does data-driven decisionmaking affect firm performance? 2011.

[26] José M Jerez, Ignacio Molina, Pedro J García-Laencina, Emilio Alba, Nuria Ribelles, Miguel Martín, and Leonardo Franco. Missing data imputation using statistical and machine learning methods in a real breast cancer problem. *Artificial intelligence in medicine*, 50(2): 105–115, 2010.

[27] S. Kandel, A. Paepcke, J. M. Hellerstein, and J. Heer. Enterprise data analysis and visualization: An interview study. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2917–2926, Dec 2012. ISSN 1077-2626. doi: 10.1109/TVCG.2012.219.

[28] Zuhair Khayyat, Ihab F Ilyas, Alekh Jindal, Samuel Madden, Mourad Ouzzani, Paolo Papotti, Jorge-Arnulfo Quiané-Ruiz, Nan Tang, and Si Yin. Bigdansing: A system for big data cleansing. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1215–1230, 2015.

[29] Won Kim, Byoung-Ju Choi, Eui-Kyeong Hong, Soo-Kyung Kim, and Doheon Lee. A taxonomy of dirty data. *Data Mining and Knowledge Discovery*, 7(1):81–99, Jan 2003. ISSN 1573-756X. doi: 10.1023/A:1021564703268. URL https://doi.org/10.1023/A:1021564703268.

[30] Steve LaValle, Eric Lesser, Rebecca Shockley, Michael S Hopkins, and Nina Kruschwitz. Big data, analytics and the path from insights to value. *MIT sloan management review*, 52 (2):21, 2011.

[31] Wei-Chao Lin and Chih-Fong Tsai. Missing value imputation: a review and analysis of the literature (2006–2017). *Artificial Intelligence Review*, pages 1–23, 2019.

[32] Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data*, volume 333. John Wiley & Sons, 1987.

[33] Peng Liu, Lei Lei, and Naijun Wu. A quantitative study of the effect of missing data in classifiers. In *The Fifth International Conference on Computer and Information Technology (CIT'05)*, pages 28–33. IEEE, 2005.

[34] Julián Luengo, Salvador García, and Francisco Herrera. On the choice of the best imputation methods for missing values considering three groups of classification methods. *Knowledge and information systems*, 32(1):77–108, 2012.

[35] Benjamin M. Marlin. Missing data problems in machine learning. 2008.

[36] Heiko Müller and Johann-Christph Freytag. *Problems, methods, and challenges in comprehensive data cleansing*. Professoren des Inst. Für Informatik, 2005.

[37] David F. Nettleton, Albert Orriols-Puig, and Albert Fornells. A study of the effect of different types of noise on the precision of supervised learning techniques. *Artificial Intelligence Review*, 33(4):275–306, Apr 2010. ISSN 1573-7462. doi: 10.1007/s10462-010-9156-z. URL https://doi.org/10.1007/s10462-010-9156-z.

[38] Bruno M Nogueira, Tadeu RA Santos, and Luis E Zárate. Comparison of classifiers efficiency on missing values recovering: application in a marketing database with massive missing data. In *2007 IEEE Symposium on Computational Intelligence and Data Mining*, pages 66–72. IEEE, 2007.

[39] Tomasz Orczyk and Piotr Porwik. Influence of missing data imputation method on the classification accuracy of the medical data. *Journal of Medical Informatics & Technologies*, 22, 2013.

[40] Eduardo HM Pena. Workload-aware discovery of integrity constraints for data cleaning. *Proceedings of the VLDB Endowment*, 11(8), 2018.

[41] Jason Poulos and Rafael Valle. Missing data imputation for supervised learning. *Applied Artificial Intelligence*, 32(2):186–196, 2018.

[42] Foster Provost and Tom Fawcett. Data science and its relationship to big data and data-driven decision making. *Big Data*, 1(1):51–59, 2013. doi: 10.1089/big.2013.1508. URL https://doi.org/10.1089/big.2013.1508. PMID: 27447038.

[43] Dorian Pyle. *Data preparation for data mining*. morgan kaufmann, 1999.

[44] Michael B Richman, Theodore B Trafalis, and Indra Adrianto. Missing data imputation through machine learning algorithms. In *Artificial Intelligence Methods in the Environmental Sciences*, pages 153–169. Springer, 2009.

[45] Maytal Saar-Tsechansky and Foster Provost. Handling missing values when applying classification models. *Journal of machine learning research*, 8(Jul):1623–1657, 2007.

[46] Claude Sammut and Geoffrey I. Webb. *Encyclopedia of Machine Learning and Data Mining*. Springer Publishing Company, Incorporated, 2nd edition, 2017. ISBN 148997685X, 9781489976857.

[47] Miriam Seoane Santos, Ricardo Cardoso Pereira, Adriana Fonseca Costa, Jastin Pompeu Soares, João Santos, and Pedro Henriques Abreu. Generating synthetic missing data: A review by missing mechanism. *IEEE Access*, 2019.

[48] Jadran Sessa and Dabeeruddin Syed. Techniques to deal with missing data. In *2016 5th international conference on electronic devices, systems and applications (ICEDSA)*, pages 1–4. IEEE, 2016.

[49] Rajesh Sharma, Matteo Magnani, and Danilo Montesi. Investigating the types and effects of missing data in multilayer networks. In *Advances in Social Networks Analysis and Mining (ASONAM), 2015 IEEE/ACM International Conference on*, pages 392–399. IEEE, 2015.

[50] Qinbao Song, Martin Shepperd, Xiangru Chen, and Jun Liu. Can k-nn imputation improve the performance of c4. 5 with small software project data sets? a comparative evaluation. *Journal of Systems and software*, 81(12):2361–2370, 2008.

[51] T Stack. Internet of things (iot) data continues to explode exponentially. who is using that data and how? *Cisco Blogs*, 5, 2018.

[52] K. Strike, K. El Emam, and N. Madhavji. Software cost estimation with incomplete data. *IEEE Transactions on Software Engineering*, 27(10):890–908, 2001. ISSN 00985589. doi: 10.1109/32.962560. URL `http://ieeexplore.ieee.org/document/962560/`.

[53] Anestis Touloumis. Simulating correlated binary and multinomial responses under marginal model specification: The simcormultres package. *The R Journal*, 8(2):79–91, 2016.

[54] Bhekisipho Twala. An empirical comparison of techniques for handling incomplete data using decision trees. *Applied Artificial Intelligence*, 23(5):373–405, 2009.

[55] Mohamed Yakout, Laure Berti-Équille, and Ahmed K Elmagarmid. Don't be scared: use scalable automatic repairing with maximal likelihood and bounded changes. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pages 553–564. ACM, 2013.

[56] H. Yin and H. Dong. The problem of noise in classification: Past, current and future work. In *2011 IEEE 3rd International Conference on Communication Software and Networks*, pages 412–416, May 2011. doi: 10.1109/ICCSN.2011.6014597.

[57] Shichao Zhang, Xindong Wu, and Manlong Zhu. Efficient missing data imputation for supervised learning. In *Cognitive Informatics (ICCI), 2010 9th IEEE International Conference on*, pages 672–679. IEEE, 2010.

[58] Xingquan Zhu and Xindong Wu. Class noise vs. attribute noise: A quantitative study. *Artificial intelligence review*, 22(3):177–210, 2004.

