UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS
UNIDADE ACADÊMICA DE GRADUAÇÃO
CURSO DE ENGENHARIA DE PRODUÇÃO

EDUARDO MAZZUCCO RODRIGUES

PREDIÇÃO DE VIOLAÇÃO DO SLA E MINERAÇÃO DE DADOS DE PROCESSO DE HELP DESK EM UMA EMPRESA MULTINACIONAL DE SOFTWARE

São Leopoldo
2018

EDUARDO MAZZUCCO RODRIGUES

PREDIÇÃO DE VIOLAÇÃO DO SLA E MINERAÇÃO DE DADOS DE PROCESSO DE HELP DESK EM UMA EMPRESA MULTINACIONAL DE SOFTWARE

Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção do título de Bacharel em Engenharia de Produção, pelo Curso de Engenharia de Produção da Universidade do Vale do Rio dos Sinos - UNISINOS

Orientador: Prof: Ms Pedro Nascimento de Lima

São Leopoldo

2018

Dedico este trabalho à minha família.

RESUMO

A constante busca das empresas por competitividade evidencia a melhoria de processos como um aspecto fundamental. Dessa forma, a utilização de conceitos e técnicas para analisar os fatores que impactam os processos de negócio se tornam indispensáveis às rotinas de tomada de decisão. Com base nisso, esse trabalho aplicou técnicas de *process mining* para criar um modelo com o intuito de predizer sob quais condições a violação do *Service Level Agreement* (SLA) possui maior probabilidade de ocorrer, bem como analisar as variáveis que explicam esse cenário. Para tanto, foi realizado um estudo de caso com dados de cinco semanas do ano de 2018. Dos resultados obtidos, apenas duas das doze variáveis de entrada dos modelos preditivos se mostraram determinantes para predizer o cumprimento ou não do SLA. Dessa maneira, foi possível identificar um cenário em que a combinação das duas variáveis relevantes resulta em maiores probabilidades de violação do SLA. Ao decorrer do trabalho, foram verificadas diferenças entre a importância das variáveis de entrada quando analisadas sob dois cenários distintos. Nesse contexto, conclui-se que a personalização das ações de melhoria para atender às diferentes etapas do processo é um fator importante a ser considerado pela gestão.

Palavras-chave: *Process mining*. Análise Preditiva. Help Desk. *Service Level Agreement.*

LISTA DE FIGURAS

LISTA DE GRÁFICOS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE SIGLAS

BI	*Business Intelligence*
BPM	*Business Process Management*
IRT	*Initial Response Time*
PIB	Produto Interno Bruto
SLA	*Service Level Agreement*
SVM	*Support Vector Machines*

SUMÁRIO

1 INTRODUÇÃO

A relação entre o desenvolvimento do setor de serviços e o crescimento da economia global vem se fortificando nas últimas duas décadas, de modo que 74% do Produto Interno Bruto (PIB) dos países desenvolvidos foi representado por tal setor (BUCKLEY; MAJUMDAR, 2018). Adicionalmente, a contribuição dos serviços para o PIB mundial e do Brasil foi de, respectivamente, 65,10% e 63,1% (THE WORLD BANK GROUP, 2017).

Frente à esse cenário, a crescente busca das empresas por ganhos de participação de mercado evidencia a necessidade de iniciativas que aumentem o nível de satisfação dos clientes, as quais perpassam pela melhoria contínua dos processos de negócio (SIHA; SAAD, 2008). Nesse contexto, emerge a importância acerca do atendimento ao *Service Level Agreement* (SLA), que é o contrato entre um provedor de serviços e seus clientes com o objetivo de definir as métricas sob as quais os processos devem operar (HELO; GUNASEKARAN; RYMASZEWSKA, 2017). Se descumprido, o SLA pode acarretar em multas e impactar negativamente a satisfação dos clientes (LEITNER et al., 2010).

A necessidade de novos métodos de avaliação do impacto das decisões e planejamentos nas operações emergiu com a crescente criticidade do atendimento aos SLA's (AINSLIE et al., 2017). Nesse contexto, Van der Aalst, Weijters e Maruster (2004) afirmam que sistemas empresariais de informação foram integrados ao gerenciamento de fluxos, possibilitando a modelagem de processos de negócio, aumentando a demanda por sistemas de *Business Process Management* (BPM).

Além dos sistemas puramente dedicados ao BPM, outras classes de *softwares*, como o *Customer Relationship Management* (CRM), adotaram a tecnologia de *workflow management* (VAN DER AALST; WEIJTERS; MARUSTER, 2004). Um dos maiores desafios que esses sistemas enfrentam é complexidade da modelagem dos fluxos, exemplificada pela frequente necessidade de um consultor construir um modelo detalhado que descreva as rotinas de trabalho (TURNER et al., 2012; VAN DER AALST; WEIJTERS; MARUSTER, 2004).

Dumas e Maggi (2014) afirmam que as práticas tradicionais de BPM se baseiam em entrevistas e observações dos processos, mas essas técnicas funcionam para mapear os processos ideais, negligenciando possíveis desvios e exceções rotineiras. Considerando-se que a ideia central do BPM é baseada em processos

explicitamente definidos, executados e com suas informações preparadas e analisadas, as abordagens tradicionais de BPM não alcançam as expectativas de acadêmicos, consultores e empresas de *software,* uma vez que não acompanham a velocidade com que os processos mudam (DUMAS et al., 2013, p. 353; VAN DER AALST et al., 2003) .

Alinhada à isso, a tendência da comunidade do BPM em focar seus estudos nas etapas baseadas em dados evidencia a necessidade de utilização de outras técnicas analíticas, como o *process mining* (VAN DER AALST, 2016, p. 44). Sendo assim, o *process mining* surge como alternativa ao problema de que as informações sobre o estado atual dos processos são limitadas, uma vez que há diferenças entre o que é descrito pelos usuários e o que realmente ocorre nos sistemas (MANS et al., 2011).

Subordinado ao BPM, o *process mining* é a modelagem dos processos com informações coletadas aonde os mesmos ocorrem, agilizando os estudos uma vez que os dados são extraídos sem a necessidade de prévia descrição das rotinas de trabalho (VAN DER AALST; WEIJETERS; MARUSTER, 2004). Adicionalmente, avanços na pesquisa do *process mining* possibilitaram descobrir, analisar e melhorar processos de negócios baseados em dados de eventos (VAN DER AALST, 2012). Consequentemente, o aumento da aplicação de tais técnicas surge a partir da necessidade das companhias em aprenderem mais sobre seus processos (TIWARI; TURNER; MAJEED, 2008).

Nas fases iniciais do BPM, os processos são tipicamente modelados com o auxílio de um consultor de negócios, o qual é contratado para implementar as melhorias propostas pela gestão. Por outro lado, o objetivo do *process mining* é coletar os dados de onde eles são gerados, afim de suportar a modelagem e a análise dos fluxos (VAN DER AALST et al., 2003).

Baseando-se no exposto acima, o *process mining* surge como uma abordagem que, juntamente ao BPM tradicional, tem a capacidade de aprimorar o gerenciamento de processos de negócios através de modelagens mais rápidas e precisas. Em um cenário onde os dados são gerados exponencialmente e os processos mudam rapidamente, essa técnica concede às empresas diferentes perspectivas para embasar a tomada de decisão.

Mediante esses fatos, o tema da pesquisa consiste na análise de um processo de help desk à luz de técnicas de *process mining*. Nas próximas seções, será

apresentado o problema de pesquisa, seguido dos objetivos geral e específicos, além da justificativa do estudo. O capítulo seguinte explora o referencial teórico sobre o qual o tema da pesquisa está fundamentado. Finalmente, a metodologia utilizada no estudo, a análise e discussão dos resultados e as conclusões serão apresentadas.

1.1 Problema de pesquisa

As empresas têm investido em sistemas que coletam dados de variadas fontes, como o CRM, com o intuito de garantir que transações e ações fiquem registradas para posterior análise (DAVENPORT, 2006). A melhoria de processos de negócio é um dos assuntos mais presentes entre os executivos do ramo de tecnologia da informação. Consequentemente, crescem a demanda e as expectativas relacionadas ao BPM (ROSEMANN; VOM BROCKE, 2010).

Nesse contexto, o gerenciamento de processos de negócio ganha visibilidade à medida que a satisfação dos clientes é impactada pelas atividades das empresas. Alinhadas a isso, as práticas de modelagem e monitoramento de processos são importantes iniciativas do BPM (ROSEMANN; VOM BROCKE, 2010). Apesar da atenção dada pelas empresas à tais práticas, a aplicação delas ainda não é diretamente conectado aos processos operacionais (VAN DER AALST; LA ROSA; SANTORO, 2016).

Considerando-se que os softwares são objetos de estudo do BPM e que a competitividade torna os help desks importantes na percepção de qualidade dos serviços, os processos de suporte precisam contribuir positivamente para a satisfação dos clientes. A rotina de tais processos é pouco previsível, uma vez que os recursos podem vir a confrontar diferentes tipos de usuários e situações. Sob essas circunstâncias, extrair conhecimento de execuções passadas através de coleta, tratamento e análise de dados pode ajudar a predizer potenciais problemas nos processos (GRIGORI et al., 2004).

É comum que empresas de tecnologia devam entregar serviços dentro de tempos estabelecidos através de SLA’s (REIJERS, 2006). A empresa analisada estabelece SLA’s juntamente aos seus clientes, de modo que o monitoramento do cumprimento dos níveis de serviço é feito através de relatórios diários gerados no CRM.

Um fator que tem o potencial de prejudicar o atendimento ao SLA de um setor de suporte a software é a liberação de atualizações no sistema (*release*). Trimestralmente, o software da área em questão é submetido a tais *releases*, que geram picos de demanda através de ligações e e-mails. Isso evidencia a importância dos registros no CRM, uma vez que o processo possui alto volume de inputs e necessita de controle acerca das comunicações com os clientes para garantir o atendimento aos SLA's.

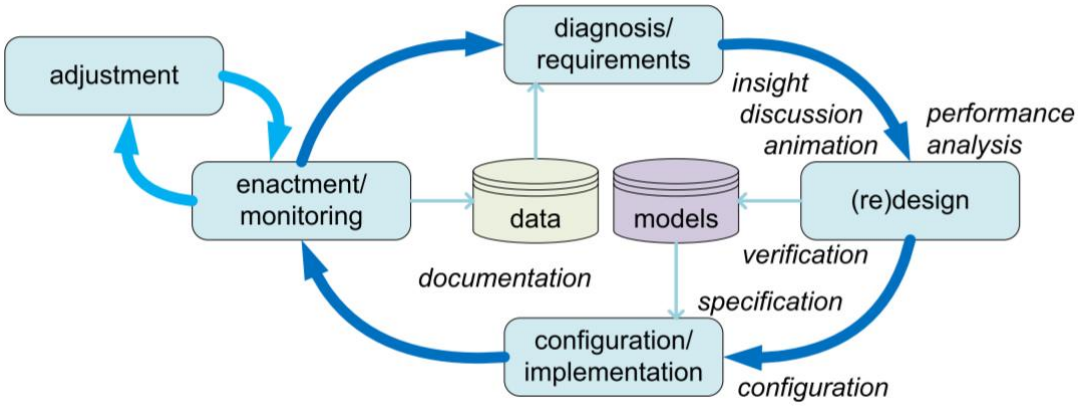
Alinhadas às necessidades de controlar e garantir o atendimento aos SLA'a, as empresas de sucesso são, frequentemente, as que mantém históricos de comunicações com os clientes para posteriores análises e ações de melhoria (DAVENPORT, 2006). Dessa forma, o BPM se posiciona entre a gestão dos processos e os dados registrados no sistema, afim de suportar a tomada de decisão de rotinas de gerenciamento de processos relacionados à tecnologia da informação (WESKE, 2007, p. 4).

Dada a necessidade de monitoramento dos processos, o *Business Intelligence* (BI) surge como alternativa para que os processos de negócio sejam analisados quantitativamente através de relatórios (GRIGORI et al., 2004). O problema de se utilizar o BPM tradicional juntamente ao BI é que essa combinação pressupõe a existência de processos formalizados, que são modelados através de conversas com especialistas do processo em questão (VAN DER AALST; WEIJTERS; MARUSTER, 2004).

Adicionalmente ao problema anterior, estão as características do processo de help desk analisado: alto volume, SLA constante e alta variabilidade da demanda. Sob esse contexto, por se basear em modelagens de processos ideais, a abordagem tradicional do BPM pode não explicitar as reais causas que possam estar impactando os indicadores do BI.

A assertividade das ações de melhoria, que perpassam pela modelagem do processo, é de suma importância para a empresa em questão, uma vez que os clientes são diretamente impactados pelo não cumprimento do SLA. Nesse contexto, o *process mining* surge como um conjunto de técnicas para modelar e analisar os processos baseado nos registros dos mesmos (VAN DER AALST; WEIJTERS; MARUSTER, 2004), podendo ser aplicado nas fases de diagnóstico, monitoramento e remodelagem do ciclo de vida do BPM.

Figura 1 - Ciclo de vida do BPM



Fonte: Van der Aalst (2016, p. 31)

A Figura 1 mostra que cada fase do ciclo de vida do BPM possui alguma interação com os dados ou com o modelo. Visto que a violação do SLA pode ser identificada a partir dos relatórios do CRM, a presente pesquisa atuará nas fases de *diagnosis/requirements* e *(re)design*, que juntas possibilitam a análise do estado atual e a identificação de pontos de melhoria do processo. Dito isso, para que o atendimento ao SLA possa ser estudado, se fazem necessárias as modelagens do processo, além das análises descritivas e de performance.

Dado o contexto apresentado, surge a questão central que motiva a realização desse trabalho: Como identificar as variáveis que possam impactar no atendimento dos níveis de serviço, de modo a servirem de inputs para modelos preditivos?

Na seção seguinte, serão apresentados os objetivos gerais e específicos.

1.2 Objetivos

1.2.1 Objetivo Geral

Este trabalho visa analisar os dados de um processo de help desk, de modo a predizer sob quais condições o SLA é atendido ou violado, bem como determinar as variáveis que melhor explicam os resultados das predições.

1.2.2 Objetivos Específicos

a) Modelar o processo atual com base no *event log;*

b) Identificar as variáveis que possam impactar na performance do sistema;
c) Avaliar o impacto de tais variáveis no atendimento ou violação do SLA;
d) Avaliar o impacto dessas variáveis no tempo de atravessamento.

1.3 Justificativa

1.3.1 Justificativa acadêmica

A justificativa acadêmica do estudo é baseada na busca por uma linha de pesquisa que enriqueça os conhecimentos acerca do tema estudado. Nesse contexto, foi realizada uma revisão sistemática da literatura que tem como objetivo mapear, encontrar, avaliar criticamente, consolidar e agregar aos estudos realizados, bem como evidenciar lacunas não exploradas (CAMARGO; MORANDI, 2015). Sendo assim, o protocolo de pesquisa referente à revisão sistemática é apresentado no Quadro 1.

Quadro 1 – Protocolo de pesquisa da revisão sistemática da literatura

***Framework* conceitual**	Revisão sistemática com o objetivo de encontrar estudos sobre *process mining* e análise preditiva em help desks
Contexto	Análise da aplicação de técnicas de *process mining* e preditivas em help desks
Horizonte	Estudos a partir de 1998
Teorias correntes	Sem limitações
Idiomas	Português, inglês e espanhol
Questão de revisão	*Process mining* e análise preditiva de atendimento aos SLA's em help desks
Critérios de inclusão	Estudos que demonstram aplicações das técnicas de *process mining*, que contenham as variáveis de entrada de modelos preditivos ou os algoritmos utilizados
Critérios de exclusão	Estudos que não abordam problemas de negócio, não apresentem relação com o suporte à software quando utilizado o termo "help desk", não citem os algoritmos preditivos ou que não estejam nas vinte primeiras páginas sob o filtro de vinte resultados por página
Termos de busca	workflow mining AND business process management
	process mining
	process mining AND call center
	business process management AND process mining
	workflow mining AND call center
	process mining AND business process management
	workflow mining
	business process management AND call center
	business process management AND call center
	business process management AND workflow mining
	Help desk
	Help desk AND predict
	Help desk AND service level agreement
	Service level agreement AND predict
	Service level agreement AND predictive
Fontes de busca	Scopus
	EBSCO

Fonte: Adaptado pelo autor com base em Camargo e Morandi (2015).

Baseando-se no protocolo de pesquisa, cada termo de busca mencionado foi procurado nas plataformas EBSCO e Scopus. Foram encontrados 72.992 resultados que, submetidos aos critérios apresentados no Quadro 2, compuseram o conjunto de estudos a serem lidos após a avaliação de seus títulos e abstracts.

Quadro 2 – Critérios de leitura dos estudos

Critérios de inclusão	**Critérios de exclusão**
Pesquisas que contenham aplicações práticas acerca do tema da pesquisa	Idioma diferente dos especificados no protocolo de pesquisa
Pesquisas que analisem e discutam os resultados encontrados	Arquivos inacessíveis
Pesquisas que expliquem os procedimentos metodológicos	Resultados duplicados
Bibliografias clássicas que sejam referências nos seus temas	Estudos além da vigésima página de resultados do termo de pesquisa (filtro por relevância)

Fonte: Elaborado pelo autor.

Dados os critérios de leitura apresentados, foram lidos 66 estudos e, de acordo com o protocolo de pesquisa**,** 51 foram considerados relevantes. O site *ResearchGate* foi utilizado para localizar e acessar estudos sob o método ‘Bola de neve’, o qual consiste na leitura das obras que foram referenciadas nos textos lidos. Sendo assim, Quadro 3 apresenta os resultados da revisão sistemática da literatura.

Quadro 3 – Resultados da revisão sistemática da literatura

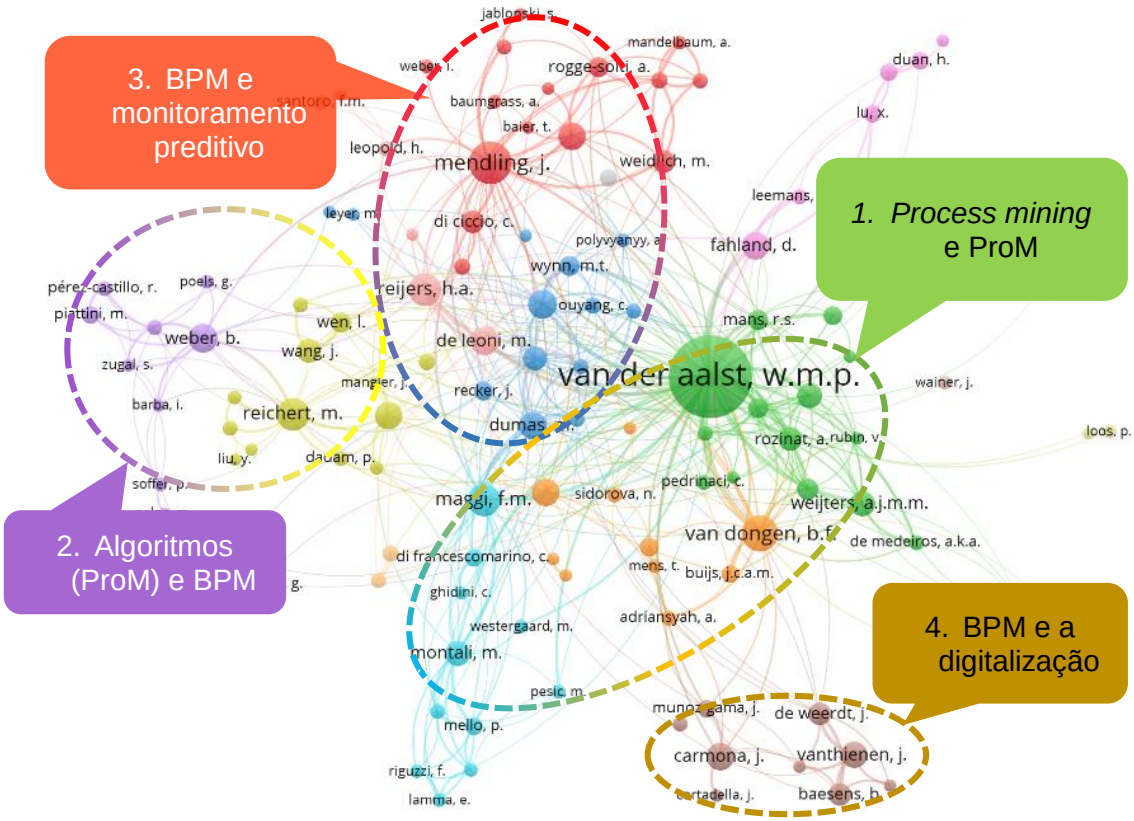
EBSCO					
Termo de pesquisa	**Índice**	**Títulos**	**Títulos relacionados**	**Abstracts relacionados**	**Artigos relacionados**
"process mining"	**Livre**	**5162**	**34**	**15**	**11**
"process mining" AND "call center"	**Livre**	**103**	**9**	**4**	**1**
"business process management" AND "process mining"	**key words e título, respectivamente**	**47**	**8**	**7**	**2**
"workflow mining" AND "call center"	Livre	13	2	0	0
"process mining" AND "business process management"	**key words e título, respectivamente**	**15**	**5**	**3**	**2**
"workflow mining"	título	104	4	3	0
"business process management" AND "call center"	key words e título, respectivamente	0	0	0	0
"business process management" AND "call center"	Título e livre, respectivamente	20	3	2	0
"business process management" AND "workflow mining"	key words e título, respectivamente	1	0	0	0
"business process management" AND "workflow mining"	key words e título, respectivamente	1	1	0	0
“help desk”	Livre	60622	8	1	0
“help desk” AND “predict”	Livre	3	1	0	0
Help desk AND service level agreement	**Título e livre, respectivamente**	**20**	**1**	**0**	**3**
Service level agreement AND predict	**Livre**	**388**	**5**	**4**	**2**
“service level agreement” AND predictive	Livre	29	2	2	0
“Bola de neve”	**Referências dos artigos relacionados**	**14**			

Scopus					
Termo de pesquisa	**Índice**	**Títulos**	**Títulos relacionados**	**Abstracts relacionados**	**Artigos relacionados**
{process mining}	**Livre**	**5207**	**22**	**15**	**4**
{process mining} AND {call center}	Livre	22	7	4	0
{business process management} AND {process mining}	**key words e título, respectivamente**	**69**	**16**	**13**	**2**
{workflow mining} AND {call center}	Título e livre, respectivamente	0	0	0	0
{process mining} AND {business process management}	**key words e título, respectivamente**	**15**	**5**	**6**	**1**
{workflow mining}	**título**	**55**	**7**	**3**	**2**
{business process management} AND {call center}	key words e título, respectivamente	0	0	0	0
{business process management} AND {call center}	Título e livre, respectivamente	4	3	2	0
{business process management} AND {workflow mining}	key words e título, respectivamente	3	2	2	0
{workflow mining} AND {business process management}	TITLE-ABS-KEY	14	3	2	0
{help desk}	TITLE-ABS-KEY	709	5	0	0
{help desk} AND predict	TITLE-ABS-KEY	4	1	0	0
Help desk and service level agreement	**TITLE-ABS-KEY**	**15**	**1**	**1**	**1**
Service level agreement AND predict	**TITLE-ABS-KEY**	**305**	**9**	**7**	**4**
{service level agreement} AND predictive	**TITLE-ABS-KEY**	**42**	**1**	**1**	**1**
“Bola de neve”	**Referências dos artigos relacionados**	**16**			

Fonte: elaborado pelo autor.

Realizadas as leituras, foram percebidas diversas vertentes de pesquisa, as quais perpassam, principalmente, pelo desenvolvimento de algoritmos para softwares de *process mining* (ProM). Além disso, alguns pesquisadores participaram de diversos estudos, de modo que diferentes abordagens foram estudadas por um seleto conjunto de autores. Para melhor compreensão desse cenário, foi realizada uma análise bibliométrica de co-citação a partir do termo "*process mining*".

Figura 2 – Mapa de co-citação do termo *process mining*



Fonte: Elaborado pelo autor.

A Figura 2 apresenta as relações encontradas pelo software VOSviewer (VAN ECK; WALTMAN, 2010) a partir de 2000 resultados extraídos da base Scopus. Para a geração do mapa de co-citação, foram considerados autores que foram referenciados mais de 40 vezes. Com base nesses parâmetros, o Quadro 4 mostra as principais vertentes de pesquisa identificadas na análise bibliométrica relacionada ao termo *"process mining"*. A relação entre as linhas de estudo e os principais autores foi encontrada a partir da combinação da revisão sistemática da literatura com o mapa de co-citação.

Quadro 4 – Linhas de estudo dos clusters

Cluster	Linha de estudo	Principais autores
1	Berço do *process mining* e desenvolvimento da ferramenta ProM	Wil van der Aalst, Fabrizio Maria Maggi, Marco Montali, A.J.M.M. Weijters, Boudewijn van Dongen
2	Desenvolvimento de algoritmos para o ProM e estudos de novas tecnologias aplicadas ao BPM	Barbara Weber, Manfred Reichert, Lijie Wen e Jianmin Wang
3	Monitoramento preditivo de processos de negócio	Jan Mendling, Marlon Dumas, Halo A. Reijers
4	Estudo do BPM no contexto da transformação digital global	Josep Carmona e Jan Vanthienen

Fonte: Elaborado pelo autor.

Finalizada a revisão sistemática da literatura, foi verificada a oportunidade de preenchimento de uma lacuna de pesquisas nacionais e internacionais acerca do tema proposto, evidenciando a relevância do mesmo. Muitos estudos focam no desenvolvimento de algoritmos para modelar situações específicas, mas os casos reais são utilizados como testes de validação modelos. Assim, esse trabalho visa aplicar técnicas de *process mining* para analisar e modelar um processo de help desk, além de explicar o comportamento das variáveis a partir das predições de violação do SLA e tempo de fechamento dos *cases*.

Van der Aalst (2016, p. 44) afirma que as abordagens anteriores do BPM eram focadas nas etapas de modelagem dos processos. Técnicas tradicionais de modelagens são suficientes nas fases iniciais do BPM, mas o desenvolvimento da indústria 4.0 traz consigo a necessidade de monitorar e remodelar os processos em tempo real, os quais são possibilitados através do *process mining* uma vez que dados de eventos de várias fontes podem ser analisados (VAN DER AALST; LA ROSA; SANTORO, 2016). Dito isso, a presente pesquisa pode contribuir com a análise de um caso real de aplicação do *process mining* a partir de diferentes relatórios, de modo a gerar entendimento do processo com a utilização de dados pulverizados.

O BPM tradicional sempre se baseou em estimativas de dados e coletas manuais de informações para descobrir, analisar e redesenhar os processos. Isso pode levar a erros na identificação de gargalos e causas dos problemas (DUMAS;

MAGGI, 2014). Os autores ainda afirmam que entrevistas e *workshops* funcionam para descobrir os processos ideais, mas falham ao evidenciar detalhes como exceções e desvios que geralmente caracterizam as rotinas diárias. Dessa forma, o estudo pode contribuir com a análise de um processo real sem a utilização de entrevistas e *workshops*, para que os desvios do fluxo ideal sejam considerados.

Turner et al. (2012) baseiam-se na pesquisa de Van der Aalst, Weijters e Maruster (2004) para conceituar três tipos de *process mining:* a descoberta dos processos, o diagnóstico dos processos o remodelagem dos processos. Tais autores conduziram um estudo que avaliou diversas ferramentas de *process mining* disponíveis no mercado. Sendo assim, o presente trabalho pode contribuir com a utilização de uma ferramenta diferente para a aplicação das técnicas de *process mining*.

Jiménez (2017) e De Vries et al. (2017) aplicaram a descoberta de processos e a verificação de conformidade identificar o estado atual e os desvios dos fluxos estudados. Os autores conduziram os estudos de modo a encontrar algoritmos que modelassem os processos de modo satisfatório, mas não realizaram predições para analisar as variáveis que impactam os cenários estudados. A presente pesquisa pode contribuir com predições para que o impacto das variáveis no processo possa ser explicado.

Chuchaimongkhon, Porouhan e Premchaiswadi (2017) estudaram um call center bancário afim de mostrar que o *process mining* não é útil apenas para a modelagem gráfica dos processos. O estudo objetivou facilitar o entendimento acerca do processo analisado, mas se limitou a avaliar os tempos de ciclo e de espera do sistema. Dessa maneira, o presente trabalho pode contribuir com a combinação de análises descritivas com a avaliação da performance do processo.

Den Hertog (2008) e Polato et al. (2018) conduziram estudos nos quais abordagens preditivas foram aplicadas afim de se estimarem os tempos de fechamento dos casos analisados. Como o estudos foram focado na aderência dos algoritmos para que novas técnicas preditivas pudessem ser desenvolvidas, há uma lacuna a ser preenchida em relação às implicações práticas das predições. Sendo assim, o trabalho pode contribuir com a análise das variáveis que possam impactar a performance de um caso real, de modo a evidenciar as possíveis implicações gerenciais das predições.

Ainslie et al. (2017) utilizaram redes neurais para predizer os níveis de serviço com base nas atividades completadas, iniciadas, em processamento e na capacidade de um determinado processo. Os autores chegaram em resultados satisfatórios no que tange a acurácia do algoritmo, mas não analisaram o comportamento e o impacto das variáveis de entrada do modelo. Considerando que os autores propuseram o algoritmo *Support Vector Machines* (SVM) como base para futuras pesquisas, a aplicação dessa técnica no presente trabalho, juntamente à análise do comportamento das variáveis, podem ser contribuições relevantes.

Na próxima seção, será apresentada a justificativa empresarial.

1.3.2 Justificativa empresarial

No âmbito da organização, o resultado das análises realizadas nesse estudo é relevante, considerando-se a importância de atendimentos ágeis e cumprimento dos SLA's acordados. A gestão atual concentra os esforços no controle dos tempos de resposta de maneira reativa, através de relatórios diários e recursos flexíveis que podem ser alocados para trabalhar em dias de maior demanda.

O uso do *process mining* contribui para modelagens baseadas em evidências dos estados reais dos processos (DUMAS; MAGGI, 2014). Sendo assim, a presente pesquisa poderá contribuir para que os gestores tenham uma visão detalhada acerca dos caminhos e desvios do processo. Além disso, os resultados do modelo descoberto serão utilizados como inputs em modelos preditivos, de modo a prover um melhor entendimento sobre as variáveis que impactam o sistema.

Os relatórios gerados no sistema permitem a verificação do estado momentâneo dos atendimentos e, se algum atendimento está com o SLA a expirar, um recurso é alocado para tal caso para que um encaminhamento inicial seja dado. O *process mining*, associado aos modelos preditivos, pode contribuir nesse contexto uma vez que o processo será descoberto como ele realmente é realizado e os tempos de atravessamento serão classificados dadas as variáveis selecionadas.

Considerando-se as ações de melhoria dos processos, o *process mining* pode contribuir com modelagens rápidas dos fluxos baseadas nos *event logs*. No cenário atual, as melhorias ocorrem de caráter corretivo e são implementadas sob a modalidade de tentativa e erro. Dessa forma, a descoberta de processos poderia

suportar a gestão na tomada de decisão uma vez que oferece visibilidade do processo como ele realmente é executado (DUMAS et al., 2013).

Na organização estudada, há grande disponibilidade de dados e o CRM possui relatórios customizados sob a forma de *event logs*. Dessa maneira, as técnicas de *process mining* podem ser amplamente aplicadas ao longo do tempo com exceção do monitoramento de anomalias no processo baseado em modelos anteriores, visto que o processo não possui modelagens prévias. Aliada à disponibilidade de dados e aos avanços das técnicas de *process mining,* a predição de tempos é uma solução de grande valor considerando a necessidade de atendimento dos SLA's e as expectativas de satisfação dos clientes.

Sendo assim, a presente pesquisa aplicará as técnicas de *process mining* para descobrir o fluxo que descreva a realidade com fidelidade e sirva de base para predições de violação do SLA e tempo de fechamento dos *cases*.

1.4 Delimitações

As delimitações do presente trabalho são relacionadas à abrangência da análise e aos fatores limitantes que podem impactar os resultados. Visto que as técnicas de *process mining* pressupõem dados de eventos para gerar os modelos, esse trabalho irá desconsiderar os telefonemas dos clientes em decorrência da impossibilidade de extração de eventos acerca dos mesmos.

Esta pesquisa também não considerará a disponibilidade dos recursos ao longo do dia como parte das análises. Os atendentes disponíveis serão contabilizados de maneira que a capacidade diária seja expressa em termos de recursos por dia, mas a dedicação dos mesmos às diferentes atividades será desconsiderada.

Finalmente, o estudo agregará atividades que resultam em interações pontuais com o CRM para melhorar a interpretação dos modelos e preservar a identidade da empresa. Dessa forma, todas as alterações que não acarretem atividades subsequentes e que sejam interações pontuais com o intuito de atualizar os *cases* sejam agregadas em uma única atividade.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os conceitos relevantes em relação ao tema estudado. Serão definidas as três áreas de aplicação do *process mining* (descoberta, monitoramento e remodelagem), além das relações entre *data mining*, *event logs* e *process mining*.

2.1 *Business process management* (BPM)

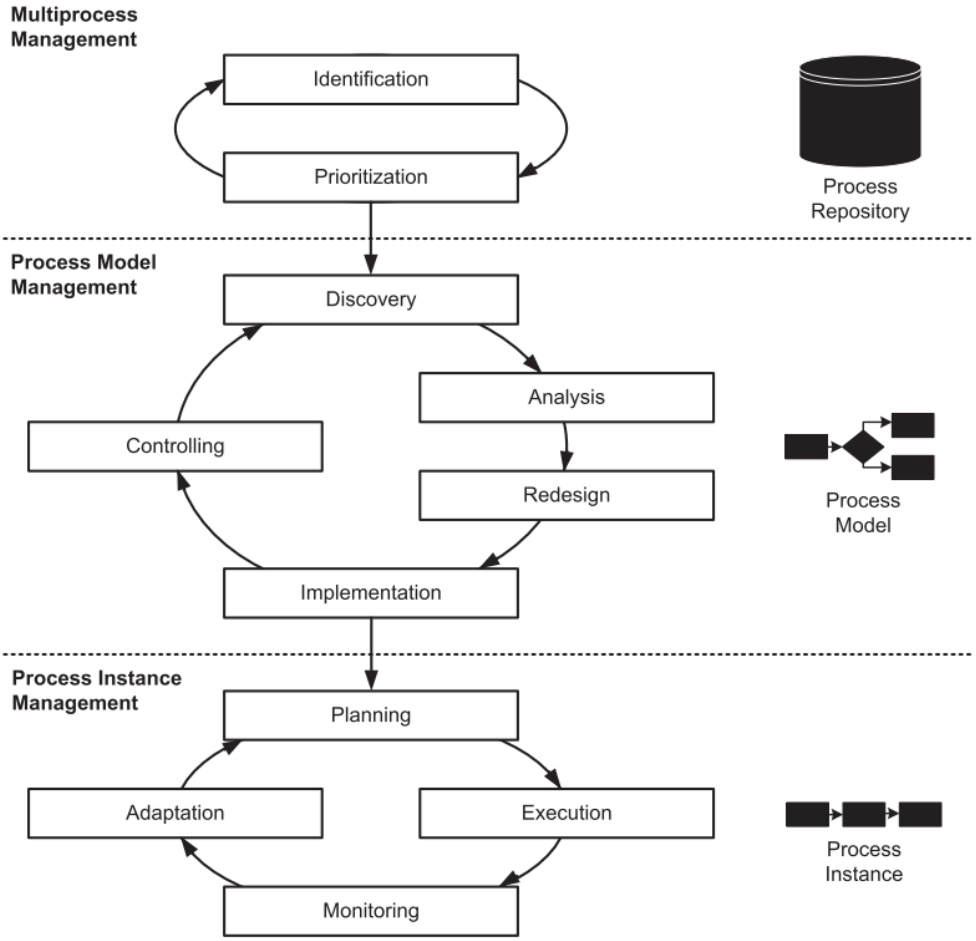
Segundo Dumas et al. (2013, p. 1), BPM é a ciência de observar como o trabalho é feito em uma organização afim garantir resultados consistentes e aproveitar as oportunidades de melhoria. Também definido como sendo a disciplina que combina abordagens de modelagem, execução, controle, medida e otimização de processos de negócios (VAN DER AALST, 2016, p. 44), o BPM ganhou destaque com o crescimento dos sistemas de informação e com a divisão dos processos de negócio em diversos departamentos dentro das empresas (MENDLING et al., 2017).

Conforme Weske (2007, p. 3), o BPM é baseado na observação de que cada produto que uma empresa entrega ao mercado é o resultado de um número de atividades performadas. Os processos de negócio são formados por diversas dessas atividades as quais são origem aos eventos, que são definidos por Dumas et al. (2013, p. 3) como o registro sem duração de ações que ocorrem automaticamente. Por exemplo, a chegada de um caminhão em uma fábrica é um evento.

Além dos eventos e atividades, os processos de negócio são compostos por decisões e recursos, sejam eles humanos, físicos ou virtuais (DUMAS et al., 2013, p. 4). O resultado dessa sequência de atividades consumidoras de recursos é o último elemento essencial dos processos de negócio. Sendo assim, um processo é uma sequência de atividades no tempo e no espaço, com início e fim, entradas e saídas (LACERDA; RODRIGUES; SILVA, 2009).

Mendling et al. (2017) propõem uma divisão do BPM em três níveis (Figura 3). No primeiro nível, os resultados do gerenciamento de processos são frequentemente armazenados em um repositório central de processos. O segundo nível é focado no gerenciamento de um único processo e é conhecido como o ciclo de vida do BPM. O foco do terceiro nível é o gerenciamento dos pré-requisitos e tempos aos quais os processos precisam atender.

Figura 3 – Três níveis do BPM



Fonte: Mendling et al. (2017)

Weske (2007, p. 6) afirma que o BPM é baseado em um conjunto de técnicas, que suportam a modelagem e a análise dos processos de negócio, os quais são compostos por atividades coordenadamente realizadas em um ambiente organizacional para alcançar um objetivo. A aplicação das técnicas de BPM nos processos de negócio são possíveis através de um *Business Process Management System,* que é um software genérico que objetiva a modelagem dos processos para coordenar a integridade dos mesmos.

2.2 *Event logs*

Turner et al. (2012) afirmam que os *event logs* possuem dados reais dos processos e esses dados são armazenados pelos sistemas de informação de acordo com a Tabela 1. Os processos são compostos por *cases*, que por sua vez consistem em eventos ordenados de acordo com os *timestamps*, podendo possuir atributos

como usuários, atividades e custos (VAN DER AALST; LA ROSA; SANTORO, 2016). Norambuena e Zepeda (2017) reforçam o conceito dizendo que *event logs* são registros de todos os eventos realizados nos processos.

Tabela 1 – Exemplo de *event log*

Case_id	*Event_id*	*Timestamp*	Usuário	Atividade
1	12	30-12-2012:10.02	Usuário 1	Registro
1	123	30-12-2012:10.04	Usuário 2	Submissão
1	1234	30-12-2012:11.00	Usuário 1	Revisão
1	12345	30-12-2012:11.03	Usuário 1	Aprovação

Fonte: Elaborado pelo autor.

Van der Aalst (2016, p. 5) destaca a importância dos *event logs* em um cenário de crescimento exponencial de geração de dados, no qual o termo *Big Data* ganha relevância e a conectividade entre os sistemas de gestão, redes sociais e objetos físicos é possível em decorrência da internet. Nesse contexto, Van der Aalst (2016, p. 129-130) conceitua o termo *Internet of Events* (IoE), que é consequência da operacionalização de diversas outras abordagens geradoras de eventos, como as apresentadas no Quadro 5.

Quadro 5 – Abordagens que produzem eventos

Abordagem	**Conceito**
Internet of Content	Toda a informação criada para melhorar o conhecimento acerca de determinados assuntos
Internet of People	Todos os dados relacionados às interações sociais
Internet of Things	Todos os objetos físicos conectados à internet
Internet of Locations	Todos os dados associados à posições geográficas

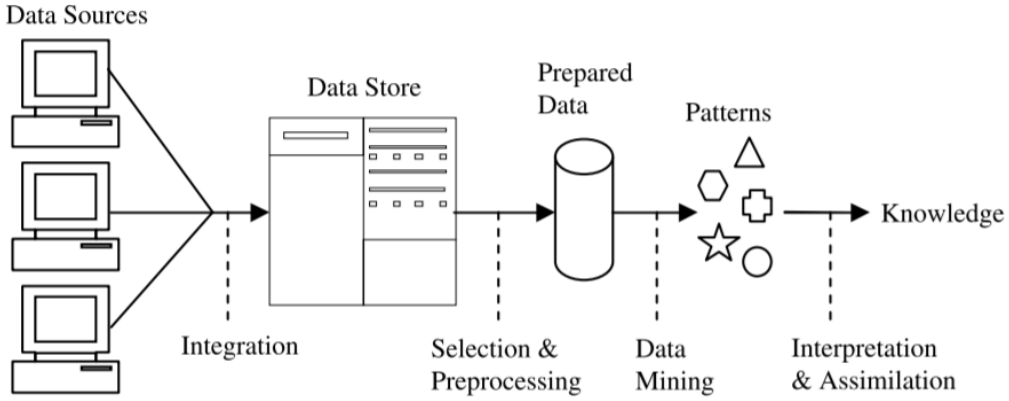
Fonte: Van der Aalst (2016, p. 129-130).

Os *event logs* podem ser definidos como o ponto de partida do *workflow mining*, uma vez que o objetivo principal do mesmo é extrair conhecimento de um processo de negócio a partir dos *event logs* (VAN DER AALST et al., 2010). Dito isso, Van der Aalst (2012) complementa afirmando que a presença de tais dados permite novas formas de análises de processos baseadas em fatos observados, ao invés de modelos manualmente feitos. Sendo assim, o ponto de partida do *process mining* é o *event log*.

2.3 *Data mining*

Conforme Hand et al. (2001, p. 6), *Data mining* é a análise de conjuntos de dados observacionais para encontrar relações não suspeitas e resumir os dados de modo compreensível e útil. Chen et al. (1996) contribuem afirmando que *data mining* significa um processo de extração de informações implícitas, desconhecidas e potencialmente úteis das bases de dados. Fayyad et al. (1996) completam o conceito dizendo que *data mining* é a aplicação de algoritmos específicos para extrair padrões dos dados, como exemplificado na Figura 4.

Figura 4 – Representação do processo de *data mining*



Fonte: Bramer (2013, p. 2)

Os conjuntos de dados utilizados em *data mining* são medidas abstraídas dos processos, por exemplo, para cada objeto de um conjunto, há medidas associadas ao mesmo (HAND et al., 2001, p. 8). Os dados de entrada são tipicamente tabelas e as saídas podem ser regras, *clusters*, gráficos, equações, padrões, entre outros (VAN DER AALST, 2016, p. 89).

Hand et al. (2001, p.6) afirmam que os conjuntos de dados examinados em *data mining* são usualmente grandes, de modo a diferenciar a técnica da clássica análise exploratória dos dados praticada por estatísticos. Além disso, os problemas relacionados a conjuntos de dados maiores são mais frequentes, visto que o armazenamento, o acesso, a representatividade, o período de análise e as relações entre os eventos não podem ter poucas chances de representar a realidade.

Um dos maiores desafios da *data mining* é o gerenciamento de diferentes tipos de dados, pois não se pode assumir que as bases de dados são homogêneas e livres de erros (BRAMER, 2013, p. 13; CHEN et al., 1996). Alinhado a isso, Van der Aalst

(2016, p. 92) complementa afirmando que os dados são tipicamente processados antes da aplicação de qualquer técnica de *data mining*, por exemplo, linhas e colunas podem ser removidas. Fayyad et al. (1996) complementam o conceito dizendo que a limpeza dos dados inclui a remoção de anomalias e as decisões acerca da falta de dados.

Resumidamente, Hand et al. (2001, p. 7) propõem os passos abaixo como etapas do processo de descobrimento de relações em um conjunto de dados:

a) Determinar a natureza e a estrutura da representação a ser usada;
b) Decidir como quantificar e comparar como diferentes representações se adequam aos dados;
c) Escolher um algoritmo que otimize o nível das representações;
d) Decidir quais princípios de gerenciamento de dados são necessários para implementar os algoritmos eficientemente.

2.3.1 Classificação das técnicas de *data mining*

Segundo Chen, Han e Yu (1996) diversos critérios de classificação podem ser utilizados para categorizar os métodos de *data mining*. O mesmo autor exemplifica as classificações da seguinte maneira:

a) Tipos de bases de dados a serem exploradas;
b) Tipo de conhecimento a ser extraído;
c) Tipos de técnicas de mineração utilizadas.

Considerando-se a afirmação de que diferentes critérios podem ser utilizados na classificação das técnicas de *data mining*, a literatura evidencia um padrão (Quadro 6) emergente de categorização baseado na existência de um atributo que sirva de referência para as predições dos algoritmos (BRAMER, 2013, p. 4; HAND et al., 2001, p. 103; VAN DER AALST, 2016, p. 103).

Quadro 6 – Classificação de técnicas de *data mining*

Autores	**Aprendizado supervisionado**	**Aprendizado não-supervisionado**
Van der Aalst (2016)	“[...] pressupõe variáveis com comportamentos conhecidos [...] o objetivo é explicar variáveis dependentes através do comportamento de variáveis independentes.”	“[...] assumem-se dados com comportamentos desconhecidos [...] as variáveis não são divididas entre respostas e predições.”
Chen, Han e Yu (1996)	“Atributos de comportamento conhecido são considerados nos cálculos [...]”	“[...] objetos analisados não são pré-estabelecidos [...] distâncias geométricas não podem ser medidas mas objetos representam uma certa classe conceitual baseado no grupo a que pertencem.”
Bramer (2013)	“[...] há um atributo específico designado e o objetivo é usar os dados para predizer o valor daquele atributo nos eventos ainda não vistos.”	“Dados que não possuem nenhum atributo especial designado [...] o objetivo é simplesmente extrair o maior número de informações dos dados disponíveis”
Hand, Mannila e Smyth (2001)	“[...] possuem o objetivo específico de permitir a predição dos valores desconhecidos de uma variável de interesse dados valores conhecidos de outras variáveis”	“[...] é uma descrição local, aplicada à alguns subconjuntos amostrais, podendo mostrar apenas poucas evidências de comportamento de dados ou caracterizar estruturas persistentes e não usuais nos dados.”

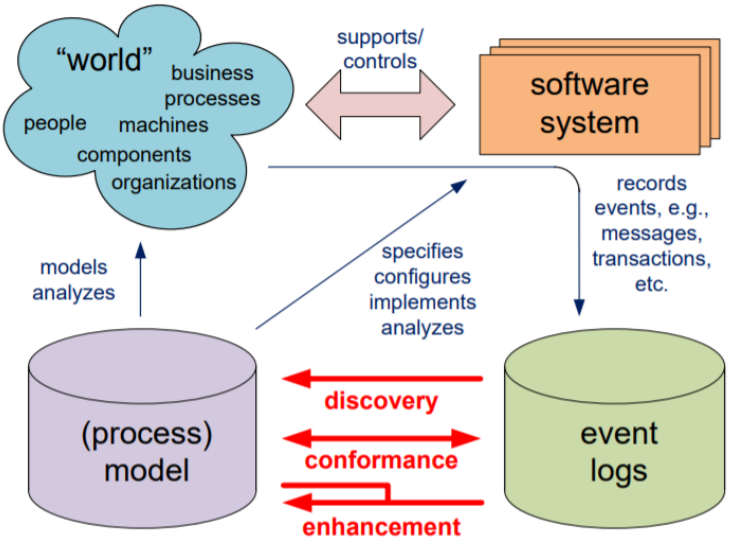
Fonte: Elaborado pelo autor com case em Bramer (2013); Chen et al. (1996); Hand et al. (2001) e Van der Aalst (2016)

2.4 *Process mining*

O *process mining* é definido por Van der Aalst et al. (2004) como sendo a modelagem dos fluxos de processo a partir dos *event logs* dos sistemas, permitindo descobrir como as pessoas realmente trabalham. Em termos de disciplina, o *process mining* está situado entre o *data mining* e a modelagem de processos (VAN DER AALST, 2016). Tiwari et al. (2012) reforçam o conceito dizendo que o *process mining* é uma técnica para extrair informações dos *event log* através do uso de práticas de *data mining*.

Segundo Mendling et al. (2013, p. 162), a pesquisa em torno do *process mining* visa à descoberta de informações significantes a partir de dados granulares, de modo que a precisão pode ser alta, pois os *event logs* possuem tempos de execução registrados. Conforme Van der Aalst (2016), o *process mining* pode ser posicionado ao final do ciclo de vida do BPM, afim de analisar os dados do sistema para disponibilizar uma melhor visão do processo atual. O objetivo do *process mining* é descobrir, checar ou melhorar os processos reais através da extração de conhecimento dos dados dos sistemas, como demonstrado na Figura 5.

Figura 5 - Fluxo conceitual do *process mining*



Fonte: Van Der Aalst (2012, p.32)

Turner et al. (2012) destacam que não há modelo predefinidos na descoberta de processos, uma vez que os dados são minerados sem o uso de padrões anteriores à aplicação dos algoritmos. Van der Aalst (2016, p. 163) afirma que a descoberta de

processos é uma das tarefas mais desafiadoras do *process mining*. Os modelos de processo são construídos a partir da captura dos comportamentos verificados nos *event logs*.

Van der Aalst (2012) conceitua a monitoramento de processos como sendo a comparação de um modelo existente com o resultado da extração de um outro modelo dos *event logs*. Tal abordagem pode ser utilizada para verificar se a realidade, como registrada no *event log*, está em conformidade com o modelo existente.

A melhoria dos processos é abordada pelo *process mining* através das melhorias dos modelos. Enquanto a monitoramento analisa o alinhamento entre um modelo e a realidade, a melhoria visa às possíveis mudanças ou extensões que podem ser feitas em um modelo existente (VAN DER AALST, 2012).

Além dos objetivos, o *process mining* também pode ser classificado em relação às perspectivas analisadas, as quais servem para guiar o foco com o qual a análise será feita (VAN DER AALST, 2016). O Quadro 7 apresenta as principais perspectivas identificadas na revisão da literatura.

Quadro 7 – Perspectivas dos processos

Perspectiva	**Descrição**
Controle de fluxos	Utiliza o *event log* para modelar os fluxos da informação dos processos
Organizacional	Evidencia as relações entre os diferentes recursos envolvidos nos processos
Performance	Aplica o *process mining* para gerar análises de performance a partir dos *event logs*
Dados	Geração de conhecimento quantitativo do processo com base nos dados e evidências do mesmo
Tempo	Analise os processos sob o ponto de vista da análise temporal

Fonte: Elaborado pelo autor com base em Van der Aalst (2016).

Conforme Dumas e Maggi (2014), o *process mining* pode ainda ser classificado quanto aos tipos de abordagem, os quais podem ser subdivididos de acordo com o Quadro 8. As técnicas descritivas suportam a modelagem dos processos atuais e as

preditivas visam à predição de comportamentos futuros sob certas hipóteses ou condições.

Quadro 8 – Tipos de abordagens do *process mining*

Tipo de Abordagem	**Objetivo**	**Classificação**	**Descrição**	**Aplicações**
Descritiva	Descoberta e análise de desvios	*Process Performance Analytics*	Geração de visão da performance quantitativa do processo a partir de um *event log*	NORAMBUENA e ZEPEDA (2017)
	Descoberta e análise de desvios	*Automated Process (model) Discovery*	Descoberta de processo através de modelagens gráficas	VAN DER AALST; WEIJTERS e MARUSTER (2004)
	Análise de desvios e Remodelagem	*Model Enhancement*	Geração de novo modelo melhorado de processo a partir da adição e remoção de eventos ou atributos	VAN DER AALST; LA ROSA e SANTORO, (2016)
	Descoberta, análise de desvios e remodelagem	*Deviance Mining*	Diagnostica desvios em relação à regras predefinidas ou descobre as mesmas	MANS et al. (2011)
	Descoberta e análise de desvios	*Process Variant and Outlier Detection*	Descobre subconjuntos de variações esperadas do processo ou evidencia outliers suficientemente significativos	MANS et al. (2011)

Tipo de Abordagem	Objetivo	Classificação	Descrição	Aplicações
Preditiva	Análise de desvios	*Predictive Deviance Monitoring*	Predição em tempo real de futuros desvios nos processos e recomendação acerca da execução dos casos	VAN DER AALST; SCHONENBERG e SONG, (2011)
	Remodelagem	*Data-driven process simulation*	Simulações que representam a realidade com precisão e que podem determiner potenciais impactos de mudanças	DEN HERTOG (2008)

Fonte: Elaborado pelo autor com base em Dumas e Maggi (2015).