[59] XINGQUAN ZHU, XINDONG WU, and QIJUN CHEN. Bridging local and global data cleansing: Identifying class noise in large, distributed data datasets. *Data Mining and Knowledge Discovery*, 12(2):275–308, May 2006. ISSN 1573-756X. doi: 10.1007/s10618-005-0012-8. URL https://doi.org/10.1007/s10618-005-0012-8.

**

APÊNDICE A – RESULTADOS ADICIONAIS

Os gráficos abaixo representam o intervalo quantil da acurácia por tamanho amostral e mecanismos geradores de dados incompletos (Eixo x acima), CINPUT and NINPUT (Eixo y da esquerda para a direita), e poder preditivo das variáveis de entrada (Eixo x abaixo).

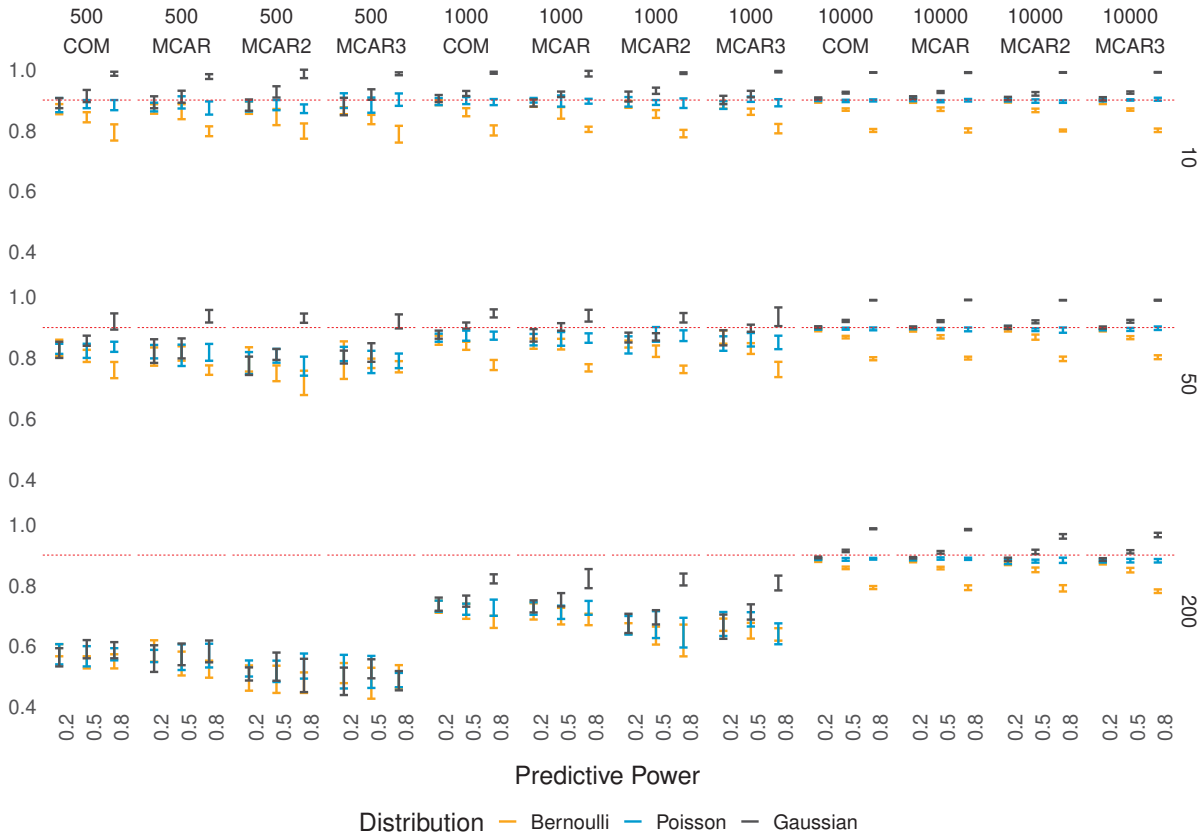


Figura A.1: Intervalo interquartil da acurácia (CINPUT: 0; PMD: 10%; TMD MAR: 0.5).

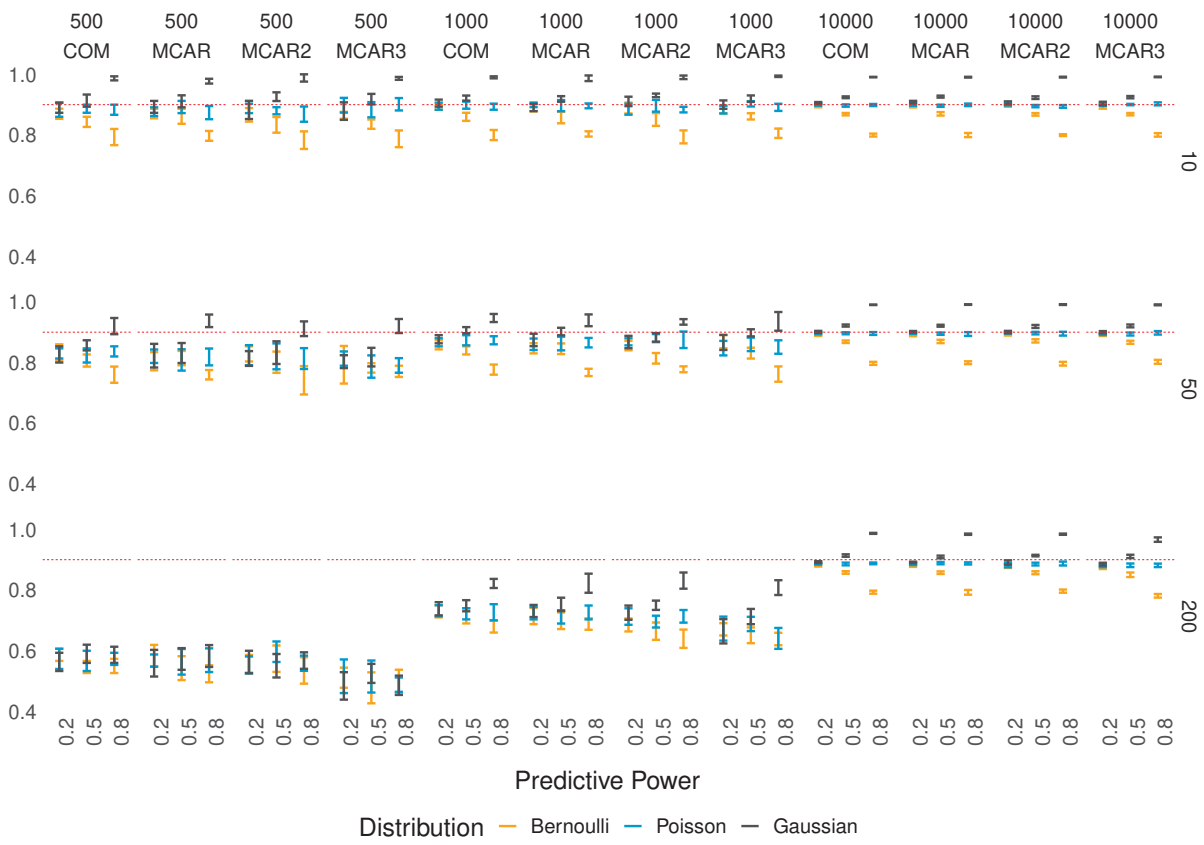


Figura A.2: Intervalo interquartil da acurácia (CINPUT: 0; PMD: 10%; TMD MAR: 0.8).