Posteriormente às classificações relacionadas ao tipo de abordagem, objetivo e perspectiva de análise, o *process mining* precisa que um algoritmo ou heurística seja suficientemente representativo no que tange a aderência do modelo ao *event log*.

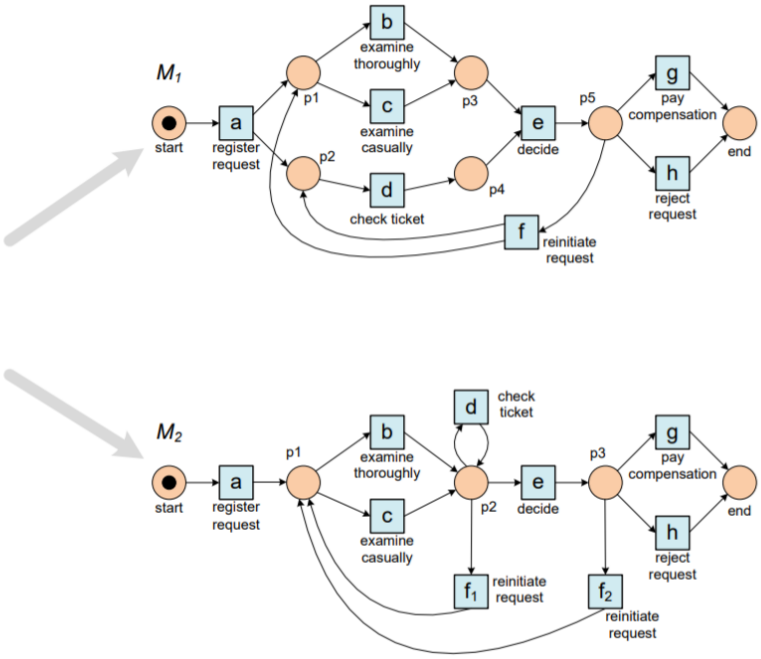
2.4.1 Descoberta de processos

A descoberta de processos endereçada pelo *process mining* visa à construção de um modelo que capture o comportamento de um *event log* de maneira representativa (DUMAS et al., 2013, p. 360). De acordo com Van der Aalst (2016, p. 163) a descoberta de processos pode ser definida através da abordagem de descoberta geral de processo (Figura 6) ou descoberta específica de processo:

a) Descoberta geral de processo: mapeamento de um *event log* afim de modelar o comportamento dos eventos representativamente. O desafio dessa abordagem é encontrar o melhor algoritmo e a vantagem é que os dados de entrada podem possuir muitos atributos;
b) Descoberta específica de processo: mapeamento de um evento específico do *event log* de modo que o modelo do processo já seja definido.

Figura 6 – Representação da descoberta geral de processo

#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg



Fonte: Van der Aalst (2012)

Conforme Van der Aalst (2016, p. 164), os processos também podem ser definidos a partir de uma equação $L = [(a,b,c,d)^3, (a,c,b,d)^2, (a,e,d)]$, na qual *L* representa o processo, *a,b,c,d,e* representam os eventos e os conjuntos dos eventos representam os casos e suas determinadas repetições.

Considerando-se a variedade de algoritmos disponíveis para a aplicação das técnicas do *process mining*, há cinco boas práticas que devem ser seguidas para se evitarem infidelidades do modelo (DUMAS et al., 2013, p. 360; VAN DER AALST, 2012):

a) Ordem dos eventos: os eventos precisam estar em ordem cronológica;

b) Referência dos casos: cada evento precisa se referir a um único caso;

c) Referência das atividades: cada evento é relacionado a uma atividade específica do processo;

d) Estado das atividades: as atividades do processo já precisam estar completadas;

e) Estado comportamental: se uma atividade *a* puder ser seguida por uma atividade *b* e essa sequência formar o caso *ab*, o *event log* é comportamentalmente completo.

Segundo Van der Aalst (2012), o fato da descoberta de processos permitir modelagens apenas baseadas nos eventos dos sistemas a torna a técnica de *process mining* mais proeminente. O valor para as empresas é destacado pela possibilidade de analisar as atividades da maneira com que elas realmente são realizadas.

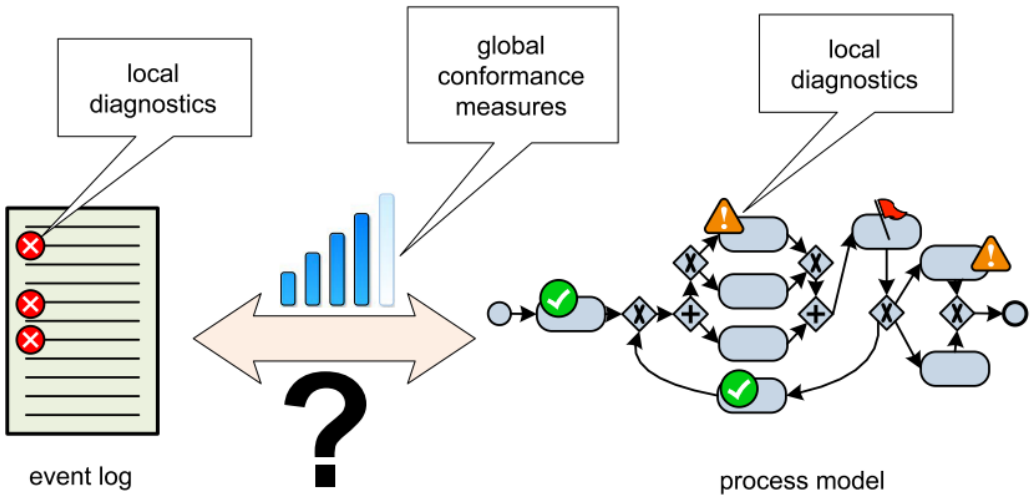
A próxima seção apresentará os conceitos relacionados ao monitoramento de processos.

2.4.2 Análise de desvios de processos

A análise de desvios é definida sob o *process mining* como sendo a revisão para verificar se as atividades são realmente executadas de acordo com o modelo ou regras pré-estabelecidas (DUMAS et al., 2013, p. 373). Van der Aalst (2012) complementa o conceito afirmando que, em tal análise, o modelo do processo existente é comparado ao *event log* de modo a verificar se a realidade não viola as diretrizes previamente definidas.

Segundo Van der Aalst (2012), diferentemente da descoberta de processos que é realizada sem detalhamentos prévios, a análise de desvios requer o *event log* e um modelo como entradas, de modo a compará-los. A Figura 7 explicita a comparação entre as entradas através de medidas de aderência que serão apresentadas na sessão 2.4.4.

Figura 7 – Representação da análise de desvios do processo



Fonte: Van der Aalst (2016, p. 244)

Van der Aalst (2016, p. 244) destaca a importância do monitoramento de processos sob os pontos de vista dos modelos errados (“Como melhorar o modelo?”)

e dos desvios nas atividades (“Como melhorar a aderência ao modelo?”). Nesse contexto, o autor classifica o monitoramento em dois propósitos:

a) Descritivo: discrepâncias entre o modelo e o *event log* indicam que o modelo precisa ser revisto de modo a representar melhor a realidade;
b) Normativo: discrepâncias entre as entradas incidam que o processo precisa ser melhor controlado ou que ações permitidas não foram inicialmente modeladas.

O monitoramento de caráter descritivo é relacionado ao estudo do alinhamento de negócios, no qual o objetivo é garantir que os sistemas de informação estejam alinhados às necessidades das pessoas. Exemplos apresentados por Van der Aalst (2016, p. 245) incluem tanto a compra de softwares que não se adequam à realidade das empresas, quanto a relação sequencial entre mudanças de processos e atualizações nos softwares, visto que os sistemas costumam ser atualizados após a implementação das melhorias.

Conforme Van der Aalst (2016, p. 245), as auditorias costumam ser alinhadas ao estudo do caráter normativo do monitoramento de processos. Verificações de validade e confiança nos sistemas de informação permitem às empresas garantir que as atividades são executadas de acordo com as diretrizes definidas pela gestão.

Na próxima sessão, serão definidos os conceitos acerca da remodelagem de processos.

2.4.3 Remodelagem de processos

A abordagem do monitoramento pode preceder as revisões dos processos, que podem ser melhorados para garantir o alinhamento entre as atividades previstas nos modelos e as executadas na realidade, ou estendidos para ser mais representativos em relação às perspectivas (VAN DER AALST, 2012).

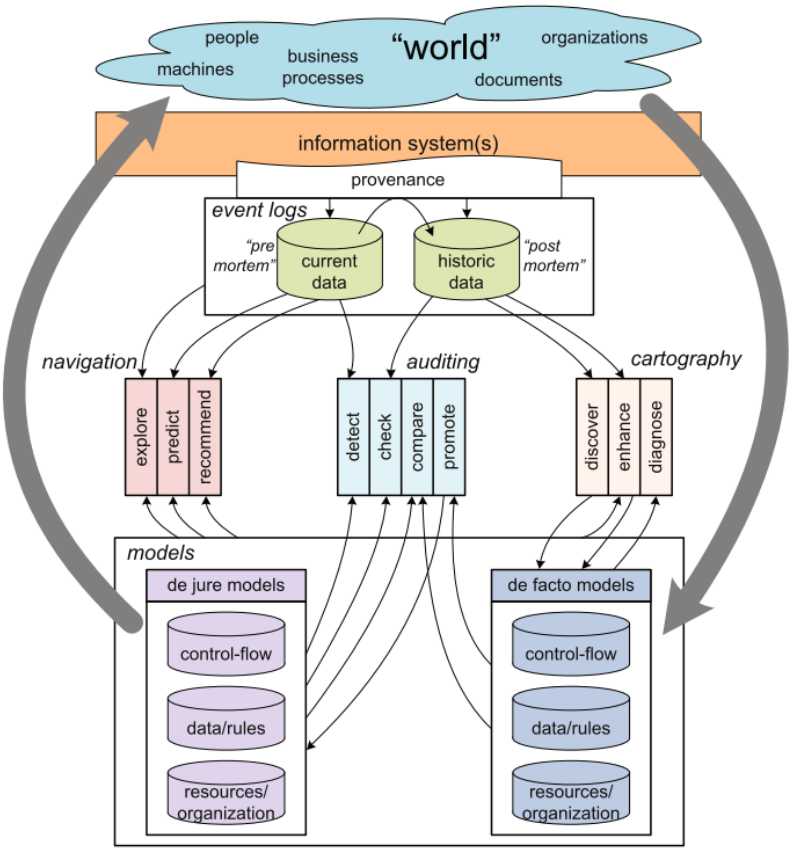
Sendo assim, o remodelagem dos processos é possível através da extensão de um modelo existente utilizando o *event log* (VAN DER AALST, 2012). Tais modificações podem contribuir com perspectivas como tempos, recursos, custos, clientes ou prioridades de atendimento. Nesse contexto, estudos recentes indicam a emersão do termo “Suporte Operacional”, de modo que as técnicas de *process mining* são utilizadas para realizar análises em tempo real afim de descrever, predizer ou recomendar futuras atividades com base nos eventos anteriores (MENDLING et al.,

2017; NORAMBUENA; ZEPEDA, 2017; VAN DER AALST, 2016; VAN DER AALST; LA ROSA; SANTORO, 2016; VAN DER AALST; SCHONENBERG; SONG, 2011).

A abundância de dados de eventos e o desenvolvimento da indústria 4.0 modificam a demanda pelo BPM rapidamente, evidenciando a necessidade de melhorias automações de processo em tempo real como as propostas pelo suporte operacional (VAN DER AALST, 2018). Pode-se afirmar que a importância da análise dos eventos cresce em um contexto que integra sistemas, sensores, redes e serviços não somente dentro das companhias, como entre elas (HERMANN; PENTEK; OTTO, 2016).

O suporte operacional redefine a remodelagem de processos uma vez que incorpora informações externas como características do negócio, quantidade de pessoas, regras da organização ou tempos de processamento (VAN DER AALST, 2016, p. 301). Dessa maneira, a Figura 8 apresenta uma proposição atualizada que engloba atividades emergentes sob o escopo do *process mining* e que possibilitam o suporte operacional em tempo real.

Figura 8 – Redefinição do *process mining*



Fonte: Van der Aalst (2016, p. 302)

3 METODOLOGIA

Esse capítulo expõe a metodologia utilizada no desenvolvimento do estudo. Serão apresentados o delineamento, o método de pesquisa, método de trabalho, as técnicas de coleta e tratamento de dados, além das etapas de aplicação do *process mining* de modo a assegurar o rigor do presente trabalho.

3.1 Delineamento da pesquisa

Uma pesquisa é composta por ações propostas para solucionar um problema através de procedimentos racionais e sistemáticos (SILVA; MENEZES, 2005). De acordo com Dresch, Lacerda e Júnior (2015, p. 15), tais procedimentos dependem da motivação pela qual a pesquisa está sendo realizada.

Quanto à classificação desses procedimentos, pode-se afirmar que não há consenso, visto que há diversas abordagens de classificação (MIGUEL, 2007; SILVA; MENEZES, 2005). Nesse contexto, o Quadro 9 apresenta e descreve as classificações encontradas nos estudos de Dresch, Lacerda e Júnior (2015), Gil (1991) e Silva e Menezes (2005).

Quadro 9 – Classificação do estudo

Critério	**Subclassificação aplicável**	**Descrição**
Objetivo	Exploratória	O objetivo é tornar os problemas mais explícitos ou construir hipóteses acerca dos mesmos
Método científico	Indutiva	Baseia-se na observação da realidade para a descoberta de conjunturas que podem contribuir para soluções ou novas teorias em relação a um problema prático
Natureza	Aplicada	Geração de conhecimento dirigidos à solução de problemas práticos
Abordagem	Quantitativa	Análise baseada em informações quantificáveis
Procedimento técnico	Estudo de caso	Aprofundamento em poucos objetos de estudo com o objetivo de obter conhecimentos detalhados acerca dos mesmos

Fonte: Elaborado pelo autor com base em Dresch, Lacerda e Júnior (2015), Gil (1991) e Silva e Menezes (2005).

Considerando-se as classificações descritas, o presente estudo é caracterizado como exploratório, pois objetiva ajudar na explicação do problema de pesquisa e contribuir para a geração de hipóteses. Sendo assim, o problema explorado parte do fato de que, em períodos de *release*, o atendimento ao SLA é impactado. As hipóteses futuras poderão ser formuladas com base na modelagem obtida e nos resultados da análise preditiva dos tempos de processamento.

Quanto ao método científico, a pesquisa é tratada como indutiva em decorrência do seu objeto ser uma situação prática e por ser desenvolvida com o intuito de observar as variáveis do processo que impactam no tempo de processamento dos atendimentos. Alinhada ao método científico, o estudo é aplicado pois objetiva a geração de conhecimentos dirigidos à solução de problemas práticos.

A abordagem do estudo pode ser definida como qualitativa e quantitativa, uma vez que o modelo do processo atual será analisado de maneira subjetiva para futuras proposições, mas os tempos de processamento e a qualidade dos algoritmos de predição serão calculados quantitativamente.

O procedimento técnico do estudo será o estudo de caso, o qual é definido por Eisenardt (1989) como sendo a estratégia de pesquisa que foca no entendimento da dinâmica presente um determinado cenário. Yin (1994) completa afirmando que o estudo de caso permite a realização de pesquisas sem alterar características dos projetos. Dessa forma, os dados serão extraídos do sistema que registra as atividades do processo existente e as análises serão feitas sem interferências no objeto de estudo.

Na próxima seção, o procedimento técnico será detalhado enquanto método de pesquisa do presente estudo.

3.2 Método de pesquisa

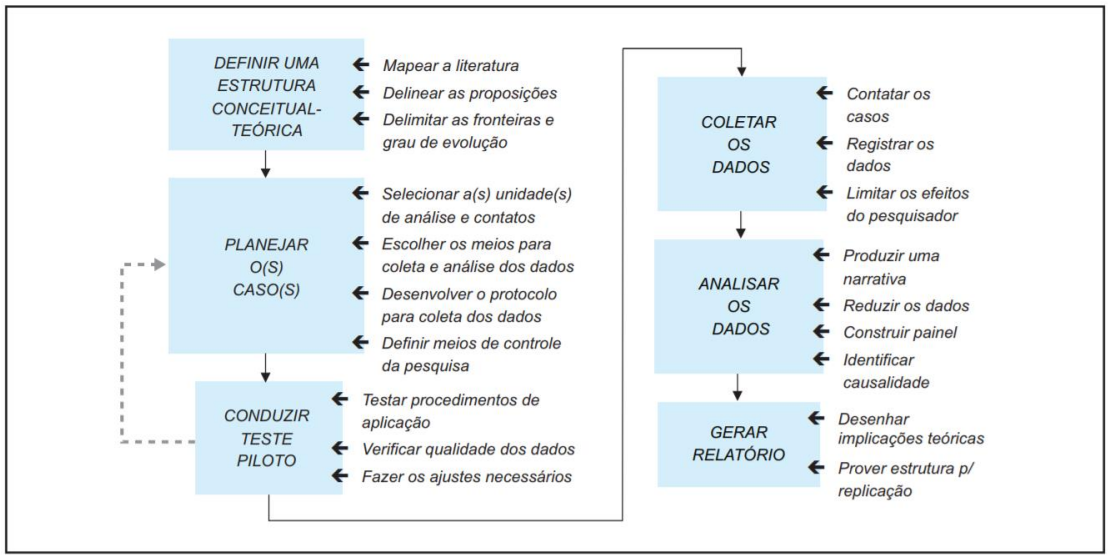
Segundo Dresch, Lacerda e Júnior (2015, p. 20), a importância de delinear e justificar o método de pesquisa é garantir que os pesquisadores realmente respondam ao problema de pesquisa. Dessa forma, desenvolver trabalhos utilizando a metodologia correta é importante não apenas pela necessidade de embasamento científico adequado, mas pela busca da melhor abordagem para definir as questões e técnicas de pesquisa que guiem o seu planejamento e condução (MIGUEL, 2007).

Considerando-se o método científico indutivo e os objetivos do estudo em questão, o estudo de caso será o método de pesquisa utilizado, pois o processo será modelado e analisado quantitativamente sem alterações, afim de criar hipóteses acerca do impacto de determinadas variáveis sobre o tempo de processamento. Esse objetivo vai de encontro ao conceito de estudo de caso proposto por Gil (1991, p. 54), o qual diz que os resultados são apresentados sob a forma de hipóteses, não de conclusões.

Um estudo de caso investiga um fenômeno dentro de um contexto real, no qual os limites entre ambos não é claramente definido. Yin (1994) classifica os estudos de caso em dois principais grupos: (I) Casos únicos, nos quais as hipóteses são geradas em apenas um contexto. (II) Casos múltiplos, os quais englobam dois ou mais contextos reais para a criação das hipóteses (MIGUEL, 2007).

Miguel (2007) ainda classifica as decisões metodológicas de acordo com os níveis de abrangência e profundidade com as quais elas foram tomadas: (I) Nível estratégico, no qual a abordagem metodológica é definida de acordo com fatores como o objetivo e o problema de pesquisa. (II) Nível operacional, no qual é definido o planejamento para a condução da pesquisa, como apresentado na Figura 9.

Figura 9 – Planejamento para condução de estudo de caso



Fonte: Miguel (2007)

A estrutura conceitual do estudo é composta pela revisão sistemática da literatura apresentada no item 1.3 e pelos conceitos das seções 2.3 e 2.4. Foi evidenciada uma emergente necessidade de aplicação de técnicas de *process mining* sob o escopo do BPM, mas o foco dos pesquisadores é, predominantemente, no

desenvolvimento de algoritmos precisos. Dessa forma, surge a oportunidade de aliar tais técnicas às análises preditivas como as conduzidas nesse estudo.

No que tange o planejamento do caso, a seleção da unidade de análise foi definida no item 1. Além disso, os desdobramentos acerca da coleta e do tratamento de dados serãp descritos nos itens 3.4 e 3.5, respectivamente.

A análise dos dados ocorrerá de acordo com as diferentes técnicas de *process mining* e será descrita no item 3.6. Adicionalmente, os resultados serão em modelos descritivos (seção 4.3) e modelos preditivos (seção 4.4).

As seções 5 e 6 apresentarão, respectivamente, a discussão dos resultados e as considerações finais acerca do estudo. A seção 5 analisará a presente pesquisa à luz das contribuições acadêmicas e gerenciais que poderão ser extraídas do estudo. A última seção retomará as etapas do estudo, determinará as sugestões para as pesquisas futuras e apresentará as conclusões finais do trabalho.

Na próxima sessão, será apresentado o método de trabalho no qual o estudo se baseia.

3.3 Método de trabalho

Nesta sessão, será apresentado o método de trabalho no qual o estudo é baseado. De acordo com Dresch, Lacerda e Antunes (2015), o método de trabalho define a sequência lógica que um pesquisador deve seguir para atingir os objetivos da pesquisa. Os autores ainda afirmam que um método de trabalho apropriado oferece clareza e transparência em relação ao processo de pesquisa, o que ajuda a mesma a ser validada, reconhecida e replicada por outros pesquisadores.

No método de trabalho, o pesquisador deve descrever e justificar o método de pesquisa, além das técnicas de coleta e análise de dados que serão utilizadas na execução do estudo (DRESCH; LACERDA; ANTUNES, 2015). Dessa forma, o método de trabalho do presente estudo foi elaborado baseado em Van der Aalst (2016), que sugere cinco estágios básicos para a aplicação do *process mining*:

a) Planejamento e justificativa: definição do tipo de projeto de acordo com os objetivos do estudo;
b) Extração de dados: coleta de dados dos sistemas de informação de modo que os eventos sejam subordinados aos casos e organizados por tempo;

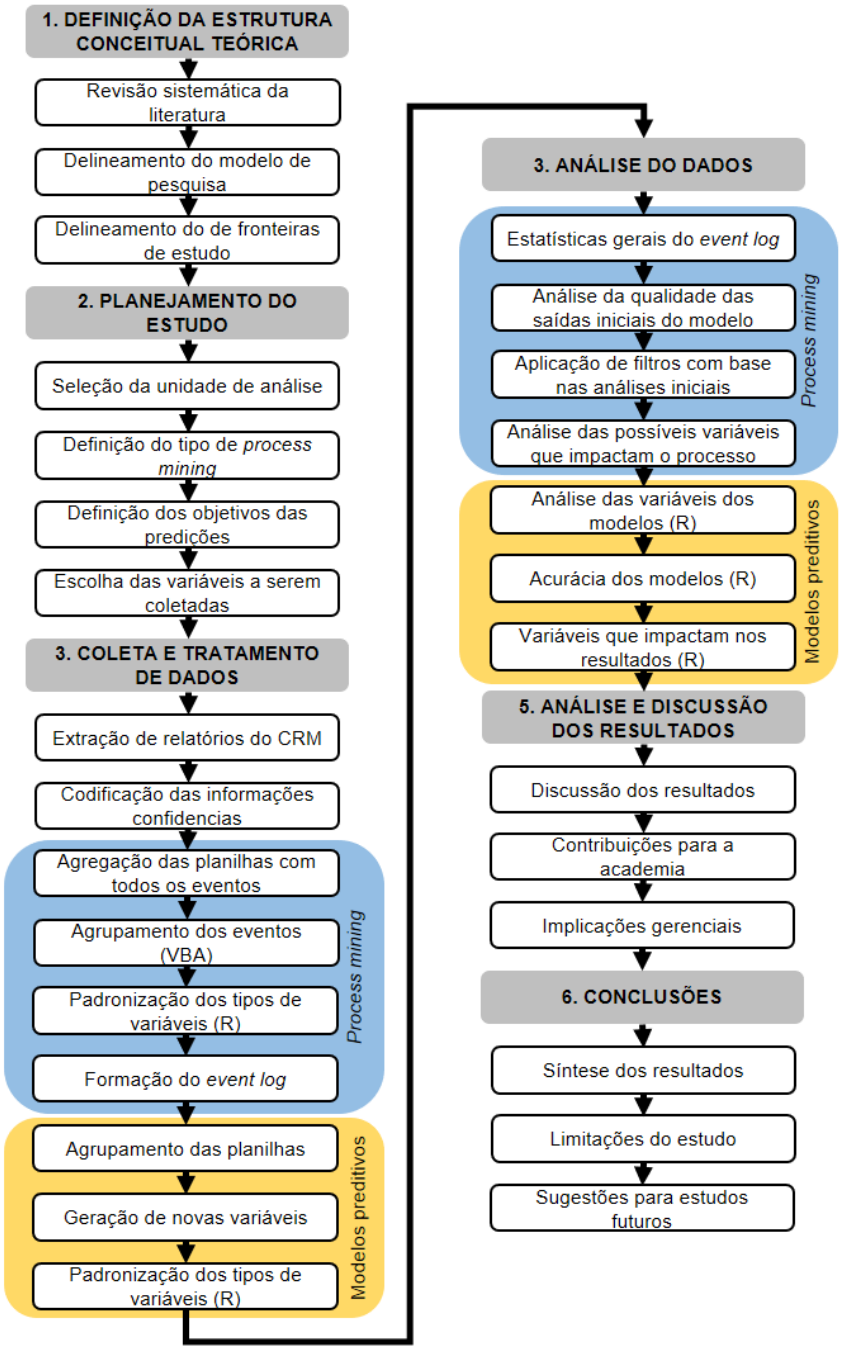
c) Criação de modelos de controle: geração de modelo de descoberta ou monitoramento de processos baseado em um fluxo já modelado ou em *event logs*;
d) Adição de perspectivas aos modelos: enriquecimento do modelo com informações organizacionais, tempos de processamento ou estruturas de decisão;
e) Suporte operacional: desenvolvimento de atividades adicionais como predições, detecções e recomendações, de modo que os resultados sejam autoexplicativos, sem a necessidade de interpretação.

O método acima exposto serve de base para a condução do estudo, mas não aborda em detalhes os procedimentos acerca das predições realizadas na pesquisa. Dessa forma, a classificação da violação do SLA e do tempo de fechamento serão conduzidas com base em Kuhn e Johnson (2013):

a) Pré-processamento dos dados: adaptação das classes das variáveis, substituição de dados inexistentes, análises de correlação e variância;
b) Estimação dos parâmetros do modelo: divisão entre dados de treinamento e teste com base no tamanho da amostra e nos tipos de variáveis;
c) Seleção de variáveis para o modelo: análises de correlação, aplicação do algoritmo *Boruta* e utilização da técnica *Recursive Feature Selection* serão os métodos de seleção de variáveis (seção 3.6.2.1);
d) Avaliar a performance do modelo: matrizes de confusão com análises da acurácia e do coeficiente *kappa* serão utilizadas, assim como os gráficos de dependência das variáveis de resposta;
e) Ajuste das regras de predição: mudanças nos parâmetros dos algoritmos com base na performance dos modelos.

Elementos adicionais do método de trabalho foram concebidos com base em Miguel (2007) e Yin (2006), os quais descrevem as etapas e descrevem um modelo para a condução de estudos de caso. Sendo assim, os cinco estágios para a aplicação do *process mining*, os passos recomendados para a aplicação de algoritmos de classificação e suas customizações pertinentes a esse estudo são adaptados ao fluxo de condução do estudo de caso (Figura 10).

Figura 10 – Método de trabalho



Fonte: Elaborado pelo autor

Na fase 1 foi realizada uma revisão sistemática da literatura baseada no *framework* apresentado no capítulo 1. Através de artigos, dissertações, teses e livros, foi possível identificar trabalhos relacionados ao *process mining* e à predição de tempos de atendimento. O delineamento e a delimitação das fronteiras da pesquisa foram definidos com base nas lacunas encontradas na literatura.

Na fase 2, a unidade de análise foi escolhida em decorrência da recorrente violação de SLA's que o processo apresenta. O planejamento do estudo de caso foi desenvolvido com base nos estágios para aplicação do *process mining,* que será a base das análises descritivas, e dos modelos preditivos. Para a análise descritiva do processo, o tipo escolhido foi a descoberta, visto que seus resultados poderiam ser entradas para a classificação dos SLA's. As variáveis a serem coletadas e a justificativa para tais serão descritas no item 3.4.

A fase 3 apresenta os procedimentos para coleta e tratamento dos dados. Os relatórios foram gerados no CRM e inicialmente tratados com funções de planilhas eletrônicas tanto para o *process mining* quanto para os modelos preditivos. Em ambos os casos, os dados foram tradados em planilhas eletrônicas e exportados para a plataforma R para serem manipulados com o auxílio das bibliotecas *tidyverse* (WICKHAM, 2014) e *BupaR* (JANSSENSWILLEN; DEPAIRE, 2017). As seções 3.4 e 3.5 detalham tais procedimentos.

Na fase 4, a técnica de descoberta do processo foi aplicada sobre o *event log* e os resultados dessa análise foram utilizados como entradas para os modelos preditivos de classificação de violação do SLA e fechamento dos *cases.* Para as predições, primeiramente foram realizadas análises de importância e correlação das variáveis, para que os dados considerados relevantes componham os modelos. Os algoritmos utilizados foram definidos na seção 3.6.2.2 e os resultados dos mesmos subsidiaram a avaliação do comportamento das variáveis frente à violação de SLA e tempo de fechamento dos *cases.*

A discussão dos resultados da pesquisa ocorreu na fase 5. Foram analisadas as contribuições acadêmicas e empresariais do estudo para que as mesmas pudessem ser sintetizadas na fase 6, na qual foram feitas considerações acerca das limitações do estudo de caso e sugeridas questões de pesquisas futuras.

Na próxima sessão, serão detalhados os procedimentos de coleta de dados para o estudo.

3.4 Coleta de dados

Pesquisas direcionadas aos negócios são compostas pela coleta de informações para melhorar a tomada de decisão. Sendo assim, a definição das técnicas de coleta de dados garante a operacionalização dos métodos de pesquisa e

de trabalho definidos anteriormente (DRESCH; LACERDA; JÚNIOR, 2015; HAIR et al., 2011).

A coleta de dados iniciou com uma reunião com a gerente do setor para definir as diretrizes de tal ação. Foi decidido que todas as variáveis que pudessem identificar algum aspecto da empresa seriam substituídas por números. Dessa forma, as variáveis “recurso”, “problema”, “editado por”, “evento”, “tipo de usuário”, “valor antigo” e “novo valor” foram substituídas por valores numéricos para fins de preservar a identidade da empresa.

Decididos os moldes nos quais os dados seriam coletados, foram selecionados, a partir dos resultados da revisão sistemática da literatura, os estudos que aliaram *process mining* à predição de tempos. Sendo assim, as variáveis utilizadas em tais pesquisas em questão são apresentadas no Quadro 10.

Quadro 10 – Levantamento das variáveis de entrada e de resposta

Estudo	**Principais variáveis de entrada**	**Variáveis de resposta**
Process Mining to forecast the future of running cases	*Timestamps*, recursos e atividade atual	Próxima atividade de um *case*
Process Mining in a manufacturing company for predictions and planning	Recursos, atividade atual e status do *case*	Tempo de processamento restante
Queue mining delay prediction in multi-class service processes	*Timestamps,* status atual do *case*, progresso do *case* e segmentação do cliente	Tempo total de processamento
Time prediction based on process mining	*Timestamp*, recursos, atividades e custos associados a cada atividade	Tempo de processamento restante

Fonte: Elaborado pelo autor com base em Pospísil et al. (2013), Pravilovic, Appice e Malerba (2014), Senderovich et al. (2015) e Van der Aalst, Schonenberg e Song (2011)

Considerando-se que o *process mining* é derivado do *data mining* e que a amplitude do volume de dados utilizados em diferentes pesquisas é grande, o *event log* considerado para a presente situação possui 41.440 eventos. O período de coleta de dados para as análises descritiva e preditiva são diferentes pois os objetivos dos modelos são distintos. Nessa pesquisa, a modelagem descritiva tem o objetivo de criar entendimento acerca do processo rotineiro com o mínimo de interferências que possam causar picos de demanda. Sendo assim, foi escolhido um período de coleta que não apresentasse nenhuma manutenção no sistema nas 2 semanas anteriores.

Quadro 11 – Períodos de coleta de dados

Tipo de análise	**Período de coleta**	**Justificativa**
Modelagem descritiva	23 a 29 de setembro	Período estável
Modelagem Preditiva	7 a 17 de março	Primeiro *release* do ano
	11 a 21 de julho	Segundo *release* do ano
	5 a 15 de setembro	Período sem *release*

Fonte: Elaborado pelo autor

O Quadro 11 apresenta a justificativa acerca de cada período de coleta de dados. Para a análise descritiva, serão utilizados os dados de uma semana para que se possa avaliar o sistema sob a ótica da sequência dos dias sem a interferência de manutenções. A predição de tempos foi realizada com dados de três períodos de modo a garantir que a variável *release* fosse considerada. Sendo assim, foram analisadas a primeira semana após os dois primeiros *releases* do ano e uma semana de setembro sem a influência de atualizações do sistema, totalizando 16.465 observações.

Quadro 12 – Variáveis coletadas

Tipo de análise	Variável a coletar	Origem dos dados	Período de coleta
Descritiva	*Case number*	CRM	23 a 29 de setembro
	Recurso		
	Data de edição		
	Data do e-mail		
	Data de comentários		
	Evento		
	Valor antigo		
	Novo valor		
	E-mail Status		
Preditiva	Períodos de *release*	Sistema de gerenciamento de ambientes	Março e Julho (2018)
	Case number	CRM	Março, Julho e Setembro (2018)
	Idioma		
	Tipo de usuário		
	Dia da semana		
	Problema		
	Abertura do *case*		
	Tempo inicial de reposta (IRT)		
	Valor antigo		
	Novo valor		
	Nacionalidade		
	Time		

Fonte: Elaborado pelo autor.

O Quadro 12 mostra que há variáveis que foram coletadas para os dois estudos em questão. Isso ocorreu pois o *case number,* o novo valor e o valor antigo são necessários para o tratamento de dados tanto na análise descritiva quanto na preditiva. Todos os dados foram extraídos do CRM com exceção dos períodos de *release*, que estão disponíveis em um sistema de gerenciamento de ambientes. As Tabelas 2, 3, 4 e 5 apresentam, respectivamente, os dados de troca de e-mails, eventos dos *cases*, tempo inicial de resposta e datas dos comentários.

Tabela 2 – Dados de troca de e-mails

Case Number	Recurso	Tipo de usuário	Dia da semana	IRT	Problema	Abertura do *case*	Email Status	Data do e-mail	Idioma
1707895	01	1	*Sunday*	09/02/2018 04:21	1	09/02/2018 00:54	Recebe	09/02/2018 00:54	*English*
1707895	01	1	*Sunday*	09/02/2018 04:21	1	09/02/2018 00:54	Envia	09/02/2018 12:27	*English*
1707895	01	1	*Sunday*	09/02/2018 04:21	1	09/02/2018 00:54	Recebe	09/03/2018 21:04	*English*
1707895	01	1	*Sunday*	09/02/2018 04:21	1	09/02/2018 00:54	Recebe	09/03/2018 19:59	*English*
1707895	01	1	*Sunday*	09/02/2018 04:21	1	09/02/2018 00:54	Envia	09/02/2018 04:21	*English*
1707895	01	1	*Sunday*	09/02/2018 04:21	1	09/02/2018 00:54	Envia	09/02/2018 00:54	*English*

Fonte: Elaborado pelo autor.

A Tabela 2 apresenta os dados de troca de e-mails coletados. A coluna “Email status” representa o sentido da comunicação, de modo que quando os atendentes recebem um e-mail, um evento é criado. Para fins dessa pesquisa, tal evento assume o valor “Recebe”, assim como o contrário é representado por “Envia”.

Tabela 3 – *Logs* do CRM

Case Number	**Data da edição**	**Evento**	**Valor antigo**	**Novo valor**
1707895	09/02/2018 00:54	01	01	03
1707895	09/02/2018 00:54	02	01	02
1707895	09/02/2018 00:54	03	02	03
1707895	09/02/2018 00:54	02	01	04
1707895	09/02/2018 02:40	01	03	05
1707895	09/02/2018 02:40	04	06	02

Fonte: Elaborado pelo autor.

A Tabela 3 apresenta os *logs* extraídos do CRM. Nesse relatório, as colunas “Valor antigo” e “Novo valor” representam a mudança de estado de um determinado evento. Para fins de interpretação, o primeiro registro evidencia uma mudança no *case* 1707895, o qual teve o valor antigo do Evento 01 substituído pelo novo valor 03.

Tabela 4 – Tempo até a primeira resposta

Case Number	**IRT (mins)**
1707895	207.00
1710809	12.58
1707895	207.00
1710809	12.58
1707895	207.00
1710810	6.08

Fonte: Elaborado pelo autor

A Tabela 4 apresenta os tempos até a primeira resposta dos *cases* analisados. Tais informações serão utilizadas para compor a variável de violação do SLA. Dito isso, a manipulação e utilização desses dados serão detalhados na seção 3.5.2

Tabela 5 - Comentários

Case Number	**Data do comentário**
1707895	09/02/2018 00:55
1707895	09/02/2018 00:58
1707896	09/02/2018 03:06
1707897	09/02/2018 03:18
1707897	09/02/2018 03:39
1707898	09/02/2018 03:44

Fonte: Elaborado pelo autor.

Finalmente, a Tabela 5 apresenta a data e o horário dos comentários que foram inseridos nos *cases*. Dada a desagregação dos dados, evidenciou-se a necessidade de processamento prévio para possibilitar a interpretação das tabelas por parte dos algoritmos. Dessa maneira, o tratamento dos dados será descrito na próxima seção.

3.5 Tratamento de dados

O tratamento de dados foi necessário em decorrência dos diferentes formatos que as variáveis aparecem nos relatórios do CRM. Foram utilizadas planilhas eletrônicas e a plataforma R, de modo a filtrar, substituir e compor as tabelas que serviram de entradas para os modelos preditivos e descritivos. Além do ajuste nos dados existentes, os relatórios foram manipulados para que novas variáveis fossem criadas.

3.5.1 Modelos descritivos

O *process mining* deve ser aplicado sobre *event logs*. Dessa maneira, as Tabelas 2, 3 e 5 foram processadas em planilhas eletrônicas para que os comentários e e-mails pudessem ser incorporados às atividades. As variáveis data de edição, data do comentário e data do e-mail foram incorporadas à nova coluna *timestamp* com o auxílio de funções condicionais de planilhas eletrônicas. A Tabela 6 representa a agregação descrita.

Tabela 6 – Tratamento de dados I

Case Number	**timestamp**	**Evento**	**Valor antigo**	**Novo valor**
1707895	09/02/2018 00:54	01	01	01
1707895	09/02/2018 00:54	02	01	02
1707895	09/02/2018 00:54	03	03	03
1707895	09/02/2018 00:54	02	01	04
1707895	09/02/2018 02:40	01	03	05
1707895	09/02/2018 02:40	04	06	02

Fonte: Elaborado pelo autor.

Todas as atividades analisadas agora compõem a planilha que foi convertida em *event log*. Porém, muitas ações representam interações pontuais e comuns à todos os *cases*, como a entrada dos dados do cliente, a escolha do problema e do tipo de usuário. Além disso, há situações em que diferentes sequências de atividades são realizadas para alcançar o mesmo objetivo. Dessa maneira, o APÊNDICE A apresenta o algoritmo utilizado para agregar tais dados.

Tabela 7 – Tratamento de dados II

case_number	***history_id***	**status**	***resource***	***timestamp***	**Evento**	**Valor Antigo**	**Novo Valor**	***adjustment***
1723990	1	Closed	1	9/23/2018 12:09 AM	Created.			create
1723990	4	Closed	1	9/23/2018 12:09 AM	Case Owner	Customer Support	queued	
1723990	5	Closed	1	9/23/2018 12:09 AM	Email	NA	send_email	send_email
1723990	8	Closed	1	9/23/2018 12:11 AM	Case Owner	Customer Support	04	grab
1723990	9	Closed	1	9/23/2018 12:12 AM	Contact Name	01	edit	
1723990	10	Closed	1	9/23/2018 12:12 AM	Issue		Logon	edit
1723990	11	Closed	1	9/23/2018 12:12 AM	First Call Resolution?	Yes	edit	

Fonte: Elaborado pelo autor.

A Tabela 7 apresenta os dados que foram utilizados para a composição do *event log*. As colunas *case_number*, *status* e *resource* foram renomeadas para se adequarem à terminologia do *process mining*. As variáveis *history_id* e *adjustment* foram criadas para, respectivamente, identificar cada evento e sua atividade subordinada. Aplicados os filtros e feitas todas as substituições pertinentes à proteção dos dados confidenciais, as colunas que não serão utilizadas foram excluídas.

Tabela 8 – Tratamento de dados III

case_number	***history_id***	***status***	***resource***	***timestamp***	***adjustment***
1723990	1	Closed	1	9/23/2018 12:09 AM	*create*
1723990	4	Closed	1	9/23/2018 12:09 AM	*queued*
1723990	5	Closed	1	9/23/2018 12:09 AM	*send_email*
1723990	8	Closed	1	9/23/2018 12:11 AM	*grab*
1723990	9	Closed	1	9/23/2018 12:12 AM	*edit*
1723990	10	Closed	1	9/23/2018 12:12 AM	*edit*
1723990	11	Closed	1	9/23/2018 12:12 AM	*edit*

Fonte: Elaborado pelo autor.

A Tabela 8 representa os dados que foram convertidos em um arquivo csv e utilizados como entrada para a plataforma R. Considerando-se que o *event log* é construído com base nos tipos de variáveis, a Tabela 9 apresenta a estrutura inicial dos dados.

Tabela 9 – Estrutura inicial dos dados

case_number	***history_id***	***status***	***resource***	***timestamp***	***adjustment***
integer	*integer*	*character*	*integer*	*character*	*character*
1723990	1	"Closed"	1	"9/23/2018 12:09 AM"	*"create"*
1723990	4	"Closed"	1	"9/23/2018 12:09 AM"	*"queued"*
1723990	5	"Closed"	1	"9/23/2018 12:09 AM"	*"send_email"*
1723990	8	"Closed"	1	"9/23/2018 12:11 AM"	*"grab"*
1723990	9	"Closed"	1	"9/23/2018 12:11 AM"	*"edit"*
1723990	10	"Closed"	1	"9/23/2018 12:15 AM"	*"edit"*
1723990	11	"Closed"	1	"9/23/2018 12:15 AM"	*"send_email"*

Fonte: Elaborado pelo autor.

A Tabela 9 apresenta a estrutura inicial que os dados tinham quando foram carregados na plataforma R. As colunas *case_number*, *history_id* e *resource* são do tipo *integer*, e precisaram ser transformadas em *factors*. A variável *status* também foi convertida em *factor* e os *timestamps* foram ajustados com o auxílio do pacote *lubridate*. Os dados contidos na coluna *adjustment* foram mantidos como *character*. Para que o *event log* possa ser definido, faltam ainda o ajuste do comprimento dos caracteres, a substituição de “-“ por “_” e a ordenação dos eventos por *timestamp* e *case_number*, respectivamente.

Tabela 10 – Estrutura final do *event log*

case_number	***history_id***	***status***	***resource***	***timestamp***	***adjustment***	***.order***
character	*character*	*factor*	*factor*	*POSIXct*	*factor*	*integer*
1723990	1	"Closed"	1	2018-09-23 00:09:00	*"create"*	1
1723990	4	"Closed"	1	2018-09-23 00:09:00	*"queued"*	2
1723990	5	"Closed"	1	2018-09-23 00:09:00	*"send_email"*	3
1723990	8	"Closed"	1	2018-09-23 00:11:00	*"grab"*	4
1723990	9	"Closed"	1	2018-09-23 00:11:00	*"edit"*	5
1723990	10	"Closed"	1	2018-09-23 00:15:00	*"edit"*	6
1723990	11	"Closed"	1	2018-09-23 00:15:00	*"send_email"*	7

Fonte: Elaborado pelo autor.

A Tabela 10 representa o *event log* que serviu de entrada para o modelo de *process mining*. O pacote utilizado para a composição de tais dados foi o BupaR e a coluna *.order* é automaticamente criada pela função *eventlog().* Os tipos de variáveis foram automaticamente definidos, com destaque para a coluna *timestamp*, que foi convertida no formato *POSIXct*, o qual oi resultado da aplicação do pacote *lubridate*.

Cumpridas as etapas de tratamento de dados para o *process mining,* resultados puderam ser extraídos e o algoritmo utilizado para a construção do *event log* final está apresentado no APÊNDICE B. A próxima seção descreve o pré-processamento dos dados para os modelos preditivos.

3.5.2 Modelos preditivos

O tratamento dos dados de entrada para as predições de violação do SLA e tempo de fechamento dos *cases* foi composto pela extração de novas variáveis a partir das planilhas iniciais. Os dados a serem manipulados satisfizeram as diretrizes do Quadro 9. Sendo assim, o resultado de tal pré-processamento foi um arquivo csv com 18 colunas numéricas e 16.465 observações.

Assim como nos modelos descritivos, a primeira ação necessária foi a codificação de todas as informações confidenciais. Além disso, há modelos preditivos de classificação que reconhecem apenas variáveis numéricas, então todas as colunas foram convertidas em números e os dados começaram a ser processados a partir da Tabela 7.

Considerando-se que todas as planilhas coletadas estavam sob a forma de eventos, diversas funções condicionais e de procura de valores foram aplicadas sobre a Tabela 7, de modo a extrair todas as variáveis de interesse que se mostraram potencialmente relevantes na etapa de modelagem descritiva. Dessa forma, o Quadro 13 apresenta detalhes sobre as colunas que foram geradas para os modelos preditivos.

Quadro 13 – Composição das variáveis dos modelos preditivos

Variável	**Justificativa**	**Método**
IRT	Será a variável de resposta binária para indicar violação do SLA	Função condicional (*if*) em planilha eletrônica a partir da tabela 4

Variável	Justificativa	Método
Nacionalidade	Composição do recurso em decorrência do tamanho do vetor suportado na plataforma R	Funções condicionais (*if*) e procura de valores (*vlookup*) a partir das colunas Valor Antigo e Novo Valor da tabela 7
Time		
Recurso por dia	Introduzir a variável de capacidade do sistema	Contagens condicionais (*count.if*) e funções de procura de valores.
Volume Anterior	O volume de *cases* criados no dia anterior pode afetar a performance do sistema (seção 4.3)	
Volume por hora	O volume de *cases* criados na última hora pode afetar o sistema (seção 4.3)	Procura de valores e contagens condicionais a partir da coluna Evento da tabela 7
Release	Mudanças no sistema podem acarretar erros e aumentar a demanda	Funções condicionais com base nas datas de criação dos *cases*
Created	Variável auxiliar para compor o *work in progress* (wip)	Funções condicionais e de contagem com base nas colunas Novo Valor e Valor Antigo da tabela 7.
Reopened		
wip	Representar a demanda das últimas 96 horas em decorrência do volume poder afetar a performance do sistema (seção 4.3)	*Created+Reopened-Closed*
Hora	O horário de abertura dos *cases* pode afetar o sistema (seção 4.3)	Funções de data e hora aplicadas sobre a coluna *timestamp* da tabela 7
Atravessamento	Tempo de atravessamento para composição da variável de resposta	Aplicação da função *throughput_time()* na plataforma R
Atravessamento maior que um dia	Variável de resposta com definição de ponto de corte arbitrário	Funções condicionais de planilhas eletrônicas

Fonte: Elaborado pelo autor.

O Quadro 13 apresenta as variáveis que foram derivadas da Tabela 7. Pode-se concluir que grande parte do processamento foi realizado com funções de planilhas eletrônicas e que algumas saídas dos modelos descritivos (seção 4.3) serviram de entradas para os modelos preditivos. Tendo os textos já convertidos em números em decorrência da confidencialidade das informações, os eventos da Tabela 2 foram agregados em *cases* únicos.

Tabela 11 – Dados iniciais com *cases* únicos

Case Number	Recurso	Tipo de usuário	Dia da semana	IRT	Problema	Abertura do *case*	Email Status	Data do e-mail	Idioma
1707895	01	1	*Sunday*	09/02/2018 04:21	1	09/02/2018 00:54	Recebe	09/02/2018 00:54	01
1707896	02	1	*Sunday*	09/02/2018 02:34	2	09/02/2018 01:23	Recebe	09/02/2018 01:23	01
1707897	01	1	*Sunday*	09/02/2018 03:40	1	09/02/2018 01:50	Recebe	09/03/2018 01:50	01
1707898	03	2	*Sunday*	09/02/2018 03:01	3	09/02/2018 02:04	Recebe	09/03/2018 02:04	01
1707899	04	1	*Sunday*	09/02/2018 03:28	4	09/02/2018 02:26	Recebe	09/02/2018 02:26	01
1707900	04	1	*Sunday*	09/02/2018 04:37	1	09/02/2018 02:41	Recebe	09/02/2018 02:41	01

Fonte: Elaborado pelo autor.

A Tabela 11 apresenta a conversão dos eventos da Tabela 2 em observações únicas. Isso possibilitou a agregação de informações acerca da característica dos *cases*, para que os mesmos fossem analisados pelos algoritmos. Assim, como todas as outras colunas, as variáveis Idioma e Dia da semana foram convertida em números para que diferentes abordagens pudessem ser testadas.

Tabela 12 – Variáveis existentes na coleta de dados inicial

Case Number	**Tipo de usuário**	**Dia da semana**	**IRT (mins)**	**Problema**	***Release***	**Idioma**
1707895	1	1	207	1	0	01
1707896	1	1	71	2	0	01
1707897	1	1	110	1	0	01
1707898	2	1	57	3	0	01
1707899	1	1	62	4	0	01
1707900	1	1	116	1	0	01

Fonte: Elaborado pelo autor.

A Tabela 12 apresenta as variáveis de entrada dos modelos preditivos, com exceção do *case number,* que já constavam na coleta inicial de dados. A ocorrência ou não de *release* e o tempo da primeira resposta (IRT) foram agregados de forma que o primeiro é uma variável binária e o segundo é expresso em minutos. Como detalhado na seção 4.3, a demanda varia não apenas conforme o dia da semana, mas também de acordo com a hora do dia. Sendo assim, a Tabela 12 não abrange os resultados dos modelos descritivos.

Tabela 13 – Adição de variáveis derivadas dos dados iniciais

Case Number	Tipo de usuário	Dia da semana	Volulme anterior	*Created*	*Reopened*	Recursos por dia	Volume por hora	Hora	Violou o SLA?	IRT (mins)	Problema	*Release*	Idioma
1707895	1	1	111	2821	474	18	0	0	1	207	1	0	01
1707896	1	1	111	2822	475	18	1	1	0	71	2	0	01
1707897	1	1	111	2823	476	18	2	1	0	110	1	0	01
1707898	2	1	111	2824	476	18	3	2	0	57	3	0	01
1707899	1	1	111	2825	477	18	4	2	0	62	4	0	01
1707900	1	1	111	2826	477	18	5	2	0	116	1	0	01

Fonte: Elaborado pelo autor.

Tabela 14 – Representação de todas as variáveis

Case Number	Time	Nacionalidade	Tipo de usuário	Dia da semana	Volulme anterior	*Created*	*Reopened*	wip	Recursos por dia	Volume por hora	Hora	Violou o SLA?	IRT (mins)	Problema	*Release*	Idioma	*Lead time* (mins)	*Lead time* > 1 dia
1707895	1	1	1	1	111	2821	474	501	18	0	0	1	207	1	0	01	291	0
1707896	1	1	1	1	111	2822	475	502	18	1	1	0	71	2	0	01	160	0
1707897	1	1	1	1	111	2823	476	503	18	2	1	0	110	1	0	01	118	0
1707898	1	1	2	1	111	2824	476	500	18	3	2	0	57	3	0	01	64	0
1707899	1	1	1	1	111	2825	477	501	18	4	2	0	62	4	0	01	98	0
1707900	1	1	1	1	111	2826	477	500	18	5	2	0	116	1	0	01	154	0

Fonte: Elaborado pelo autor

A Tabela 13 apresenta os dados adicionais que foram extraídos com funções de planilhas eletrônicas. As variáveis “Volume anterior” e “Volume por hora” representam, respectivamente, a quantidade de *cases* criados no dia anterior e no mesmo dia até determinada hora de uma observação. Da mesma forma, “*Created”* e “*Reopened”* foram criadas com funções que, a partir da Tabela 7, contaram os *cases* dos últimos quatro dias (ver Gráfico 4). As colunas “Recurso por dia”, “Violou o SLA?” e “Hora” significam, respectivamente, a quantidade de atendentes trabalhando na fila em determinado dia, se a primeira resposta de um *case* demorou mais de 120 minutos e qual a hora do dia em que ele foi criado.

A Tabela 14 representa a planilha final que, com exceção da coluna *“Case number”*, foi exportada para a plataforma R. A variável “wip” representa a quantidade de *cases* em processamento dos 5.760 minutos (quatro dias). As colunas do *lead time* passaram pelo processo descrito na seção 5.3.1 e foram geradas através de uma função do pacote *BupaR*. As variáveis “Time” e “Nacionalidade” foram criadas para decompor os recursos que, dessa maneira, são agregados em grupos de até 22 atendentes, satisfazendo os requisitos da vetorização da plataforma R.