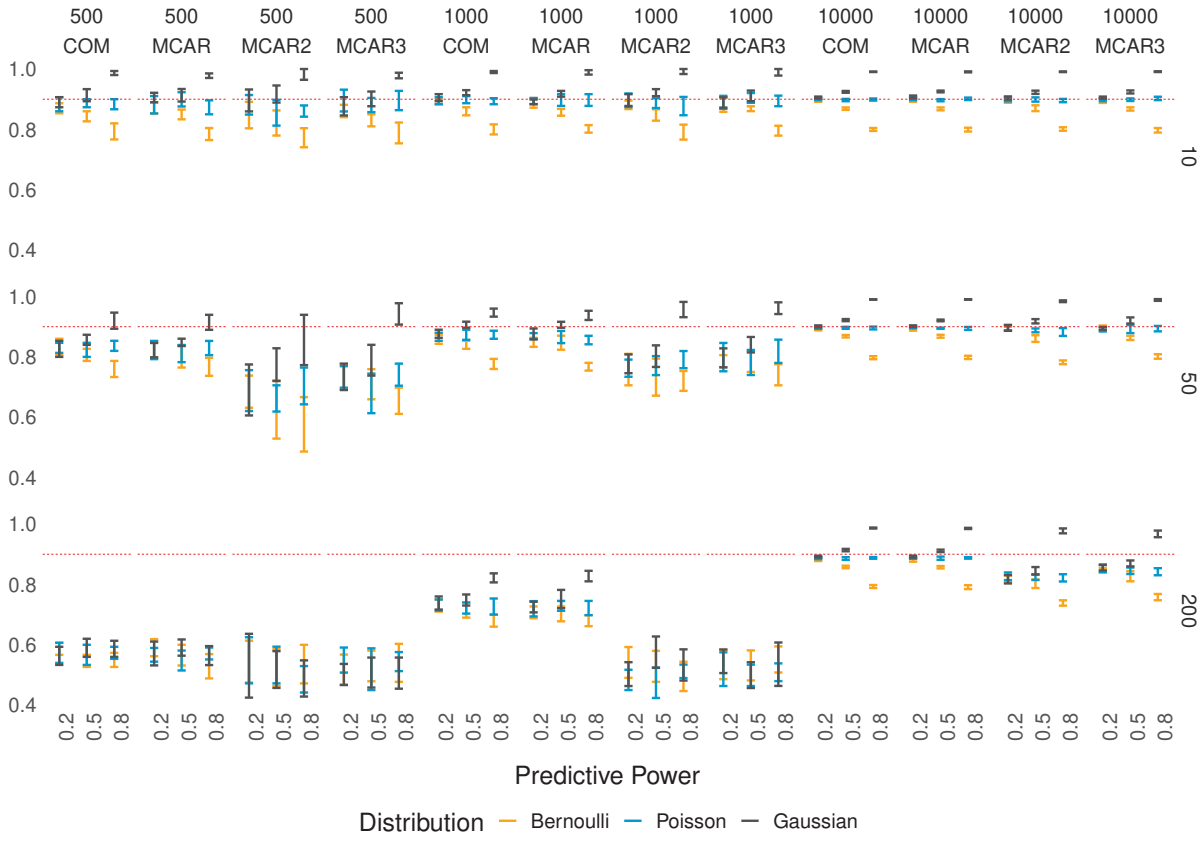


Figura A.3: Intervalo interquartil da acurácia (CINPUT: 0; PMD: 30%; TMD MAR: 0.5).

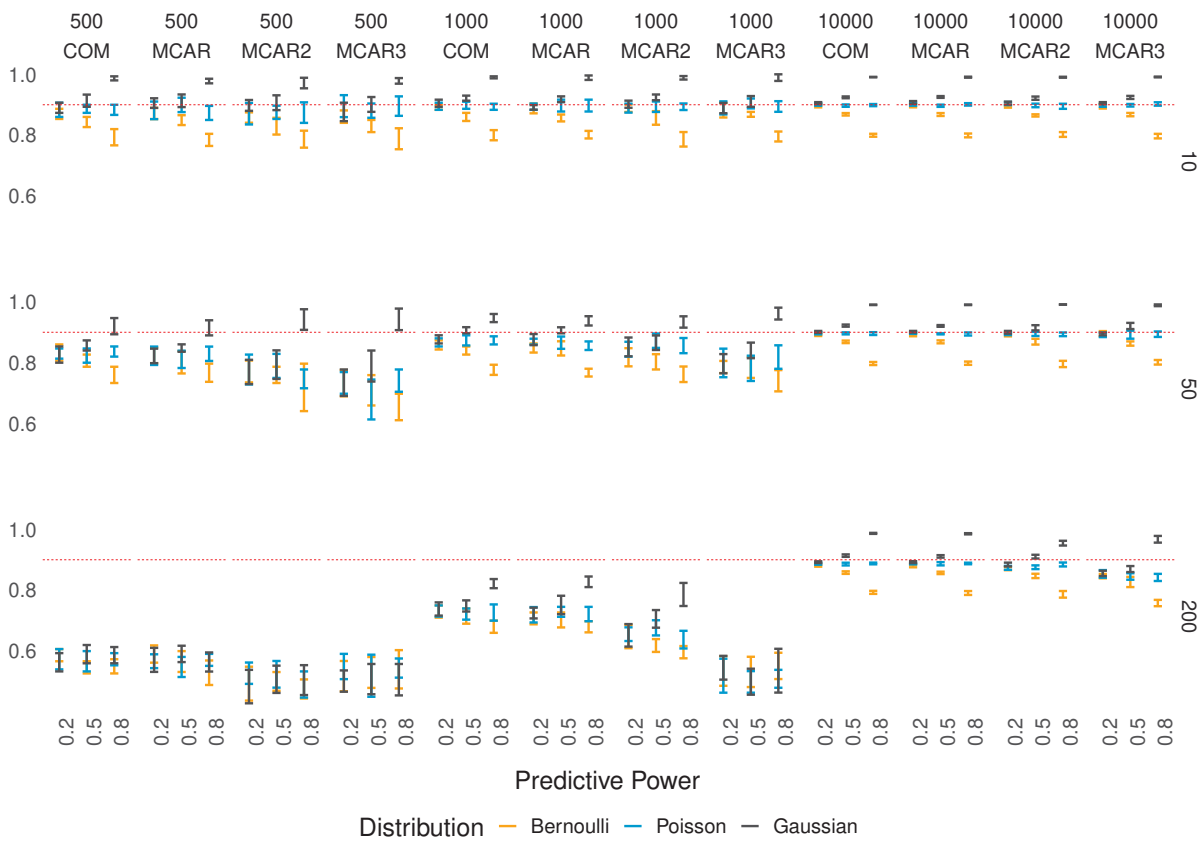


Figura A.4: Intervalo interquartil da acurácia (CINPUT: 0; PMD: 30%; TMD MAR: 0.8).