Tabela 15 – Dados finais de entrada dos modelos preditivos

irt *factor 2 levels*	**tempo_irt** *numeric*	**nacionalidade** *factor 8 levels*	**time** *factor 4 levels*	**recurso_dia** *numeric*	**volume_anterior** *numeric*	**volue_hora** *numeric*	**dia** *factor 7 levels*	**tipo_user** *factor 7 levels*	**problema** *factor 41 levels*	**idioma** *factor 15 levels*	**release** *factor 2 levels*	**created** *numeric*	**reopened** *numeric*	**wip** *numeric*	**hora** *factor 24 levels*	**atravessmento** *numeric*	**maior_dia** *numeric*
0	3	2	2	18	111	0	1	1	17	1	1	2821	474	501	0	91	0
0	60.2	2	2	18	111	1	1	2	6	1	1	2822	475	502	0	60	0
0	4.37	2	2	18	111	2	1	1	7	1	1	2823	476	503	0	6	0
0	3.43	2	2	18	111	3	1	1	18	1	1	2824	477	500	2	8	0
0	9.67	2	2	18	111	4	1	1	6	1	1	2825	478	501	2	10	0

Fonte: Elaborado pelo autor.

A Tabela 15 representa uma amostra dos dados tratados que foram utilizados como entradas para os modelos preditivos. Pode-se notar que as variáveis classificatórias (*factor)* com maior número de níveis (*levels*) foram o “problema” e a “hora”. Tais tipos de dados são analisados discretamente pelos algoritmos, de modo que foram necessárias mudanças nas classes das variáveis para se adequarem às diferentes análises realizadas no presente estudo.

Na próxima seção, serão descritos os procedimentos realizados para a análise dos dados.

3.6 Análise dos resultados

3.6.1 Modelos descritivos

A aplicação das técnicas de *process mining* foi realizada com base nas etapas propostas por Van der Aalst (2016). O planejamento do caso a ser estudado e a coleta de dados foram detalhados nas seções anteriores. Os modelos de controle foram gerados ao longo das análises e estão representados pelos resultados sem filtros. A adição de lógicas pertinentes ao processo ocorreram à medida que filtros e agrupamentos foram sendo aplicados para facilitar a interpretação dos resultados. Os algoritmos utilizados estão descritos no APÊNDICE C.

A análise descritiva foi realizada com a utilização do pacote *edea*R, o qual possui funções de análise exploratória e descritiva de *event logs*. Informações como o número de *cases*, atividades, eventos e caminhos foram geradas para que as dimensões e aspectos dos dados analisados pudessem ser compreendidos. Sendo assim, a frequência absoluta de cada atividade no *event log* foi apresentada, juntamente com a quantidade de níveis das mesmas.

Os níveis e a quantidade absoluta não são suficientemente explicativos, pois podem haver exceções nas quais poucas atividades foram repetidas diversas vezes em poucos *cases.* Dito isso, a frequência com que cada atividade apareceu em relação ao número de *cases* complementou a análise exploratória.

A junção de diferentes atividades em uma sequência temporal forma os caminhos, que foram detalhados de modo a evidenciar as atividades de início e fim do *event log* estudado. Foram extraídos os mapas dos caminhos de atividades para criar entendimento acerca das possíveis combinações que as mesmas podem ter no

processo analisado. Dessa maneira, foram aplicados filtros definidos de acordo com a existência de atividades em relação ao número de *cases*, de modo que fossem apresentados os caminhos que representam as atividades predominantes do período.

A mesma abordagem de filtragem foi aplicada para a interpretação das sequências de atividades que compõem o início e o fim dos *cases*. Os caminhos mais utilizados no *event log* foram considerados para que, posteriormente, os mapas do processo a matriz de predeceção fossem melhor compreendidos.

A descoberta de processo teve como base a matriz de predeceção, que foi gerada para apresentar todas as relações entre uma atividade e outra, assim como suas frequências absolutas. Os caminhos anteriormente citados são gerados a partir de tais relações, assim como os mapas do processo. Dessa maneira, a matriz de predeceção serve para concluir a interpretação dos caminhos e introduzir os mapas do processo.

Os mapas do processo representam um diagrama de espaguete com base na matriz de predeceção. Tais mapas podem ser focados na frequência das atividades ou na performance do sistema. Para a presente análise, eles foram gerados e filtrados com base nos resultados dos caminhos, de modo a se tornarem melhor interpretáveis e mais representativos. Os mapas de performance consideraram os tempos de espera entre as atividades, visto que cada uma representa interações pontuais com o CRM e seus tempos de ciclo são desprezíveis.

A análise de desempenho do processo se mostrou necessária para mostrar o impacto das duas abordagens anteriores na performance do sistema. Dessa forma, foram extraídos resultados para analisar o tempo de atravessamento dos *cases* em relação ao volume dos mesmos, de modo a compreender quanto os engajamentos mais duradouros impactam o sistema.

O tempo que os *cases* demoraram para ser processados é um dado importante sob o ponto de vista gerencial. Apesar disso, deve-se levar em conta a distribuição da demanda ao longo da semana analisada para que se possa concluir que há um padrão de comportamento. Sendo assim, foram construídos gráficos de volume e tempo de atravessamento dos *cases*, agrupados por dia de criação. Tais análises possibilitaram a influência dos dias tanto sobre a demanda, quanto sobre os *lead times.*

Finalmente, o *event log* foi analisado à luz da distribuição da demanda e do tempo de atravessamento em função da hora do dia. Tal análise foi realizada para identificar possíveis variáveis que pudessem impactar a performance do sistema ao

longo do dia. Dessa forma, a quantidade de *cases* criados e o tempo de fechamento dos mesmos foram analisados em três granularidades diferentes: hora, dia e no período como um todo.

3.6.2 Modelos preditivos

Na presente análise, foram construídos modelos preditivos para classificar se o SLA de 120 minutos para a primeira resposta será atendido ou violado, bem como o tempo fechamento de um *case* excederá 1440 minutos. Para ambas as classificações, foram utilizados os algoritmos *Random Forests* e *Support* e *Vector Machines* (SVM), respectivamente selecionados por não apresentar problemas de *overfitting* (BREIMAN, 2001) e com base na sugestão de Ainslie et al. (2017). Em decorrência da diferença entre os processos de seleção de variáveis e predições, a metodologia das duas abordagens foi explicada separadamente.

3.6.2.1 Seleção das variáveis

A seleção de variáveis de entrada para modelos preditivos pode ser feita através de filtros estatísticos ou predições (KUHN; JOHNSON, 2013). Nesse estudo, as duas modalidades foram aplicadas tanto para a classificação da violação do SLA quanto do tempo de atravessamento maior que um dia. Foram aplicadas análises de correlação, testes de hipóteses e funções do *Random Forests* para definir o conjunto final de variáveis para cada classificação.

Inicialmente, foi realizada uma análise de correlação para identificar os dados altamente correlacionados. Variáveis com alta correlação entre si podem atrapalhar os algoritmos de classificação, uma vez que não possuem comportamentos diferentes para modificar a classe de uma resposta. Sendo assim, o APÊNDICE D apresenta o algoritmo utilizado, que desconsiderou variáveis que possuem coeficiente de correlação maior que 0,75 (KUHN; JOHNSON, 2013).

Quadro 14 – Modelos preditivos de seleção de variáveis

Biblioteca e função	**Descrição**	**Teste final**	**Autor**
Biblioteca: *Boruta* Função: Boruta(resposta, data = dados, doTrace = ()	Remoção de variáveis que não impactem a variável de resposta significantemente ,a partir da aplicação de *Random Forests*	Teste de hipóteses (p-valor: 0,01)	Kursa e Rudnicki (2011)
Biblioteca: *caret* Função: rfe(x=dados, y=resposta, sizes = (), rfeControl = ()	Aplicação de algoritmos baseado em *Random Forests* para avaliar a utilização das variáveis na composição da resposta	*k-fold cross validation*	Kuhn e Johnson (2013)

Fonte: Elaborado pelo autor.

O Quadro 14 apresenta as diferenças entre as duas técnicas preditivas utilizadas para selecionar as variáveis de entrada dos modelos. O pacote *Boruta* é baseado em *Random Forests* e sua resposta é resultado de um teste de hipóteses. *Recursive feature selection* é uma técnica para avaliar a importância das variáveis através de funções preditivas e o teste final utilizado foi *cross-validation* com 10 *folds*, o qual divide os dados em 10 amostras para avaliar a relevância das variáveis em cada uma (KUHN; JOHNSON, 2013).

As duas abordagens foram aplicadas sobre os dados de treinamento tanto para a classificação de violação do SLA, quanto para o tempo de fechamento. Nesse estudo, esses dados representam 80% do total de observações e servem para treinar os algoritmos. Os outros 20% são utilizados para testar os modelos e tiveram suas variáveis mantidas de acordo com os dados de treinamento. Tal proporção é considerada padrão no campo do *machine learning* e foi aplicada através da função *createDataPartition()*, a qual divide os dados de modo a garantir que ambas as amostrar possuam os mesmos níveis (KUHN; JOHNSON, 2013). O algoritmo utilizado para a seleção de variáveis através de modelos preditivos é apresentado no APÊNDICE E.

3.6.2.2 Configuração dos modelos preditivos

O presente estudo utilizou dois algoritmos preditivos para classificar a ocorrência de violações do SLA e de tempos de atravessamento maiores que um dia.

Dessa maneira, as bibliotecas *caret* (KUHN, 2008) e *DALEX* (BIECEK, 2018) foram utilizadas para as predições e explicações dos modelos. Foi utilizada a função *train()* para treinar os dados pois a mesma possibilita a utilização de vários métodos de *machine learning* e ajusta os parâmetros automaticamente através dos argumentos *trControl* e *tuneLength*. O APÊNDICE F apresenta os modelos preditivos.

Quadro 15 – Bibliotecas e técnicas utilizadas

Biblioteca	**Técnica**	**Descrição**	**Autor**
Caret	*Random Forests*	Consiste num conjunto de árvores classificadoras que escolhem os resultados predominantes dada uma determinada entrada, com base em vetores aleatoriamente distribuídos	Breiman (2001)
	Support Vector Machines (SVM)	Cálculo da distância entre duas classes distintas e um limite denominado margem, o qual idealmente serve como separador de tais classes	Kuhn e Johnson (2013)
	Variable Importance	Caracterização do efeito geral das variáveis em um modelo	Kuhn (2008)
DALEX	*Variable Response*	Verificação da capacidade de um algoritmo em captar as relações não lineares de uma variável	Biecek (2018)
	Model Performance	Exploração contínua de erros residuais de modelos preditivos	

Fonte: Elaborado pelo autor.

Todos os métodos descritos no Quadro 15 foram aplicados nas situações de violação de SLA e tempo de atravessamento maior que um dia. O argumento *tuneLength* foi utilizado com o valor padrão dos algoritmos e o *trControl* foi customizado para validar os modelos com o método *k-fold cross-validation*. Foram utilizados 5 *folds* em decorrência do tamanho da amostra e por motivos de

performance, pois grandes massas de dados tendem a possuir erros similares sob diferentes validações (KUHN; JOHNSON, 2013)

Gerados os modelos, a função *varImp()* do *caret* foi aplicada sobre os modelos treinados para que os gráficos que explicitam a importância das variáveis pudessem ser criados. Dessa forma, cada algoritmo pôde ser avaliado não apenas quanto à sua acurácia, mas também quanto às variáveis que ele utilizou para compor a resposta

As funções *model_performance* e *variable_response* da biblioteca *DALEX* foram utilizadas para avaliar, respectivamente, a performance do modelo em relação aos seus erros e o comportamento do mesmo em função das variáveis. Para fins analíticos, quanto menor o erro residual, mais preciso é o modelo. A segunda função avalia a resposta dos modelos em relação à variação dos atributos de entrada, de modo a existirem algoritmos que se adequam melhor o comportamento de determinadas variáveis. O APÊNDICE G apresenta os modelos utilizados para a análise das variáveis.

4 ANÁLISE DOS RESULTADOS

Neste capítulo, são apresentados a empresa e o processo analisado, assim como os resultados obtidos nas análises, as quais foram realizadas com o auxílio de planilhas eletrônicas e dos pacotes *tidyverse, e1071* (DIMITRIADOU et al., 2009)*, DALEX, caret* e *bupaR,* todos disponíveis na plataforma R. Em decorrência das diferentes naturezas das análises, os resultados foram divididos em modelos descritivos e preditivos.

O *process mining*, além de prover entendimento acerca do processo e seus fatores de impacto, possibilitou a composição de variáveis como *time*, *hora*, dia da semana (*dia*), volume do dia anterior (*volume_anterior*), demanda por hora (*volume_hora*), *cases* abertos (*created)*, reabertos (*reopened*) e em processamento dos últimos 4 dias (*wip*). Dessa forma, tais atributos serviram de entradas aos modelos preditivos.

As variáveis que impactam no atendimento ao SLA são diferentes das que explicam o tempo de fechamento de um *case*. As etapas de seleção de variáveis evidenciaram que, para cada objetivo, tanto o teste de hipóteses do algoritmo *Boruta* quanto a aplicação de *Recursive Feature Selection* do *caret* apresentaram resultados diferentes. Dessa forma, os algoritmos *Random Forests* e *Support Vector Machines* (SVM) divergiram em performance para a predição de violação do SLA e para a classificação do tempo de atravessamento dos *cases*, mostrando que as variáveis que impactam uma situação não explicam a outra.

4.1 Organização analisada

A pesquisa foi realizada com dados de 8 escritórios de uma empresa multinacional de software empresarial. Cada escritório é localizado em um país diferente e a pesquisa se limita a analisar o help desk de um software específico que é comercializado pela empresa. O software em questão foi incorporado à empresa em 2014, como parte da estratégia global de aumentar a participação de mercado com softwares em nuvem.

Os turnos de trabalho são desprezíveis uma vez que todas as localidades atuam sobre a mesma fila de *cases* e as diferentes localidades existem para garantir que a capacidade prevista esteja disponível 24 horas por dia. Os clientes podem ser

de qualquer lugar do mundo e as interações com os mesmos são monitoradas através de um CRM. Na próxima seção, será apresentado o processo analisado.

4.2 Processo analisado

O processo analisado é o help desk de um software específico da empresa. As entradas do processo podem ser tanto ligações quanto e-mails, mas ambos precisam ser registrados através de *cases* que, por sua vez, requerem uma resposta inicial em até 2 horas a partir da criação de acordo com o SLA. No caso dos e-mails, o CRM envia respostas automáticas aos remetentes para que confirmem as credenciais, mas tais respostas não são computadas como cumprimento de SLA. Nas ligações, o *case* é criado pelo próprio atendente.

Há cinco tipos de usuários que contatam o help desk e cada tipo requer um atendimento diferente, uma vez que as funcionalidades do sistema são visíveis de acordo com o tipo de usuário. Tal situação é representada na análise pela variável *user_type*. Outra variável do processo é o time ao qual o atendente pertence, uma vez que a fila de *cases* em questão possui um time global responsável, mas outros times são contatados para trabalhar na fila em situações como falta de funcionários ou solicitações que requerem conhecimentos específicos. O software é submetido a uma *release* por trimestre, que implementa atualizações e correções no sistema, sendo uma possível fonte de erros e inconsistências no sistema. Dessa forma, a ocorrência de *release* é representada na análise pela variável *release*.

Criado o *case*, o mesmo pode ir para a fila de incidentes ou ser diretamente remetido a um atendente no caso de uma ligação. Assim que um recurso começa a trabalhar em um *case*, a primeira ação é completar os dados do usuário e da natureza do problema. Na análise, tal ação é representada pela atividade *edit*. À medida que o atendente investiga a solicitação, comentários, os quais serão representados pela atividade *enter_comment*, são agregados ao *case*. Se necessário, o *case* pode ser assinado para algum recurso especialista ou time de desenvolvimento, caracterizando a atividade *reassigned.*

No final do processo, são enviados e-mails para os clientes para apresentar as soluções, pedir informações adicionais ou discutir os problemas. Todas as informações relevantes continuam a ser agregadas ao *case* através de comentários

e, quando o mesmo é solucionado, o último e-mail é enviado ao cliente e o *case* é fechado, podendo ainda ter comentários adicionais ou ser reaberto.

4.3 *Process mining* – Modelos Descritivos

Nesta seção, são apresentados os resultados do *process discovery,* além das análises descritivas e de desempenho. Esses resultados servem de subsídio à melhor compreensão quantitativa do processo analisado, de modo a possibilitar o descobrimento de possíveis variáveis que possam servir de entradas para os modelos preditivos.

Os resultados contemplam um período de uma semana, contendo dados de 23 a 29 de setembro de 2018. Além disso, cada evento representa uma interação pontual com o *case* e não há registros no *event log* sobre a etapa do processo que está sendo executada em determinado momento.

4.3.1 Análise Descritiva

Na Tabela 16 são apresentadas a quantidade de *cases*, atividades, eventos, recursos e caminhos do *event log* analisado. Tais resultados têm como objetivo apresentar as quantidades absolutas de cada componente do *event log*, de modo a definir as fronteiras quantitativas de estudo às quais a análise inicial se submete.

Tabela 16 – Dimensionamento do *event log*

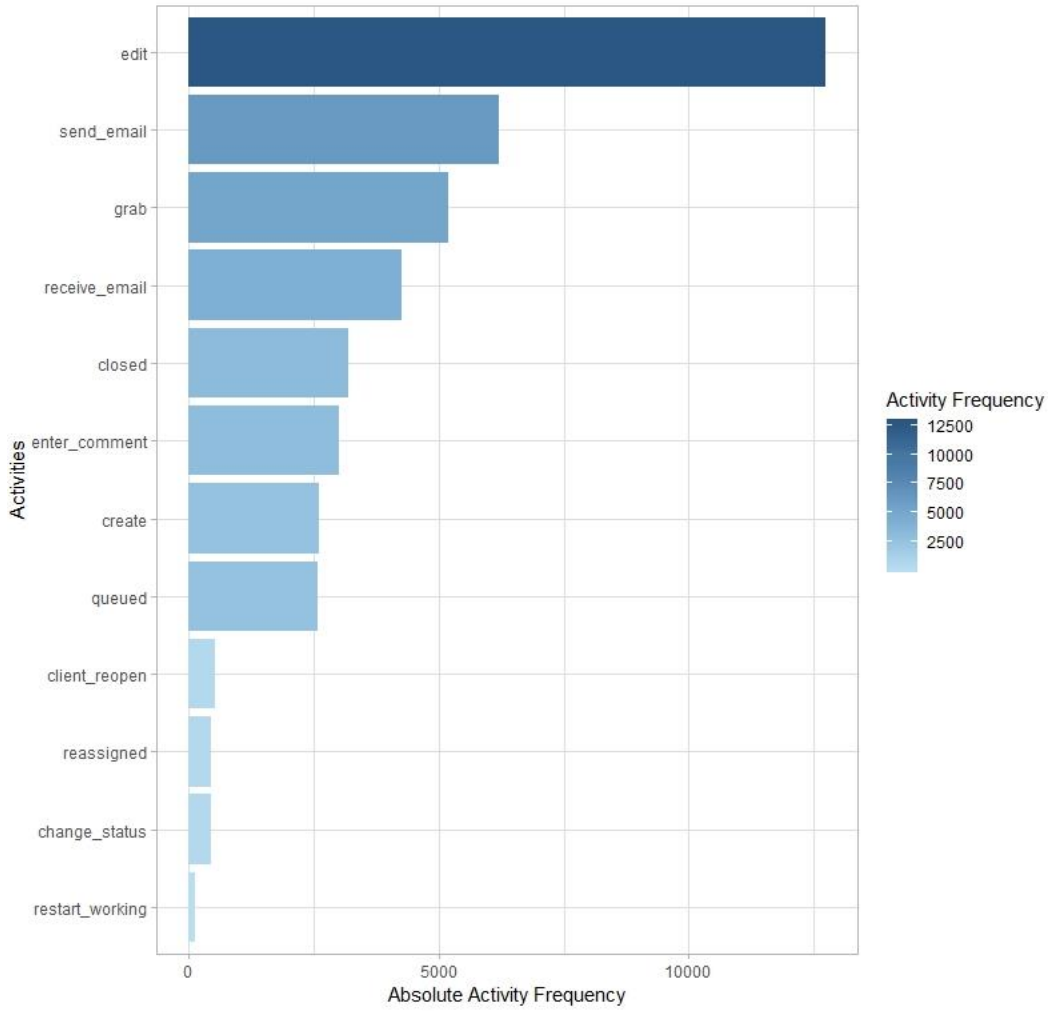
Dimensão	**Quantidade absoluta**
Cases	2619
Atividades	12
Eventos	41440
Recursos	84
Caminhos	1286

Fonte: Elaborado pelo autor.

Os resultados da Tabela 16 mostram que 1286 *cases,* 49,10% do total, possuíram caminhos únicos, evidenciando que caminhos repetidos ocorreram em 50,90% dos *cases* analisados. Além disso, considerando-se que cada *case* possui um

recurso dedicado que realiza até 12 atividades, verifica-se que houve, em média, 31,18 *cases* alocados para cada atendente.

Gráfico 1 – Frequência absoluta de atividades no *event log*



Fonte: Elaborado pelo autor.

O Gráfico 1 apresenta a frequência absoluta de cada atividade no *event log*. É possível concluir que a atividade que mais ocorreu foi a edição dos *cases*, representada pela atividade *edit* e com mais de 12500 ocorrências. O envio de e-mails, a assinatura de *cases* e o recebimento de e-mails apareceram, respectivamente, logo em seguida como atividades mais realizadas. As atividades *closed, enter_comment, create* e *queued* foram, respectivamente, as próximas atividades mais presentes no *event log*, com no mínimo 2500 aparições cada. Finalmente, com menos de 2500 ocorrências, as atividades *cliente_reopen, reassigned, change_status* e *restart_working* foram as menos realizadas.

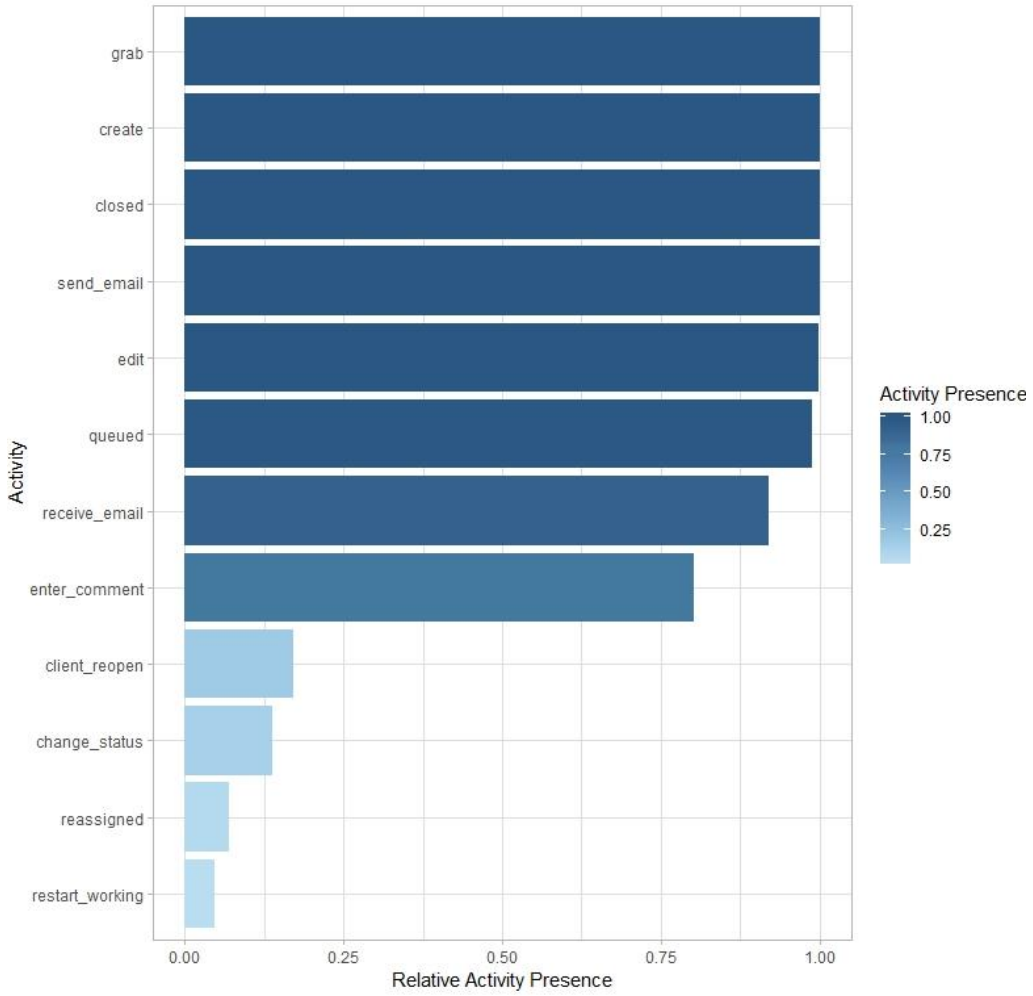
Tabela 17 – Distribuição de frequências das atividades em função dos *cases*

Atividade	**Frequência absoluta**	**Frequência relativa**	**Frequência absoluta acumulada**	**Frequência relativa acumulada**
Create	2619	12,29%	2619	12,29%
Closed	2619	12,29%	5238	24,58%
Grab	2619	12,29%	7857	36,88%
Send_email	2619	12,29%	10475	49,16%
Edit	2616	12,28%	13091	61,44%
Queued	2590	12,16%	15681	73,60%
Receive_email	2409	11,31%	18090	84,91%
Enter_comment	2098	9,85%	20188	94,75%
Cliente_reopen	449	2,11%	20637	96,86%
Change_status	360	1,69%	20997	98,55%
Reassigned	185	0,87%	21182	99,42%
Restart_working	124	0,58%	21306	100,00%
Total	**21306**	**100,00%**	**-**	**-**

Fonte: Elaborado pelo autor.

A Tabela 17 apresenta a distribuição de frequências das diferentes atividades em função da quantidade de *cases* em que elas aparecem. Tais resultados mostram que 61,44% de todas as atividades realizadas representam o fluxo básico do processo analisado (*Create, Grab, Edit, Send_email, Closed)*, seja para as ligações ou para os e-mails. Além disso, 94,75% dos *cases* possuem as atividades *queued*, *receive_email* e *enter_comment* as quais representam, respectivamente, 12,16%, 11,31% e 9,85% das atividades totais, evidenciando a relevância das mesmas no processo. As atividades *cliente_reopen*, *change_status*, *reassigned* e *restart_working* representam 5,25% do total, mostrando o caráter de exceção das mesmas.