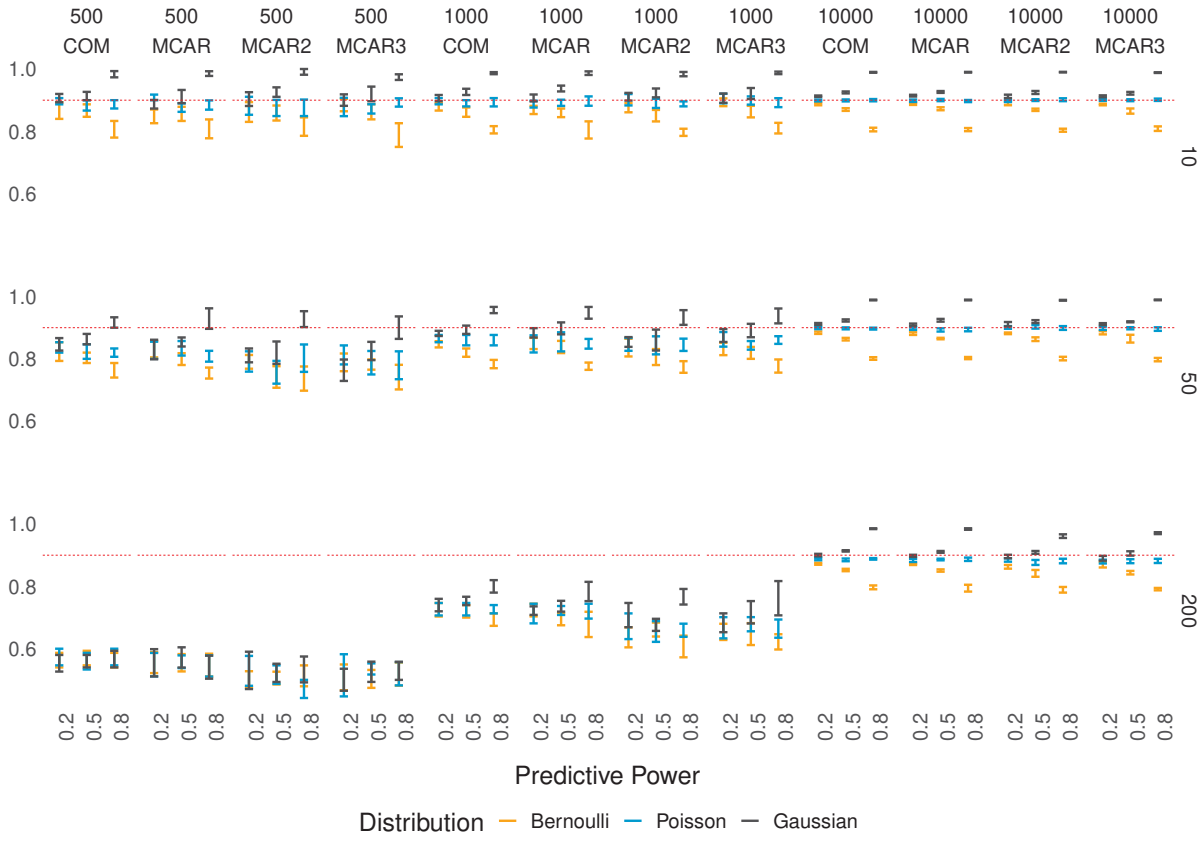


Figura A.5: Intervalo interquartil da acurácia (CINPUT: 0.5; PMD: 10%; TMD MAR: 0.5).

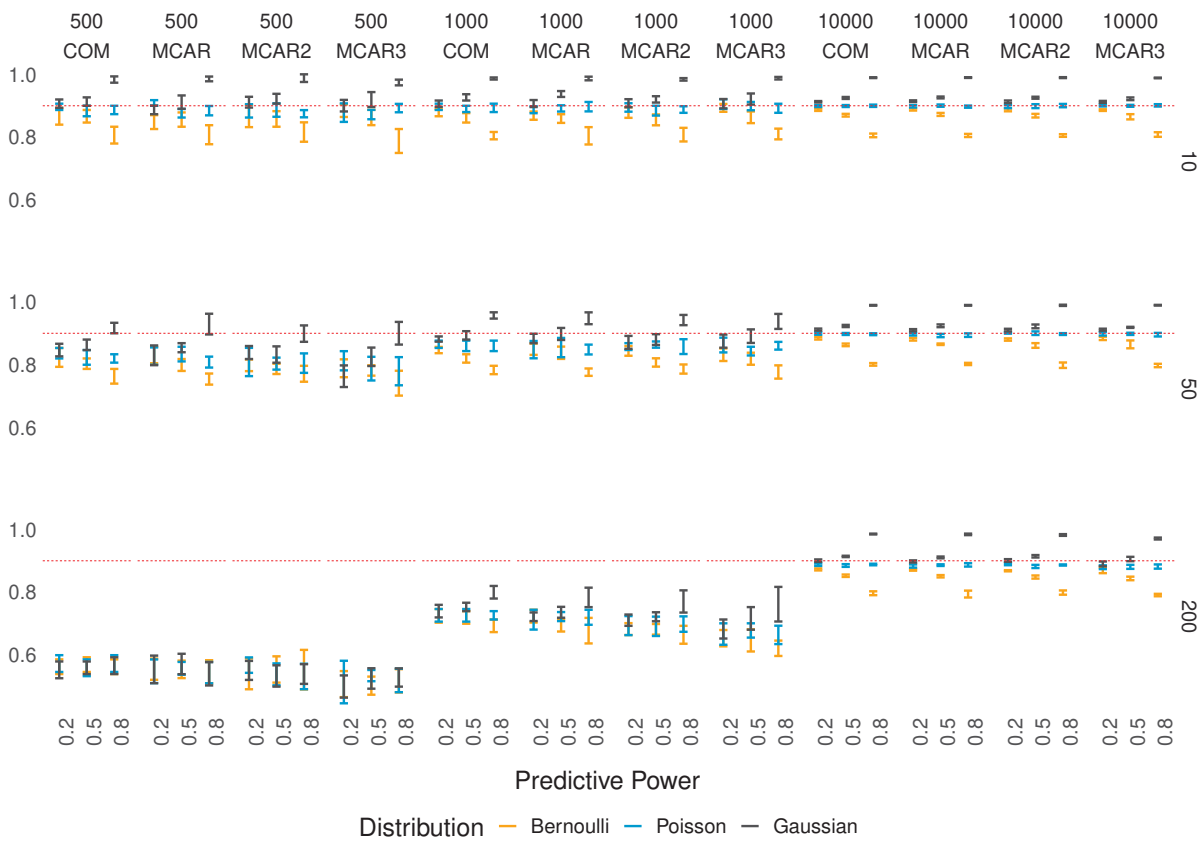


Figura A.6: Intervalo interquartil da acurácia (CINPUT: 0.5; PMD: 10%; TMD MAR: 0.8).

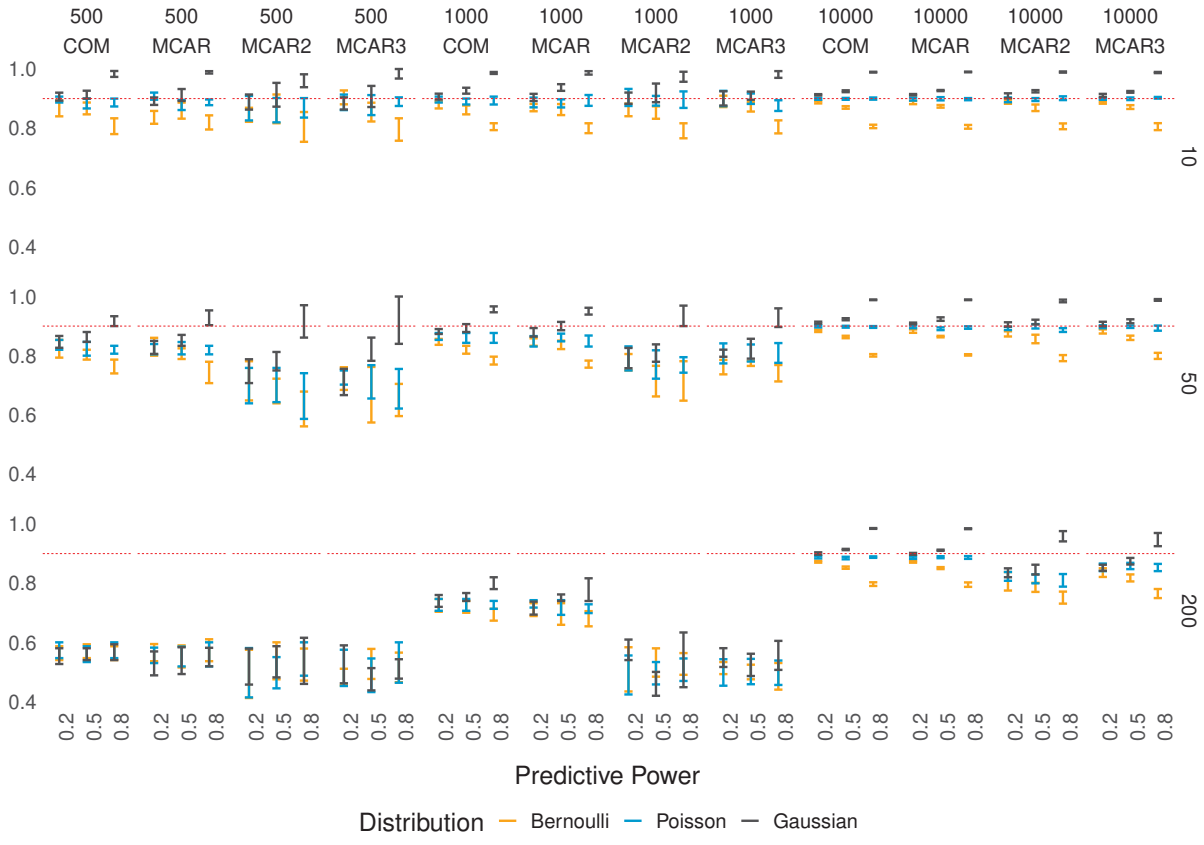


Figura A.7: Intervalo interquartil da acurácia (CINPUT: 0.5; PMD: 30%; TMD MAR: 0.5).

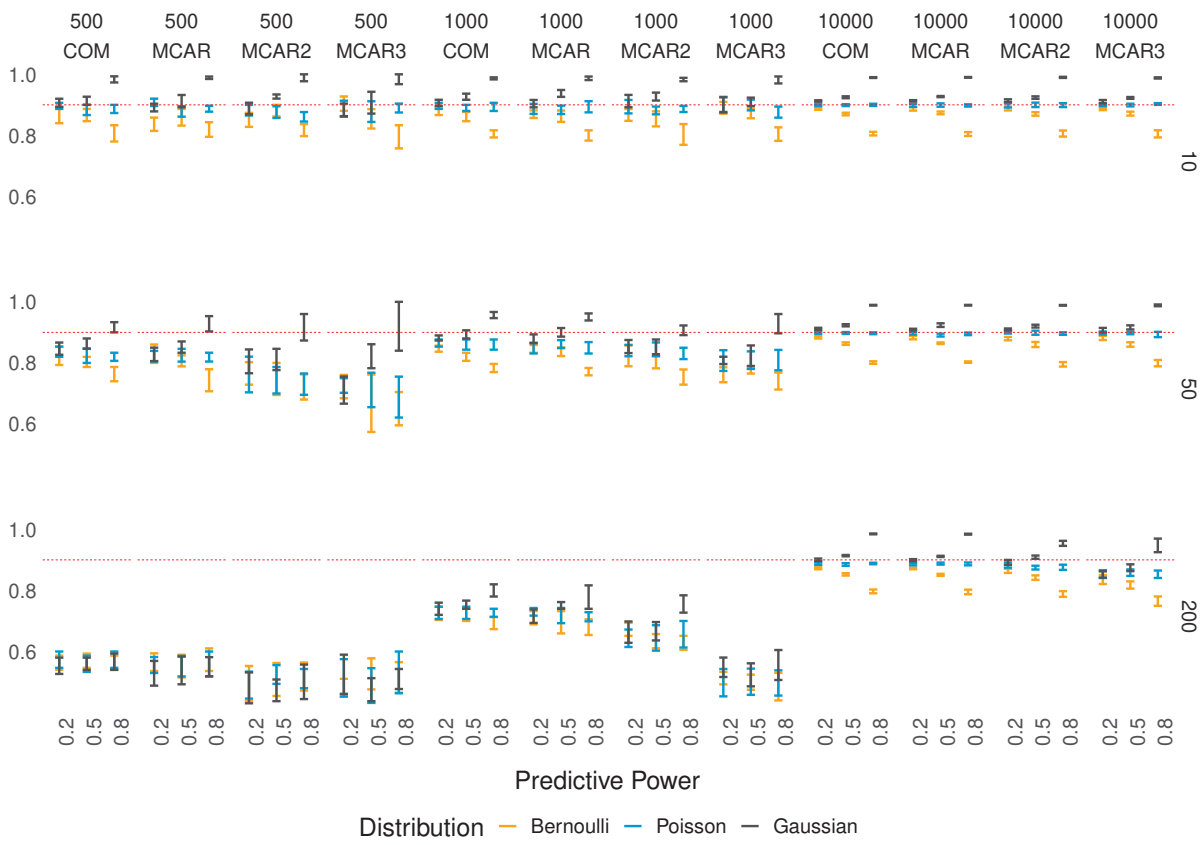


Figura A.8: Intervalo interquartil da acurácia (CINPUT: 0.5; PMD: 30%; TMD MAR: 0.8).