Gráfico 2 – Frequência relativa de atividades em função do número de *cases*



Fonte: Elaborado pelo autor.

O Gráfico 2 mostra a presença das atividades nos *cases* em termos relativos. Como verificado na Tabela 17, as atividades *create, grab, send_email* e *closed* aparecem em todos os *cases* do *event log* analisado. Tal gráfico também mostra que mais de 75% dos *cases* analisados possuem comentários (atividade *enter_comment*) e que as atividades *cliente_reopen*, *change_status*, *reassigned* e *restart_working* são as que menos aparecem nos *cases* do *event log*, demonstrando que não são predominantes as vezes em que um *case* é reaberto, reassinado para um especialista, colocado em espera ou assinado de volta ao atendente.

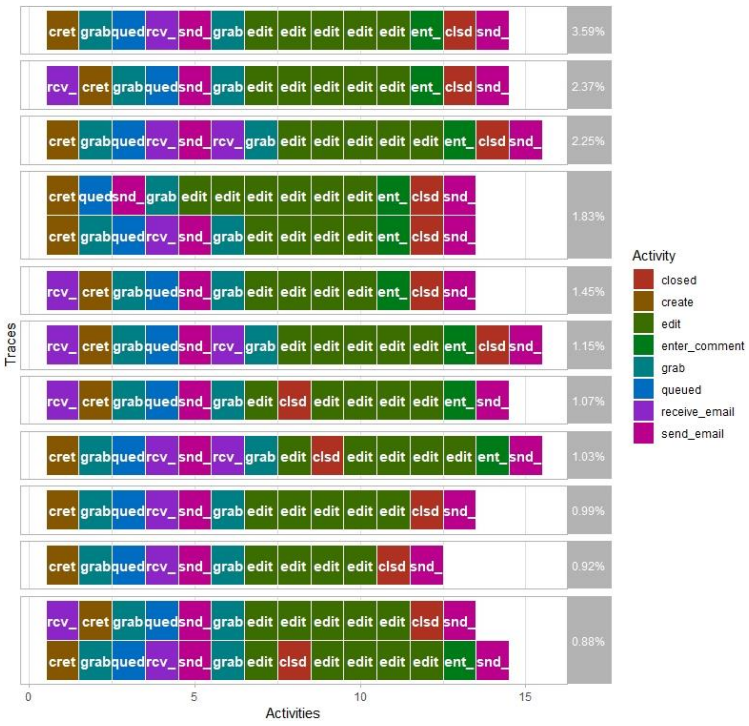
Tabela 18 – Distribuição de frequências do posicionamento das atividades

Posição	Frequência absoluta	Frequência relativa	Frequência absoluta acumulada	Frequência relativa acumulada
Inicial	3	25,00%	3	25,00%
Intermediária	2	16,67%	5	41,67%
Final	7	58,33%	12	100,00%
Total	**12**	**100,00%**	**-**	**-**

Fonte: Elaborado pelo autor.

A Tabela 18 apresenta a distribuição de frequências das atividades em relação ao posicionamento das mesmas no ciclo de vida dos *cases*. De acordo com a coluna da frequência relativa, 58,33% das atividades do *event log* são realizadas ao final dos *cases*. Além disso, a Tabela 18 mostra que apenas 16,67% das atividades não fazem parte da abertura ou do fechamento dos *cases*. Dessa forma, de modo a complementar o entendimento de que 7 atividades são realizadas em 84,91% dos *cases* (Tabela 17) e que 83,33% das atividades ocorrem nos extremos dos *cases* (Tabela 18), são apresentados os caminhos do *event log* na Figura 11.

Figura 11 – Caminhos do *event log*

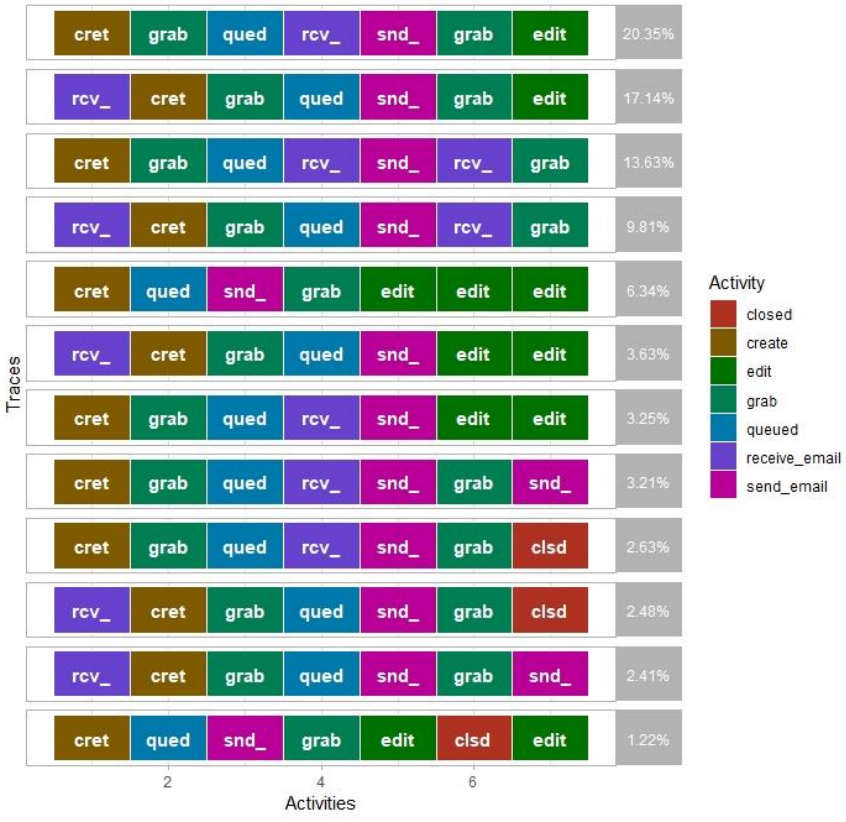


Fonte: Elaborado pelo autor.

A Figura 11 mostra que as atividades iniciais predominantes são *create* e *receive_email*. Tanto nas ligações quanto nos e-mails, a atividade *create* pode ser a primeira pois o *case* é criado automaticamente quando um novo e-mail é recebido ou quando o atendente registra a ligação. Da mesma forma, a Figura 11 mostra que a principal atividade final é *send_email*, evidenciando o fato de que o atendente escreve os comentários, edita e fecha os *cases* antes de enviar um e-mail de fechamento.

Visto que, de acordo com a Tabela 17, 84,91% dos *cases* do *event log* analisado possuem 7 das 12 atividades presentes nos dados e que, de acordo com a Tabela 18, o *event log* possui três atividades iniciais e sete finais, as Figuras 12, 13 e 14 apresentam detalhamentos dos caminhos percorridos por 84,91% dos caminhos analisados.

Figura 12 – Caminho das sete primeiras atividades

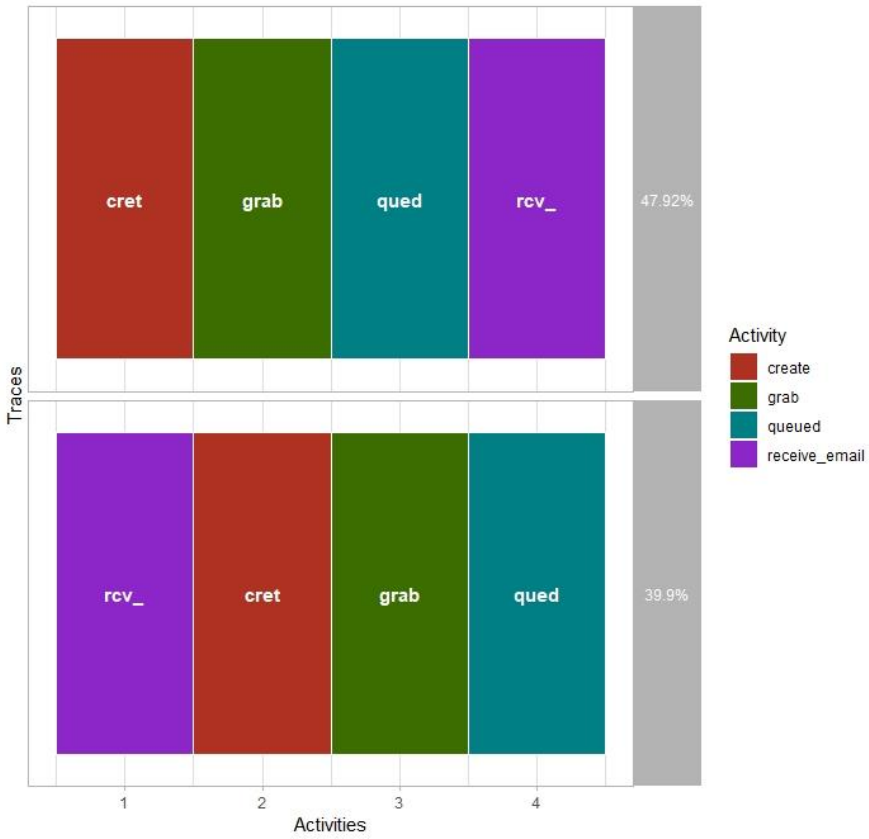


Fonte: Elaborado pelo autor.

A Figura 12 apresenta as variações relativas das sete primeiras atividades em 84,91% dos caminhos analisados. Pode-se concluir que, em 7,56% dos caminhos, não há recebimento de e-mail nas primeiras atividades, evidenciando se tratarem de

caminhos originados por ligações. É importante destacar que ao enviar um e-mail ao help desk, um *case* é automaticamente criado, assinado à fila e colocado em espera. Sendo assim, 78,54% dos caminhos apresentados podem ter as quatro primeiras atividades consideradas iguais, uma vez que elas ocorrem automaticamente ao mesmo tempo. Finalmente, conclui-se que 6,33% dos caminhos analisados na Figura 12 possuíram a atividade *closed* entre as sete primeiras.

Figura 13 – Caminho das quatro primeiras atividades

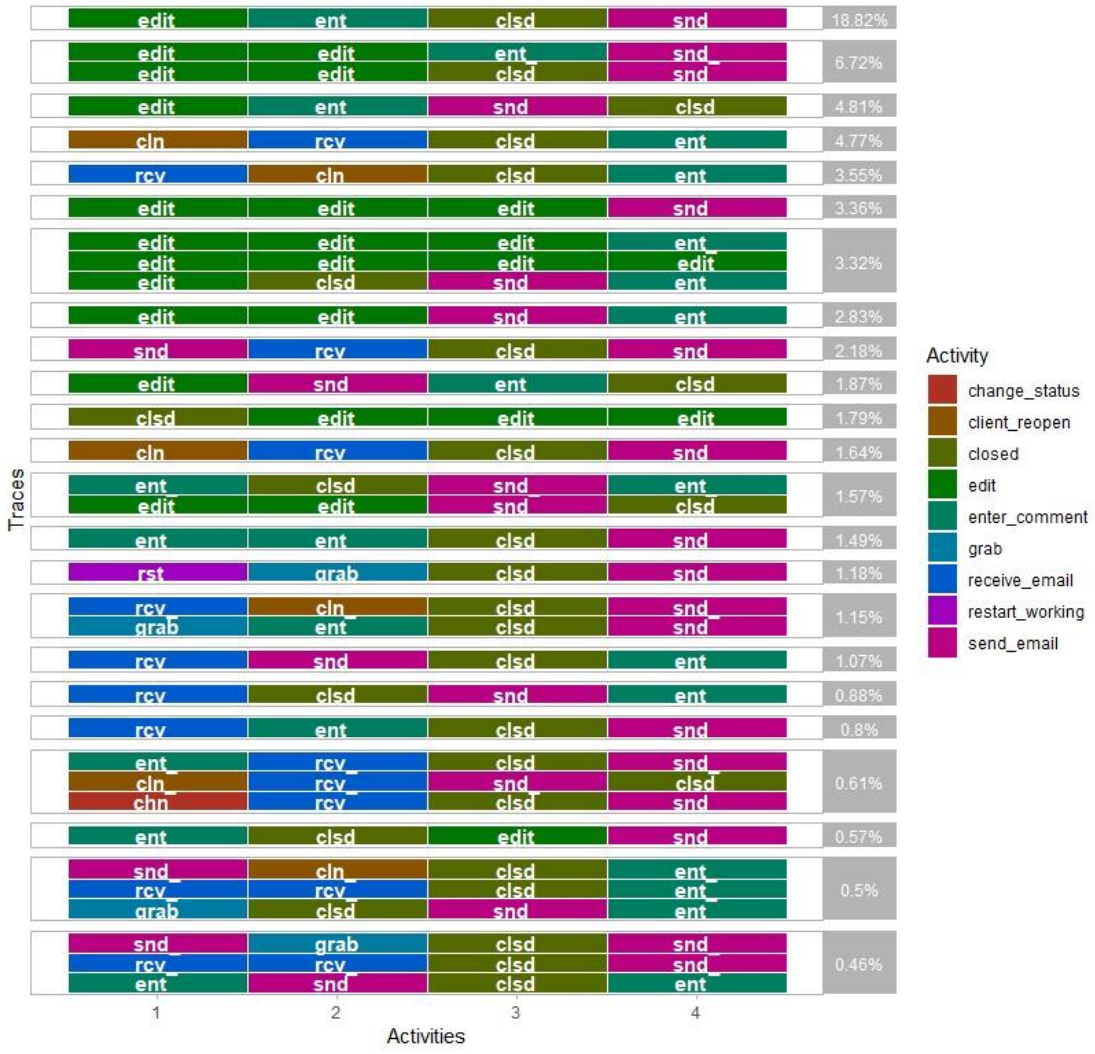


Fonte: Elaborado pelo autor.

A Figura 13 apresenta as quatro primeiras atividades verificadas em 84,91% dos caminhos analisados. Visto que houveram dois caminhos com quatro atividades permutadas e que tais atividades possuíram o mesmo *timestamp* pois foram realizadas automaticamente a partir do recebimento do e-mail do cliente, conclui-se que, em 84,91% dos caminhos do *event log* analisado, a primeira atividade foi sempre o recebimento de e-mail, que é representado na figura 4 pelo conjunto das atividades *create*, *grab*, *queued* e *receive_email*. Dito isso, também é possível concluir que as

ligações, que são as entradas no sistema as quais não possuíram o recebimento de e-mail como primeira atividade, não estão contidas em 87,82% do *event log* analisado.

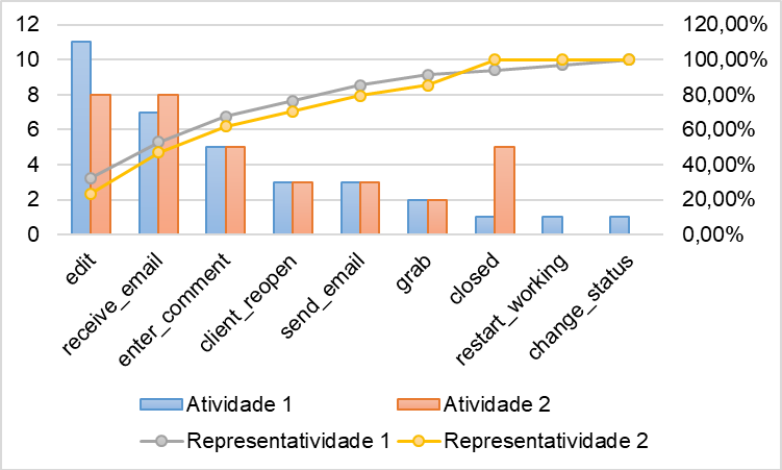
Figura 14 – Caminho das quatro últimas atividades



Fonte: Elaborado pelo autor.

A Figura 14 apresenta as quatro atividades finais realizadas em 84,91% dos caminhos analisados. O *event log* apresentou maior variação de atividades nas posições 1 e 2 do que em 3 e 4. Isso se deve pela natureza do processo que exige que todos os *cases*, ao serem fechados (*closed*), possuam um comentário (*enter_comment*) e um e-mail final enviado ao cliente (*send_email*). Visto que as duas últimas atividades do caminho analisado endereçam o fechamento dos *cases*, o gráfico 3 foi construído para detalhar as atividades das posições 1 e 2.

Gráfico 3 – Atividades verificadas nas posições 1 e 2



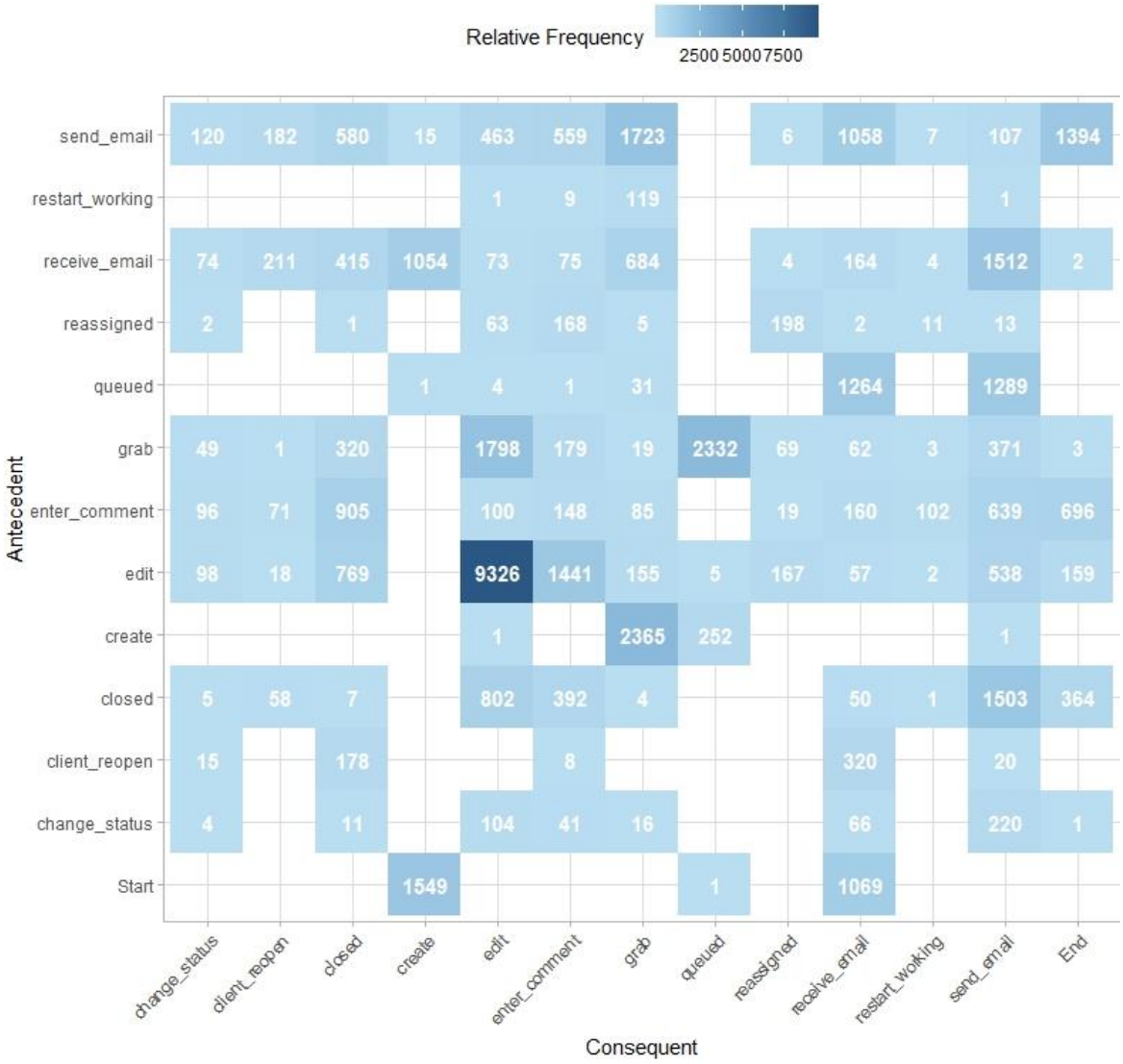
Fonte: Elaborado pelo autor.

O Gráfico 3 mostra que a atividade que mais ocorreu nos caminhos analisados da figura 13 foi *edit*, com 32,35% e 23,53% de representatividade nas posições 1 e 2, respectivamente. Em seguida, aparece a atividade *receive_email*, com 20,59% e 23,53% de representatividade nas posições 1 e 2, respectivamente. Tais atividades, combinadas à menor representatividade de *closed*, *enter_comment* e *send_email*, demonstram que as posições 1 e 2 dos caminhos da Figura 14 possuem caráter mais investigativo que as posições 3 e 4, que são basicamente compostas por atividades obrigatórias de fechamento.

4.3.2 Descoberta de processos

A descoberta de processos possibilita o entendimento acerca das relações entre as atividades do *event log*. Para isso, foi gerada uma matriz de predeceção que serve como base para os mapas do processo, os quais foram criados em duas etapas para facilitar a visualização. Sendo assim, é possível verificar as frequências com que duas atividades são realizadas em sequência, bem como as repetições e retrabalhos do processo.

Figura 15 – Matriz de predeceção do *event log*



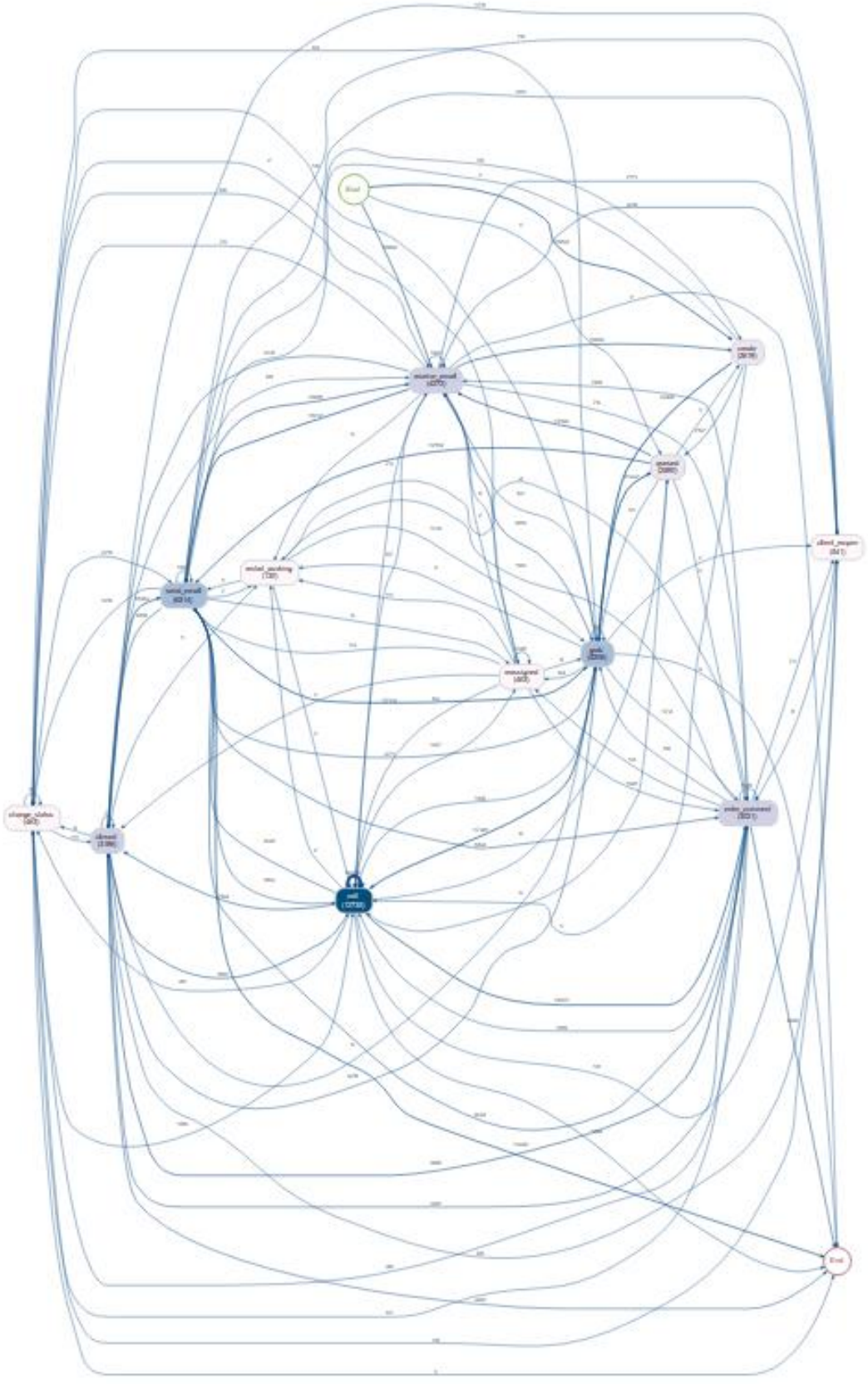
Fonte: Elaborado pelo autor.

A Figura 15 apresenta a matriz de predeceção do *event log* analisado, de modo a complementar a interpretação dos caminhos. Considerando-se que cada evento é o registro de uma atividade realizada, pode-se concluir que a atividade *edit* é a que mais ocorreu de modo consecutivo, com 9326 ocorrências. Isso acontece pois há diversos parâmetros exclusivos de cada cliente e usuário que o atendente pode definir no *case*.

A atividade *client_reopen* antecedeu 363 outras atividades que não foram fechar um *case.* Cenário esse que é parecido com o da atividade *reassigned,* a qual desencadeou 462 atividades não relacionadas a fechamento. Esses dados mostram que se um *case* foi reassinado para um time especialista ou reaberto pelo cliente, mais atividades precisaram ser realizadas. Com base na matriz apresentada na Figura 15,

foi gerado o mapa completo do processo sob a perspectiva da frequência das atividades.