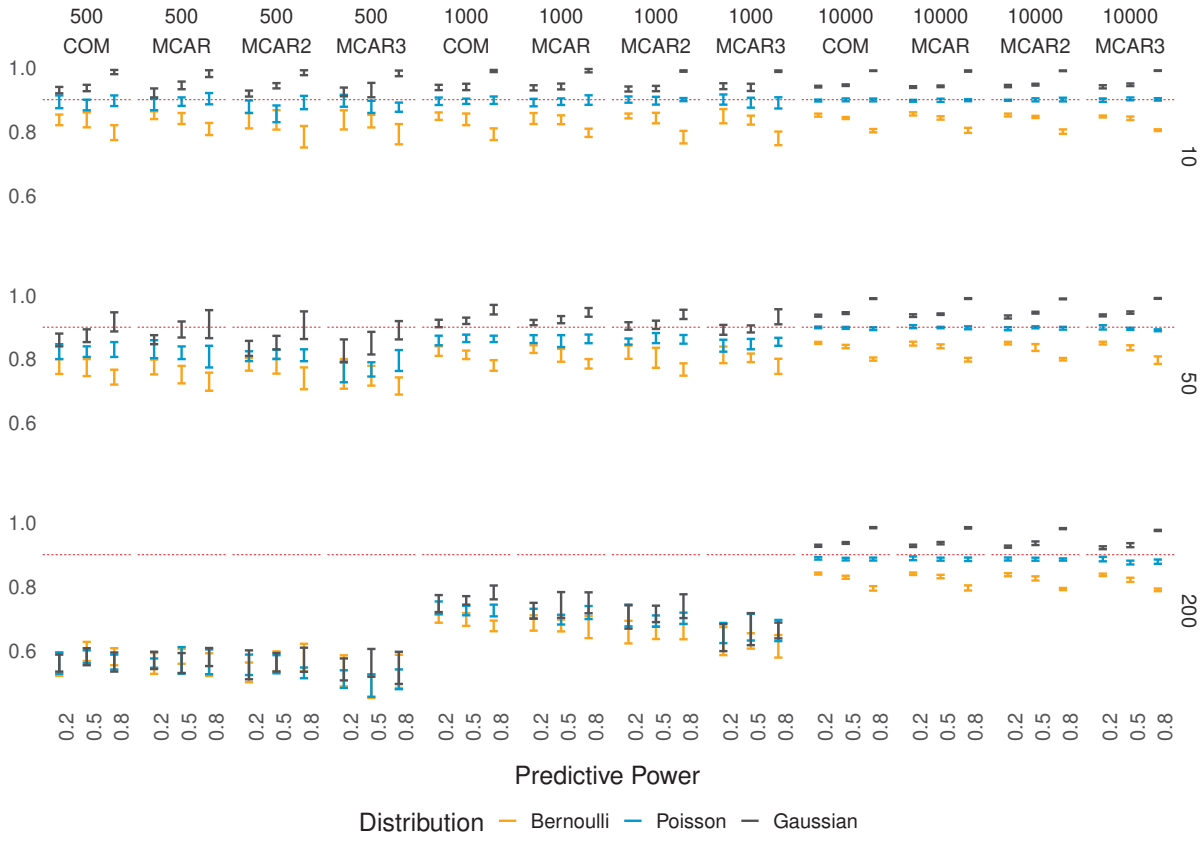


Figura A.9: Intervalo interquartil da acurácia (CINPUT: 0.8; PMD: 10%; TMD MAR: 0.8).

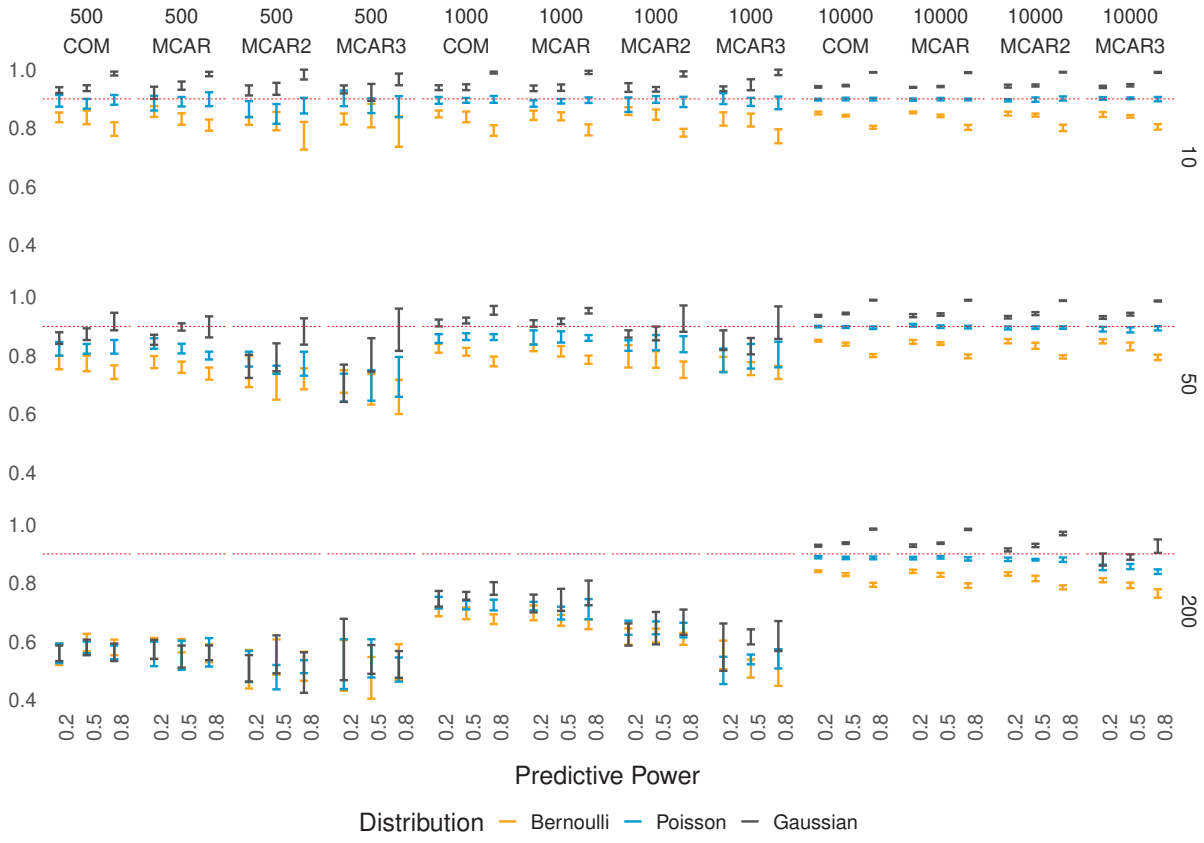


Figura A.10: Intervalo interquartil da acurácia (CINPUT: 0.8; PMD: 30%; TMD MAR: 0.8).

APÊNDICE B – CÓDIGO CALIBRAÇÃO

```
#-------------------------------------------------------------
# Packages
library(SimCorMultRes)
library(tidyverse)
library(furrr)

#-------------------------------------------------------------
# beta_calc determines the value of the betas vector parameters

# j:input variables
# ef_size: represents the effect of the input variables on
#the response variable (How much they explain)
# imp: predictive power of the input variables
#(How do they explain)

# A density function of the geometric distribution was used
# to determine the value of the betas
# For the geometric function is passed from the parameters:
#k and imp (prob of function)

# The effect control the accuracy
# The predictive power (imp) controls how input variables
#explain the answer

# The function returns a numerical vector with k elements
# which are the values
# parametric of the effects of input variables.
# The sum of the elements of the
# vector equals ef_size

beta_calc <- function(j = 10, ef_size = 0, imp = 0.5) {
  beta <- dgeom(0:(j - 1), prob = imp)
  beta <- ef_size * beta/sum(beta)
  return(beta)
}


#-------------------------------------------------------------
# Returns the matrix of input variables
# n: sample size.
# j: input variables.
# dist: distribution of input variables.
# rho: correlation between input variables in
# the Toeplitz structure.
```

```
# The function returns the matrix of input variables with
# n rows and j columns.

generate_predictors <- function(n = 100, j = 10,
                                dist = "normal", rho = 0) {
  correl <- toeplitz(x = c(1, rep(rho, j - 1)))
  X <- switch(
    EXPR = dist,
    "normal" = {
      qpar <- replicate(j,
                        list(mean = 0, sd = 1),
                        simplify = FALSE)
      rnorta(n,
             cor.matrix = correl,
             distr = rep("qnorm", j),
             qparameters = qpar)x
    },
    "binomial" = {
      qpar <- replicate(j,
                        list(size = 1, prob = 0.5),
                        simplify = FALSE)
      (rnorta(n,
              cor.matrix = correl,
              distr = rep("qbinom", j),
              qparameters = qpar) - 0.5)/0.5
    },
    "poisson" = {
      # (matrix(rpois(n * k, lambda = 10), ncol = k) - 10)/3.162278
      qpar <- replicate(j,
                        list(lambda = 10),
                        simplify = FALSE)
      (rnorta(n,
              cor.matrix = correl,
              distr = rep("qpois", j),
              qparameters = qpar) - 10)/3.162278
    })
  return(X)
}


#-------------------------------------------------------------

# Determines response variable, applies model and returns accuracy

# beta: effect vector of the input varible determined
# by calling the function beta_calc()
# X: matrix representing the input variable used to
```

```
# simulate the response variable

# The function returns the accuracy with simulated data
# according to the specifications.

simul_acc <- function(X = generate_predictors(n = 10,
                                              j = 3,
                                              dist = "normal",
                                              rho = 0),
                      beta = beta_calc(j = 3)) {
  eta <- X %*% beta
  y <- rbinom(nrow(X), size = 1, prob = binomial()$linkinv(eta))
  m0 <- glm.fit(x = cbind(1, X), y = y, family = binomial())
  acc <- sum((m0$fitted.values > 0.5) == y)/nrow(X)
  return(acc)
}


#-----------------------------------------------------------------
# Function that resolves the effect size for a fixed
# input data matrix in all steps.
# Get the effect size value given simulation conditions.

# acc: fixed accuracy to determine the ef_size
# n: sample size (rows)
# j: input variables
# imp: predictive power of the input variables
# dist: distributions of input variables
# rho: correlation between the input variables
# n_repl: Number of independent replications


solve_ef_size <- function(acc = 0.9,
                          n = 1000,
                          j = 10,
                          imp = 0.5,
                          dist = "normal",
                          rho = 0,
                          n_repl = 10,
                          verbose = TRUE) {

  f_obj <- function(ef_size, acc, j, imp, X) {
    beta_pars <- beta_calc(j = j,
                           ef_size = ef_size,
                           imp = imp)

    simul_acc <- replicate(n = n_repl,
                           simul_acc(X = X, beta = beta_pars))
```

```
    cat(".")
    acc - mean(simul_acc)
  }

  X <- generate_predictors(n = n,
                           j = j,
                           dist = dist,
                           rho = rho)

  root <- try(uniroot(f_obj,
                      interval = c(0, 30),
                      acc = acc,
                      j = j,
                      imp = imp,
                      X = X))

  acc_opt <- replicate(n = n_repl,
                       simul_acc(X = X,
                                 beta = beta_calc(j = j,
                                                  ef_size = root$root,
                                                  imp = imp)))
  acc_opt <- mean(acc_opt)
  if (verbose) {
    cat("\n")
    fmt <- paste("target acc: %0.3f",
                 "k: %d",
                 "imp: %0.3f",
                 "dist: %s",
                 "rho: %0.3f",
                 "EF_SIZE: %0.3f",
                 "MEAN ACC: %0.3f.\n",
                 sep = "; ")
    cat(sprintf(fmt = fmt,
                acc, j, imp, dist, rho, root$root, acc_opt))
  }
  if (inherits(root, "try-error")) {

    cat(sprintf("target acc: %0.3f; j: %d; imp: %0.3f;
            dist: %s; rho: %0.3f\n", acc, j, imp, dist, rho))
    return(data.frame(root = NA, f.root = NA, iter = NA,
                      init.it = NA, estim.prec = NA, acc = NA))
  } else {

    cbind(as.data.frame(root), acc_opt = acc_opt)
  }
}
```

```
#-----------------------------------------------------------------
# Function that solves the effect size for various input
# data matrices.
# Get the effect size value given simulation conditions.
# acc: fixed accuracy to determine the ef_size
# n: sample size (row)
# j:  input variables
# imp: predictive power of the input variables
# dist: distribution of input variables
# rho: correlation between input variables
# n_repl: Number of independent replications

solve_ef_size_final <- function(acc = 0.9,
                                n = 1000,
                                j = 10,
                                imp = 0.5,
                                dist = "normal",
                                rho = 0,
                                n_repl = 10,
                                verbose = TRUE) {

  res <- replicate(n = 15,
                   simplify = FALSE,
                   solve_ef_size(acc = acc,
                                 n = n,
                                 j = j,
                                 imp = imp,
                                 dist = dist,
                                 rho = rho,
                                 n_repl = n_repl,
                                 verbose = TRUE))

  tb <- do.call(rbind, res)

  # Estimativa de `ef_size` pela média.
  root_m <- round(mean(tb$root), 2)
  my_tb <- cbind(acc = acc, n = n, j = j,
                   imp = imp, dist = dist, rho = rho,
                   root_m = as.data.frame(root_m))

  write.table(my_tb, file = "teste.txt", append = TRUE,
                sep = "\t",col.names = FALSE)
  return(my_tb)
}

#-----------------------------------------------------------------
# All combinations of scenario
```

```
grid_ef_size <- as.data.frame(crossing(j = c(10, 50, 200),
                          n = c(500, 1000, 10000),
                          acc = c(0.9),
                          imp = c(0.2, 0.5, 0.8),
                          rho = c(0, 0.5, 0.8),
                          dist = c("normal", "binomial", "poisson")))

#-----------------------------------------------------------------
#Parallel cod
require(doParallel)
ncl <- detectCores() # Checa quantos núcleos existem na máquina
cl <- makeCluster(ncl)
registerDoParallel(cl) # Registra os clusters a serem utilizados

system.time({
  foreach(w=1:nrow(grid_ef_size),
        .packages=c("SimCorMultRes","furrr", "tidyverse"))
        %dopar% solve_ef_size_final(acc = grid_ef_size[w,3],
                                    n = grid_ef_size[w,2],
                                    j = grid_ef_size[w,1],
                                    imp = grid_ef_size[w,4],
                                    dist = grid_ef_size[w,6],
                                    rho = grid_ef_size[w,5])
})
stopCluster(cl)
```