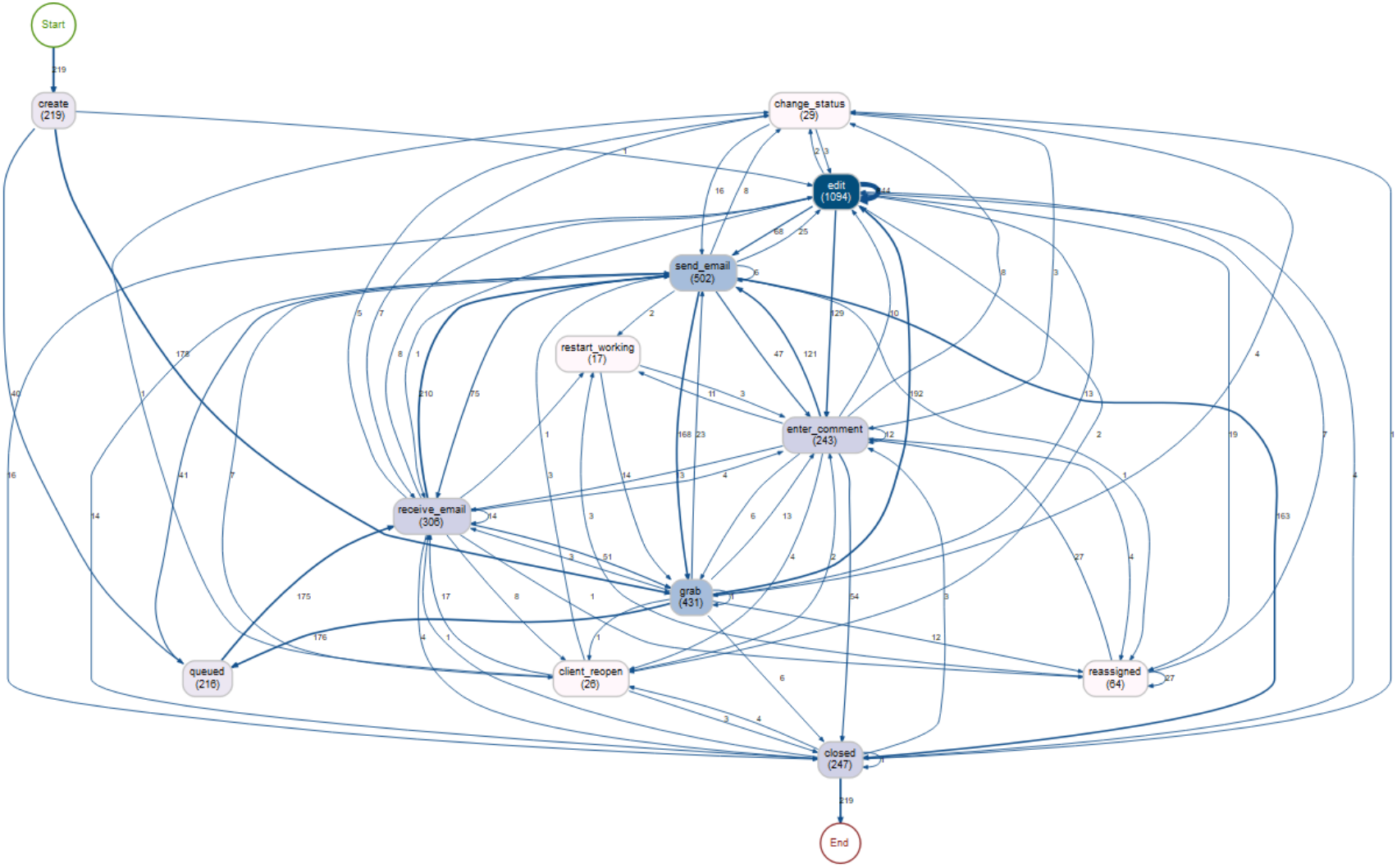
Figura 16 – Mapa do processo sem filtros



Fonte: Elaborado pelo autor.

A Figura 16 apresenta o mapa do fluxo do processo analisado com base na frequência de ocorrência das atividades representadas por eventos. Em decorrência da quantidade de caminhos possíveis apresentados na Tabela 16 e das variações de tais caminhos, mesmo que para representar ações semelhantes, apresentadas nas Figuras 11, 12, 13 e 14, o mapa do processo como apresentado na Figura 16 não é representativo, uma vez que abrange caminhos não relevantes e é de difícil compreensão. Com o objetivo de melhorar a visualização do fluxo do processo, a Figura 17 apresenta o mapa no qual a atividade inicial é a criação do *case* (*create*) e a final é o fechamento do mesmo (*closed*).

Figura 17 – Mapa de frequências do processo



Fonte: Elaborado pelo autor.

O mapa do processo apresentado na Figura 17 apresenta todos os eventos do *event log* que possuem a primeira atividade *create* e a última *closed*. Todas as 12 atividades foram agregadas e repetidamente realizadas no mapa. As atividades *reassigned*, *change_status* e *restart_working* foram respectivamente realizadas 64, 29 e 17 vezes. Pode-se concluir que as atividades que mais geraram eventos foram *edit* e *send_email*, com frequências absolutas de 1094 e 502, respectivamente

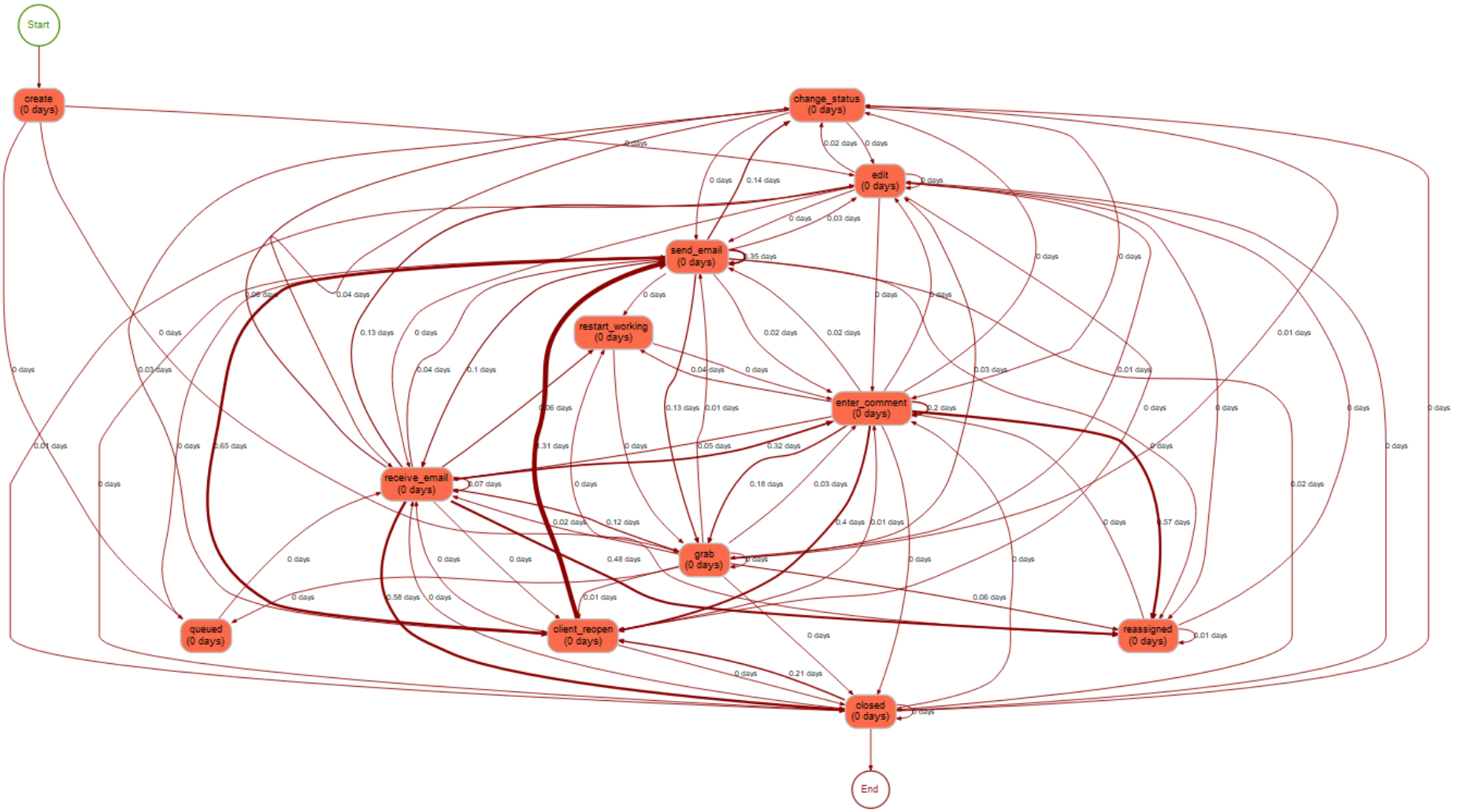
A atividade *reassign* foi constatada em 29,22% dos eventos do mapa. No entanto, 42,18% de tal ação foi realizada em *cases* existentes, demonstrando que 12,32% dos *cases* analisados no mapa foram reassinados, de modo confirmar a baixa frequência absoluta da atividade no Gráfico 1. Sendo a atividade *reassign* a origem de outras, o as frequências absolutas do *event log* são impactadas pela presença da atividade em questão, mesmo que com pouca representatividade.

Há uma diferença em termos absolutos entre as atividades *create* e *closed*. Tal situação é explicada pela frequência absoluta da atividade *cliente_reopen* que, com 26 ocorrências na Figura 17, reduz a diferença entre a quantidade de *cases* criados e fechados para 2. Como houve reincidência da atividade *closed* em um *case* e o mapa mostra que houve uma mudança de status depois de o *case* ter sido fechado, conclui-se que, em termos do número de *cases* analisados, a frequência das atividades *created* e *closed* é a mesma.

4.3.3 Análise de Desempenho do Processo

Nesta seção, serão analisados os resultados referentes ao desempenho do processo. Foram gerados o mapa do processo com foco nos tempos de espera entre as atividades, além dos gráficos de tempo de atravessamento e de volume de *cases* tanto em relação aos dias da semana, quanto às horas do dia. Dessa forma, é possível analisar não apenas os tempos do processo em si, mas também outras características da demanda que podem influenciar na performance do sistema.

Figura 18 - Mapa dos tempos de espera do processo.



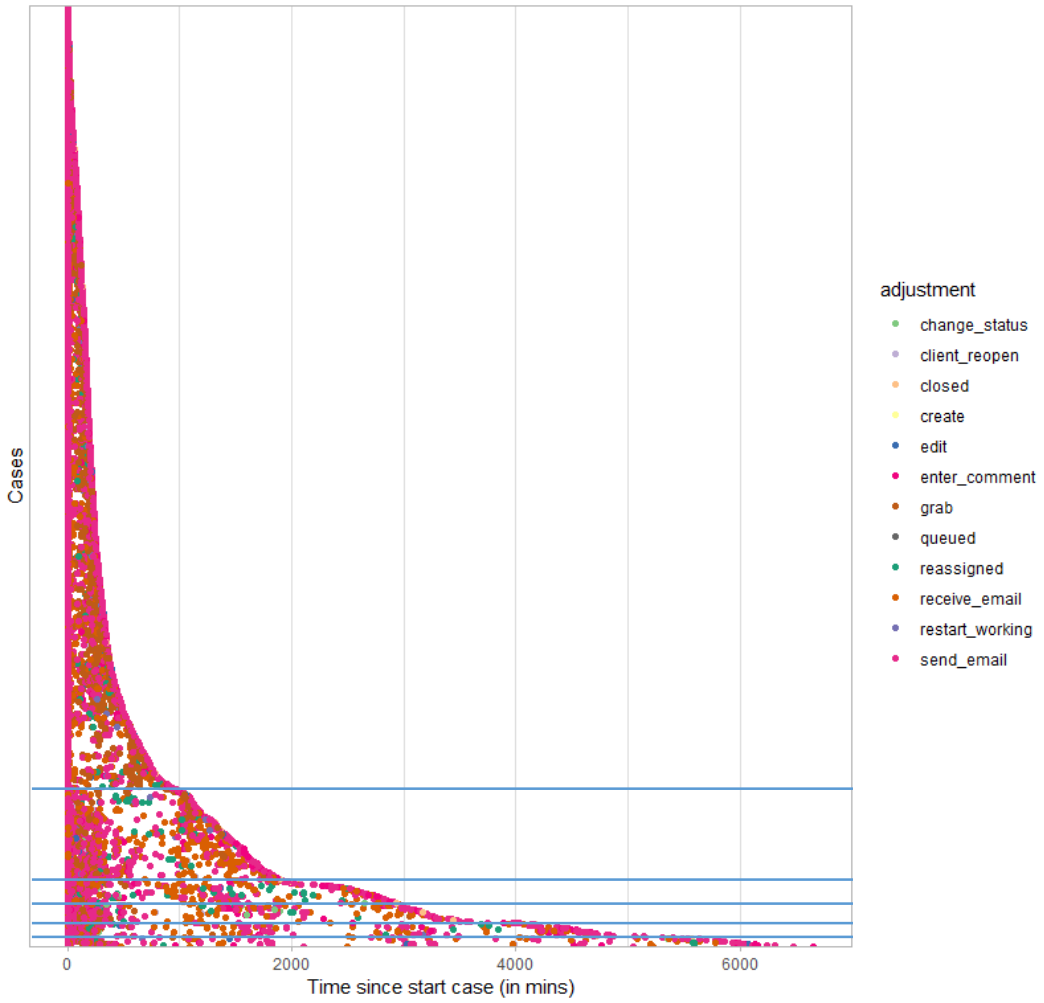
Fonte: Elaborado pelo autor.

A Figura 18 apresenta o mapa dos tempos de espera do processo. A partir dela, pode-se concluir que a maior espera ocorreu entre as atividades *cliente_reopen* e *send_email*, com 1886,4 minutos. O contrário também é verdadeiro, uma vez que o caminho inverso entre as duas atividades apresentou o segundo maior tempo de espera: 936 minutos. Isso significa que as maiores esperas do processo se deram em decorrência da comunicação adicional necessária nos *cases* reabertos pelos clientes.

A atividade *reassigned* também se apresentou relevante no mapa. O tempo de espera apresentado entre *enter_comment* e *reassigned* foi de 820,8 minutos. Já o tempo entre as atividades *receive_email* e *reassigned* foi de 691,2 minutos. O fato desses tempos se mostrarem relevantes no mapa pode acarretar maiores tempos de atravessamento dos *cases*.

Os tempos entre as repetições das mesmas atividades também se mostram relevantes na Figura 18. Considerando-se que o tempo de ciclo de cada atividade é nulo, uma vez que os eventos possuem apenas o *timestamp* em que ocorreram, a frequência com que eles se repetem e o tempo entre as ocorrências se tornam relevantes. A atividade *send_email* apresentou um tempo entre as repetições de 504 minutos. As ocorrências consecutivas da atividade *enter_comment* possuíram intervalos de 288 minutos. Pode-se concluir que tais tempos foram consequências da atividade *reassigned*, uma vez que, ao serem reassinados para outros times, os *cases* precisam ser comentados e os clientes comunicados.

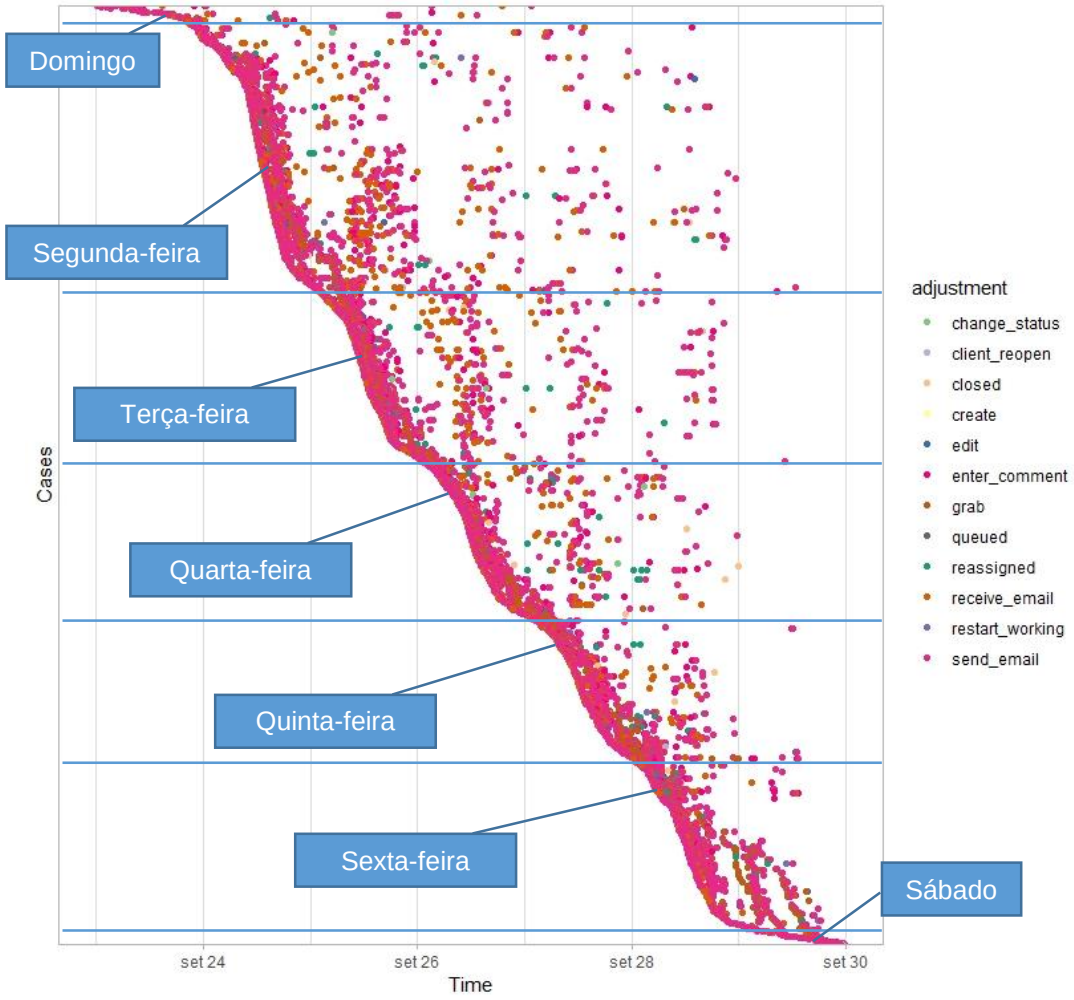
Figura 19 – Tempo de atravessamento dos *cases*



Fonte: Elaborado pelo autor.

A Figura 19 apresenta a representação do tempo de atravessamento dos *cases* ordenada em ordem decrescente. Os eixos horizontais representam a amplitude composta pelo tempo de atravessamento dos *cases* a cada 1000 minutos. Considerando-se que os *cases* estão uniformemente distribuídos no eixo das coordenadas, a maioria deles foi fechada em até 1000 minutos. Por outro lado, mesmo representando a minoria dos *cases*, as situações em que o tempo de atravessamento ultrapassa os 1000 minutos possuíram maior variabilidade, podendo alcançar 6000 minutos.

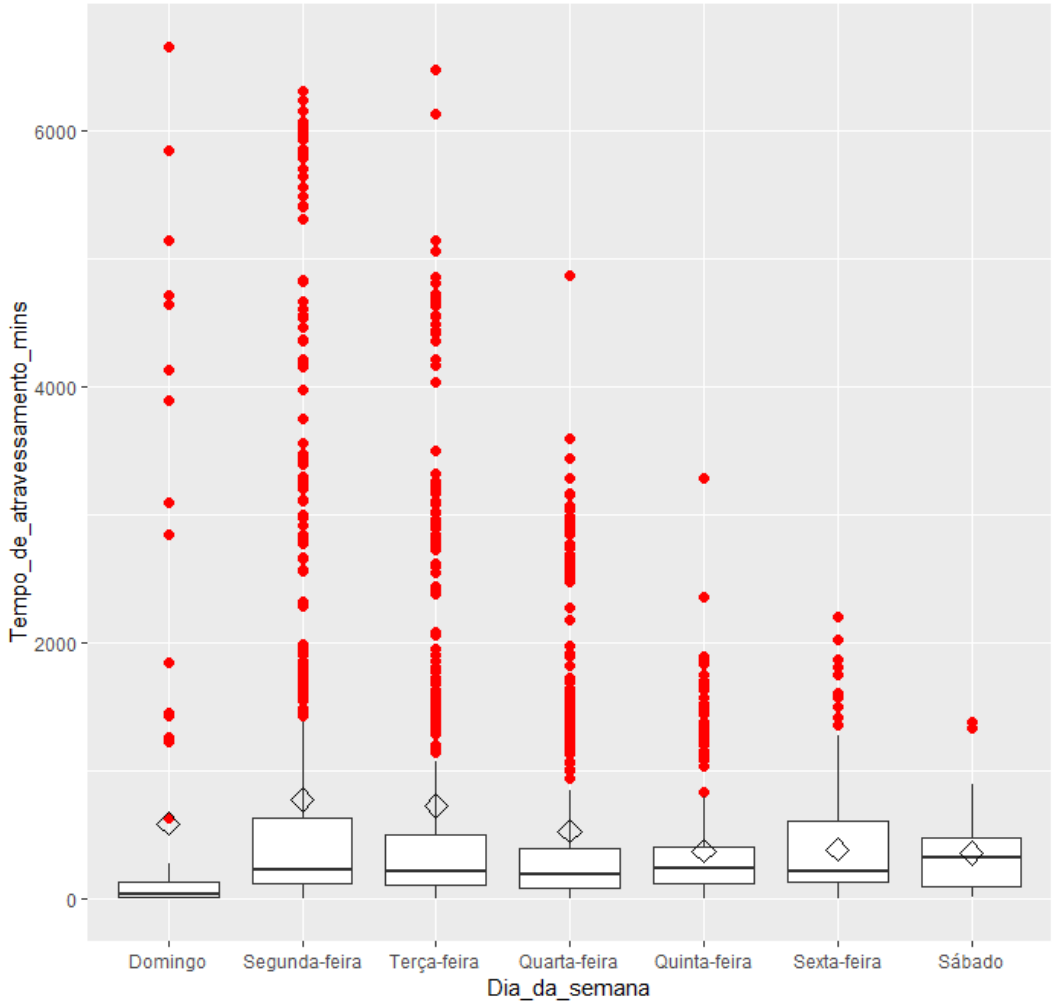
Figura 20 – *Cases* criados por dia



Fonte: Elaborado pelo autor.

A Figura 20 apresenta a criação dos *cases* ao longo dos dias da semana. Os eixos horizontais representam a amplitude definida pela quantidade de *cases* criados ao longo do dia. Considerando-se que o *event log* em questão possui dados de domingo a sábado, a demanda nesses dois dias se apresentou menor do que no restante da semana. Da mesma forma, o gráfico mostra que o volume de *cases* criados na segunda-feira e na sexta-feira foram maiores que nos outros dias. Assim, pode-se concluir que o dia da semana é uma variável que pode causar impacto na performance do *event log*.

Gráfico 4 – Tempo de atravessamento dos *cases* em função do dia de criação



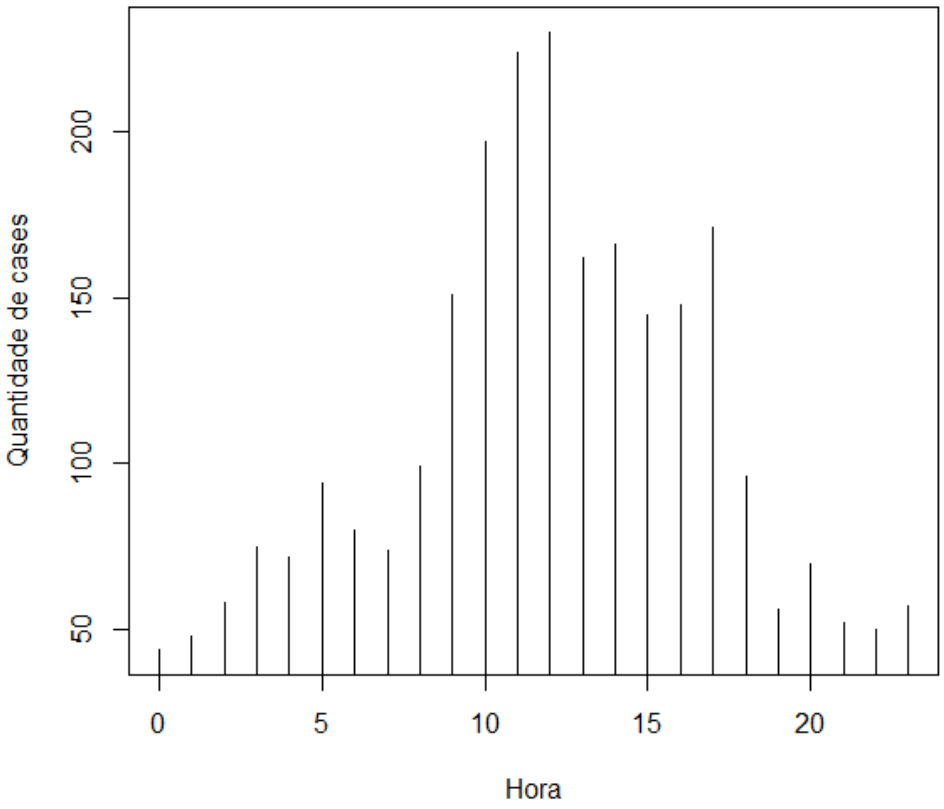
Fonte: Elaborado pelo autor.

O Gráfico 4 apresenta o dia de criação dos *cases* em função do tempo de atravessamento dos mesmos. Os losangos representam a média dos tempos e os pontos vermelhos os *outliers,* os quais tiveram os maiores tempos de atravessamento de domingo a terça-feira, mas apresentaram maior volume de segunda a quarta-feira. A maior presença de *outliers* aumentou a média do tempo de atravessamento e as maiores variabilidades foram apresentadas nos *cases* criados segunda e sexta-feira.

Os *cases* criados no início da semana podem influenciar na performance pelo resto dos dias, ao passo que, a partir de quarta-feira, o tempo máximo de atravessamento diminuiu gradativamente. Apesar das variabilidades terem sido diferentes ao longo dos dias, as medianas são praticamente as mesmas de segunda a sexta-feira. Assim, conclui-se que os *cases* criados ao longo da semana

demandaram tempos semelhantes e que o maior volume apresentado na segunda-feira, como mostra a Figura 20, influenciou na performance do resto da semana.