APÊNDICE C – CÓDIGO GERAL

```
#-----------------------------------------------------------------
# results effect code
experimentos <- my_tb

#-----------------------------------------------------------------
#Packages
require(SimCorMultRes) #rnorta
require(caret) #create partition
require(bindata)
require(ggplot2)
library(tidyverse)

#-----------------------------------------------------------------
# beta_calc determines the value of the betas vector parameters

# j:input variables
# ef_size: represents the effect of the input variables
# on the response variable (How much they explain)
# imp: predictive power of the input variables
# (How do they explain)

# A density function of the geometric distribution was
# used to determine the value of the betas
# For the geometric function is passed from the parameters:
# k and imp (prob of function)

# The effect control the accuracy
# The predictive power (imp) controls how input variables
# explain the answer

# The function returns a numerical vector with
# k elements which are the values
# parametric of the effects of input variables.
# The sum of the elements of the
# vector equals ef_size

beta_calc <- function(j = 10, ef_size = 0, imp = 0.5) {
  beta <- dgeom(0:(j - 1), prob = imp)
  beta <- ef_size * beta/sum(beta)
  return(beta)
}

#-----------------------------------------------------------------
# Returns the matrix of input variables
```

```
# n: sample size.
# j: input variables.
# dist: distribution of input variables.
# rho: correlation between input variables in the Toeplitz structure.

# The function returns the matrix of input variables with n rows and j

generate_predictors <- function(n = 100, j = 10,
                                dist = "normal", rho = 0) {
  correl <- toeplitz(x = c(1, rep(rho, j - 1)))
  X <- switch(
    EXPR = dist,
    "normal" = {
      qpar <- replicate(j,
                        list(mean = 0, sd = 1),
                        simplify = FALSE)
      rnorta(n,
             cor.matrix = correl,
             distr = rep("qnorm", j),
             qparameters = qpar)
    },
    "binomial" = {
      qpar <- replicate(j,
                        list(size = 1, prob = 0.5),
                        simplify = FALSE)
      (rnorta(n,
              cor.matrix = correl,
              distr = rep("qbinom", j),
              qparameters = qpar) - 0.5)/0.5
    },
    "poisson" = {
      # (matrix(rpois(n * k, lambda = 10), ncol = k) - 10)/3.162278
      qpar <- replicate(j,
                        list(lambda = 10),
                        simplify = FALSE)
      (rnorta(n,
              cor.matrix = correl,
              distr = rep("qpois", j),
              qparameters = qpar) - 10)/3.162278
    })
  return(X)
}

#-------------------------------------------------------------

# The function simul_model(train, test) fit the glm
# to the training data
```

```
# and performs the prediction on the test data
# returning the accuracy amd some others measurement
# of the model

# Metrics : Accuracy, Specificity, Precision, Recall, F1

simul_model <- function(train, test){


  fit <- glm(Y~. ,family = binomial(), data = train)
  pred <- predict(fit, test, type = "response")

  cm <- confusionMatrix(data = as.factor(as.numeric(pred>0.5)), referen

  ac <- cm$overall['Accuracy']
  spe <- cm$byClass['Specificity']
  prec <- cm$byClass['Precision']
  rec <- cm$byClass['Recall']
  f1 <- cm$byClass['F1']

  metricas<-c(ac,spe,prec,rec,f1)
  return(metricas)

}


#-----------------------------------------------------------------

#The missing function inserts artificially missing
# values into the original data,
#by different mechanisms and in different amount

# q: amount of missing
# t: mechanism generator of missing
# (COM (complete), MCAR, MAR, MNAR)
# r: correlation parameter fized to multivariate
# Bernoulli distribution


missing<-function(data,q,t,r){

  l <- nrow(data)
  c <- ncol(data)

  if (t == "MCAR"){

    COR <- toeplitz(c(1, rep(r, (c-1)))) #Matriz de correlação
```

```
  if (r == 0){

    m <- matrix(rep(1,l*c),nrow = l, ncol = c)
    p_missing <- rbinom(l, p = 0.1, size = 1)
    p_sum <- sum(p_missing)

    index <- which(p_missing == 1)

    for (i in index) {

      m[i,] <- rbinom(c, p = (0.9/p_sum), size = 1)
    }


  }else if( r != 0 ){

    m <- rmvbin(l, margprob = c(rep(q,c)), bincorr = COR)

  }

  m[m == 0] <- NA


  b <- which(is.na(m),arr.ind =TRUE)
  e <- nrow(b)
  z <- 1

  while(z<=e){
    data[b[z,1],b[z,2]] <- NA
    z <- z+1
  }

} else if (t == "MAR"){

  COR <- toeplitz(c(1, rep(r, (c-1)))) #Matriz de correlação
  p_missing <- sort(runif(l, min = 0.5, max = 1))
  p_mean <- mean(p_missing)
  p_target <- 1-q
  fc <- p_mean/p_target
  p_missing <- p_missing/fc
  m <- c()

  for (w in p_missing){

    mis <- rmvbin(1, margprob = c(rep(1-w,c)), bincorr = COR)
    m <- rbind(m,mis)

  }
```

```
    m[m == 0] <- NA

    b <- which(is.na(m),arr.ind =TRUE)
    e <- nrow(b)
    z <- 1

    data[] <- data[order(data[,1]), ]

    while(z<=e){
      data[b[z,1],b[z,2]] <- NA
      z <- z+1
    }

  }else if(t == "MNAR") {

    COR <- toeplitz(c(1, rep(0.8, ceiling((c-1)/2)),
                rep(0.5, floor((c-1)/2)))) #Matriz de correlação
    m <- rmvbin(l, margprob = c(rep(q,c)), bincorr = COR)

    m[m == 0] <- NA

    b <- which(is.na(m),arr.ind =TRUE)
    e <- nrow(b)
    z <- 1

    while(z<=e){
      data[b[z,1],b[z,2]] <- NA
      z <- z+1
    }
  }
  return(data)
}


#-----------------------------------------------------------------
# The function solve_simulation get the accuracy
# value given simulation conditions.

# n: sample size (row)
# j:  input variables
# i: predictive power of the input variables
# dist: distribution of input variables
# p: correlation between input variables
# ef: effect of the input variables on the
# response variable (How much they explain)

solve_simulation <- function(j, n, dist, p, i, ef){
```

```
type <- c("MCAR","MAR","MNAR")

rho <- c(0.5,0.8)
amount <- c(0.9,0.7)

results <- c()
resultsFinal <- c()
s <- 0
u <- 0
while(u < 10){

  X <- generate_predictors( n = n , j = j,
                            dist = dist, rho = p)
  beta <- beta_calc (j = j, ef_size = ef, imp = i)
  eta <- X %*% beta

  while(s < 15){


    Y <- rbinom(nrow(X), size = 1, prob = binomial()$linkinv(eta))
    data <- data.frame(cbind(X, Y))

    intrain <- createDataPartition(y = data$Y,
                                   p= 0.7, list = FALSE)

    train <- data[intrain,]
    test <- data[-intrain,]
    met <- simul_model(train, test)

    res <- c(j, n, dist, p, i, ef,"COM", 1, 1, met)
    results <- rbind(results,res)

    for(t in type){
      for( r in rho){
        for(q in amount){

          M <- missing(data[,-ncol(data)],q,t,r)
          M$Y <- data$Y
          trainM <- M[intrain,]
          testM <- M[-intrain,]
          met <- simul_model(trainM, testM)

          res <- c(j, n, dist, p, i, ef, t, r, q, met)
          results <- rbind(results,res)
        }
      }
    }
```

```
      s <- s + 1
    }
    resultsFinal <- rbind(resultsFinal, results)
    u <- u + 1
  }
   colnames(resultsFinal)[1:9] <- c("col", "row",
                                    "distribution", "corr",
                                    "power","effect", "type",
                                    "corrT", "amount")

  name<-paste("~/",j, n, dist, p, i, ef,".RData")

  saveRDS(resultsFinal, file= name )#salvando

}

#-----------------------------------------------------------------
#Parallel cod
# The experimentos data frame contain the results of
# calibration effect size
# given simulation conditions.

require(doParallel)
ncl <- detectCores()
cl <- makeCluster(ncl)
registerDoParallel(cl)

system.time({
  foreach(w=1:nrow(experimentos),
            .packages=c("SimCorMultRes","caret","bindata"))
            %dopar% solve_simulation (j = experimentos[w,1],
                                      n = experimentos[w,2],
                                      dist = experimentos[w,3],
                                      p = experimentos[w,4],
                                      i = experimentos[w,5],
                                      ef = experimentos [w,6])
})
stopCluster(cl)
```