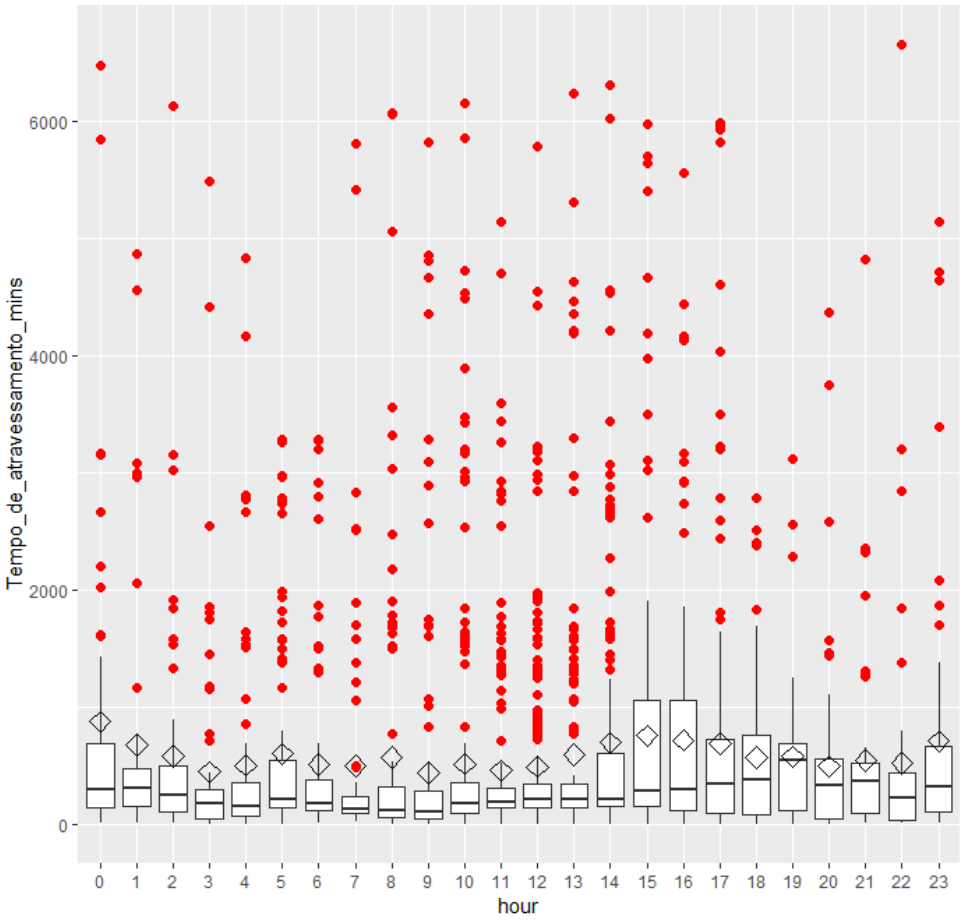
Gráfico 5 – *Cases* criados em função das horas



Fonte: Elaborado pelo autor.

O Gráfico 5 apresenta a quantidade de *cases* criados em função da hora de criação dos mesmos. De acordo com o *event log* analisado, a quantidade de *cases* criados das 9h às 17h foi superior ao resto do dia. Houve também um pico de demanda às 5h, o qual resultou em aproximadamente 100 *cases* criados. Considerando-se que a Figura 20 apresentou o volume de *cases* criados por dia e que o Gráfico 4 mostrou que tal volume impacta os tempos de atravessamento ao longo da semana, a quantidade de *cases* criados por hora pode afetar a performance do sistema.

Gráfico 6 – Tempo de atravessamento por hora



Fonte: Elaborado pelo autor.

O Gráfico 6 apresenta os tempos de atravessamento dos *cases* do *event log* de acordo com a hora em que foram criados. Da mesma forma que os picos de demanda impactaram a performance do processo nos dias posteriores, o maior volume de *cases* criados entre as 10h e às 12h aumentou a média e a variabilidade dos tempos de atravessamento a partir das 14h. Os *outliers* foram frequentes em todos os horários, mas as distâncias entre o segundo e o terceiro quartil, assim como as médias e medianas, só voltaram a diminuir entre as 6h e às 13h.

Analisando-se os resultados, conclui-se que o processo em questão possui grande variabilidade em decorrência do volume de *cases* criados entre as 9h e as 17h e, principalmente pela maior demanda apresentada às segundas-feiras. De acordo com os mapas do processo, a assinatura de *cases* para outros times especializados e a reabertura dos mesmos por parte dos clientes também são potenciais variáveis de impacto no que tange à performance.

Finalmente, pode-se concluir que, apesar dos diferentes caminhos percorridos no *event log*, muitas permutações de atividades representam as mesmas ações, o que mantem o tempo de atravessamento abaixo de um dia na maioria dos casos. Nas situações em que os *cases* demoram mais para ser fechados, o estoque em processamento pode afetar os tempos de atravessamento e de resposta.

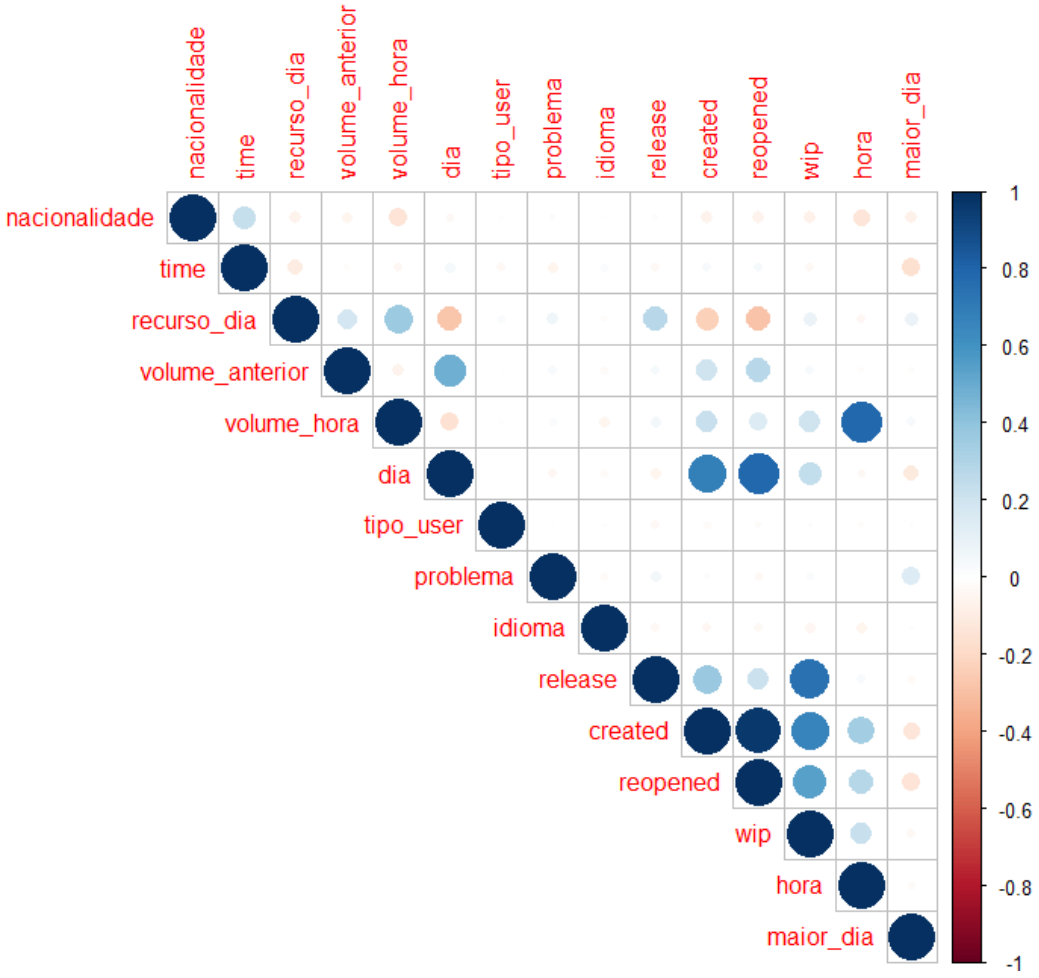
4.4 *Process Mining* – Modelos Preditivos

Esta seção apresenta os resultados da análise preditiva que, através de técnicas de classificação supervisionada, foram realizadas para avaliar o comportamento das variáveis no tocante à violação de SLA e *lead time* maior que um dia. Considerando-se as diferenças metodológicas entre a seleção das variáveis e as classificações, essas etapas estão apresentadas separadamente, de modo que a primeira é composta pela análise de correlação, pela aplicação do algoritmo *Boruta* e da função *Recursive Feature Selection*.

4.4.1 Seleção das variáveis

As variáveis coletadas para compor a seleção foram primeiramente submetidas à técnica de análise de correlação dos dados. Os atributos que apresentaram correlação positiva acima de 0,75 foram reavaliados para que apenas um permanecesse nas amostras.

Gráfico 7 – Gráfico de correlação



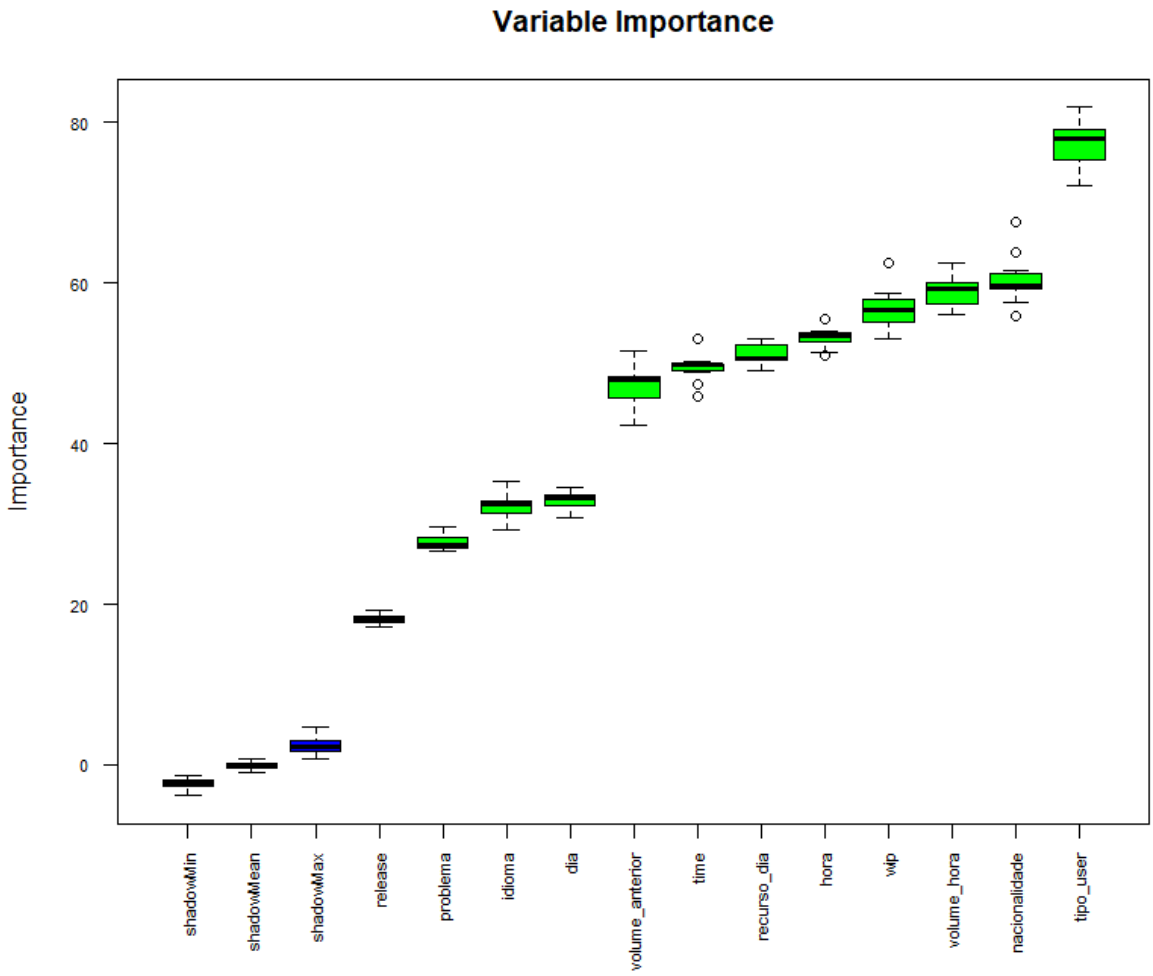
Fonte: Elaborado pelo autor.

O Gráfico 7 mostra, em uma escala de -1 a 1, as correlações entre todas as variáveis coletadas. Foi utilizada a função *cor()* da plataforma R e os atributos dia, *created* e *reopened* foram apontados como correlacionados. O dia da semana não possui forte correlação com o número de *cases* criados, mas apresentou um coeficiente de 0,82 em relação à quantidade de *cases* reabertos nos últimos 4 dias. Já as variáveis *created* e *reopened* são correlacionadas com um coeficiente de 0,97.

Considerando-se que as os aspectos *created* e *reopened* foram extraídos para compor a variável *wip*, estes foram os escolhidos para serem descartados pela análise de variáveis correlacionadas. O atributo *dia* permanecerá na análise em decorrência do possível impacto que esse pode ter no atendimento ao SLA e no tempo de atravessamento dos *cases*, como evidenciado no Gráfico 4 e na Figura 20 (seção 4.2).

Removidas as variáveis fortemente correlacionadas, os dados começaram a ser analisado separadamente à luz do objetivo das predições: contato com o cliente em até 120 minutos desde a criação do *case* e fechamento do mesmo em até 1440 minutos. Os dois cenários analisados possuem implicações gerenciais distintas, portanto, as variáveis que impactam o atendimento inicial podem não ser as mesmas relevantes para predizer se um *case* demorará mais de um dia para ser fechado. Sendo assim, as análises a seguir consideraram ambas as situações e foram baseadas em *Random Forests*, mas possuem focos diferentes.

Gráfico 8 – Importância das variáveis - IRT

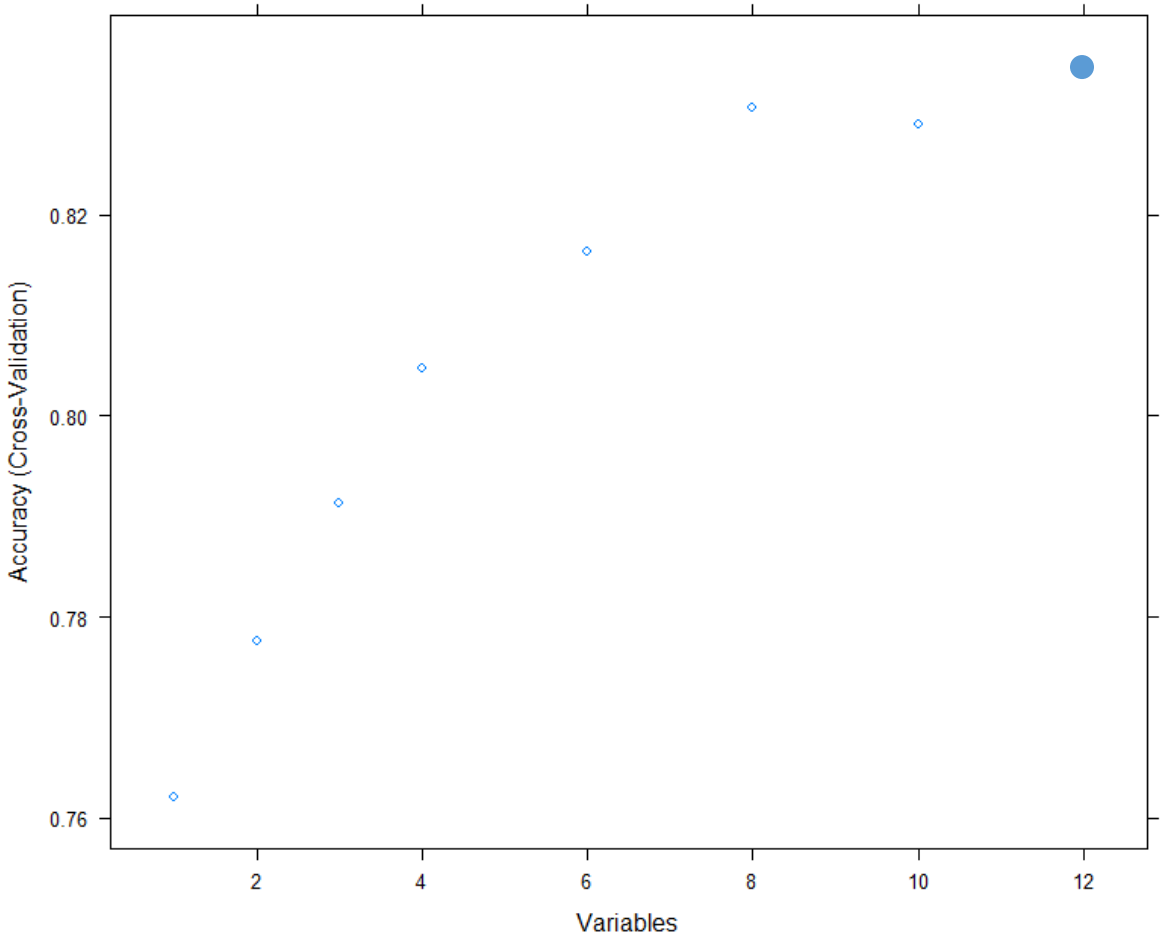


Fonte: Elaborado pelo autor.

Analisando-se o Gráfico 8, é possível concluir que todas as variáveis possuíram importância maior do que as sombras criadas pelo algoritmo (*shadowMin*, *shadowMean* e *shadowMax*). Isso significa que não se pode rejeitar a hipótese de que todas as variáveis são importantes para classificar o IRT, uma vez que possuíram

valor-p menor do que 0,1. Além disso, a variável *tipo_user*, a qual representa o tipo do usuário que está contatando o suporte, apresentou uma média de importância de 77,31. Antagonicamente, a variável *release* apresentou média de importância igual a 18,19, sendo quase 4 vezes menos do que o tipo do usuário.

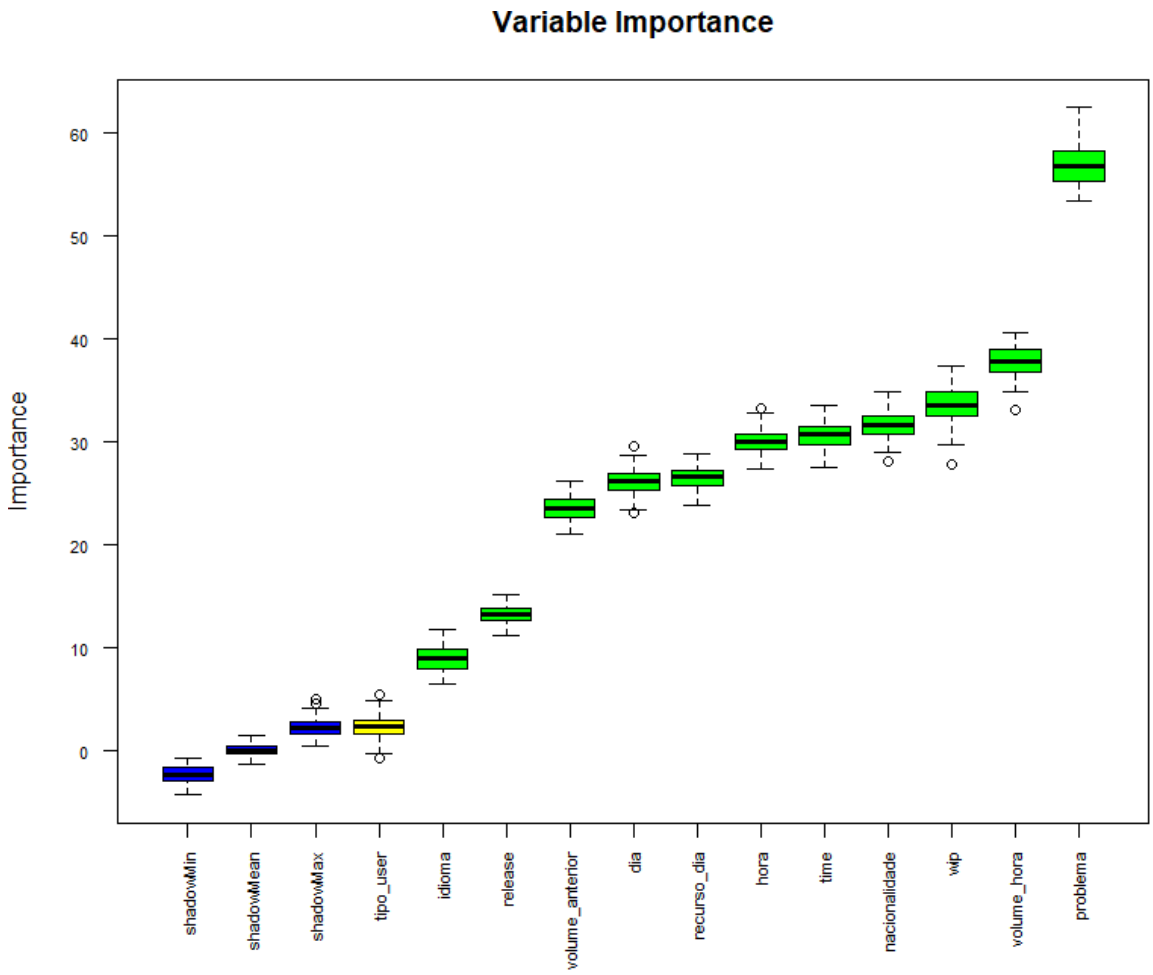
Gráfico 9 – *Recursive feature selection* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 9 apresenta a evolução da acurácia de um modelo de *Random Forests* ao passo que novas variáveis foram sendo agregadas. Aplicando-se 10-*fold cross-validation*, pode-se concluir que os menores erros ocorreram ao se utilizarem todas as variáveis descorrelacionadas. A acurácia de tal seleção foi de 83,47% e o coeficiente *Kappa* foi 0,5022 para 12 atributos de entrada. Dessa forma, após a análise de correlação e não poder ser rejeitada a hipótese de que todas as variáveis são significantes para a classificação de violação do SLA, o modelo preditivo para tal fim será composto por 12 variáveis.

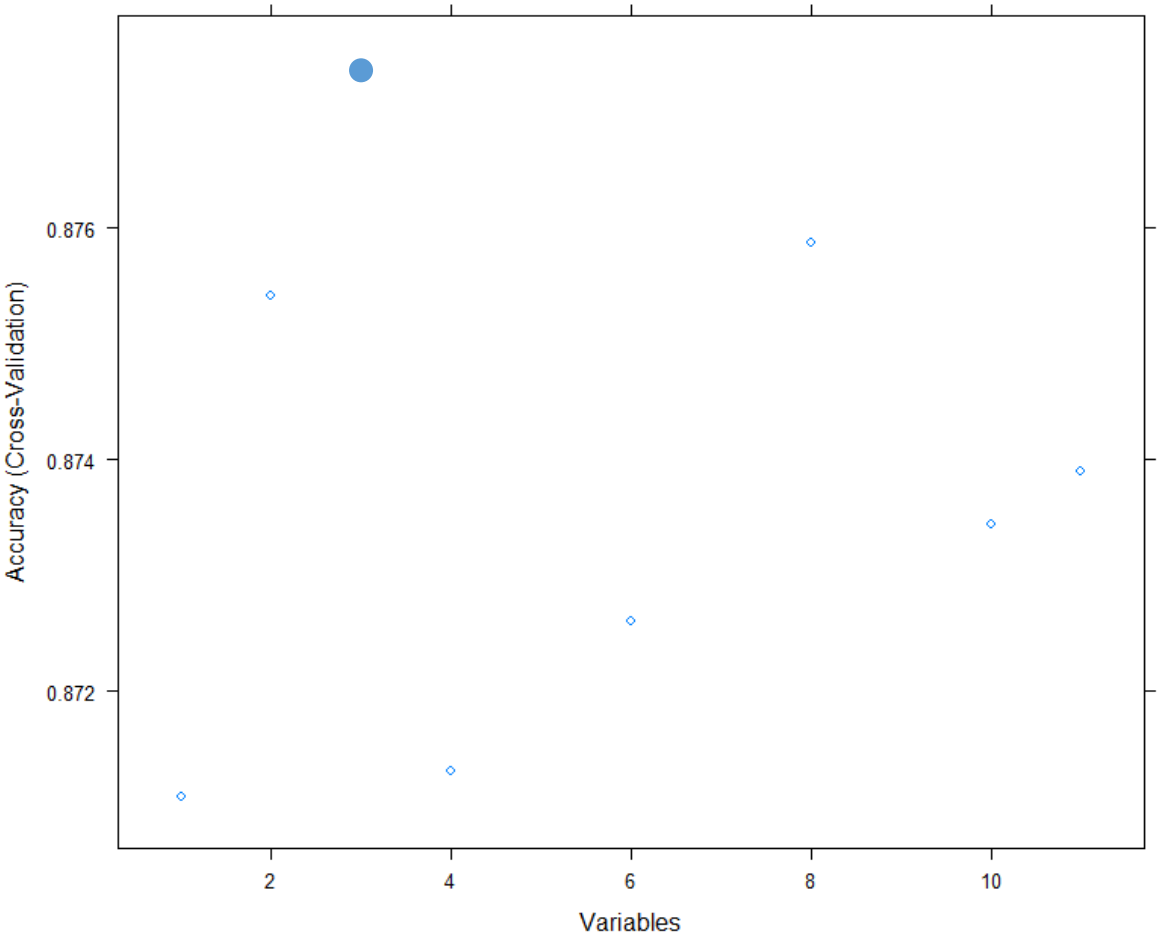
Gráfico 10 – Importância das variáveis - fechamento



Fonte: Elaborado pelo autor.

O Gráfico 10 mostra os resultados do teste de hipótese de 12 atributos em função do tempo de fechamento dos *cases* ser maior que um dia. Ao contrário do cenário anterior, a variável *tipo_user* possui uma média de importância de 2,26, sendo classificada como *Tentative* pois pode ser confundida com as sombras criadas pelo algoritmo. O problema foi isoladamente o atributo com menor valor-p, assumindo uma média de importância de 56,92.

Gráfico 11 – *Recursive feature selection* - fechamento



Fonte: Elaborado pelo autor.

Tendo em vista que a variável *tipo_user* pode não ser importante para a predição do tempo de fechamento dos *cases*, o Gráfico 11 apresenta a aplicação da técnica *recursive feature selection*. Os resultados dos algoritmos também foram avaliados através de uma 10-*fold cross-validation* e o modelo com menor erro possuiu 3 variáveis de entrada, sendo elas o problema do cliente, o país do atendente (nacionalidade) e o *wip*.

4.4.2 Classificação da violação do SLA

A violação do SLA é verificada quando a resposta inicial ao cliente é enviada após 120 minutos da criação do *case*. Dessa maneira, os resultados dos dois algoritmos utilizados para classificar tal comportamento possuem o objetivo de apontar as variáveis que mais o explicam.

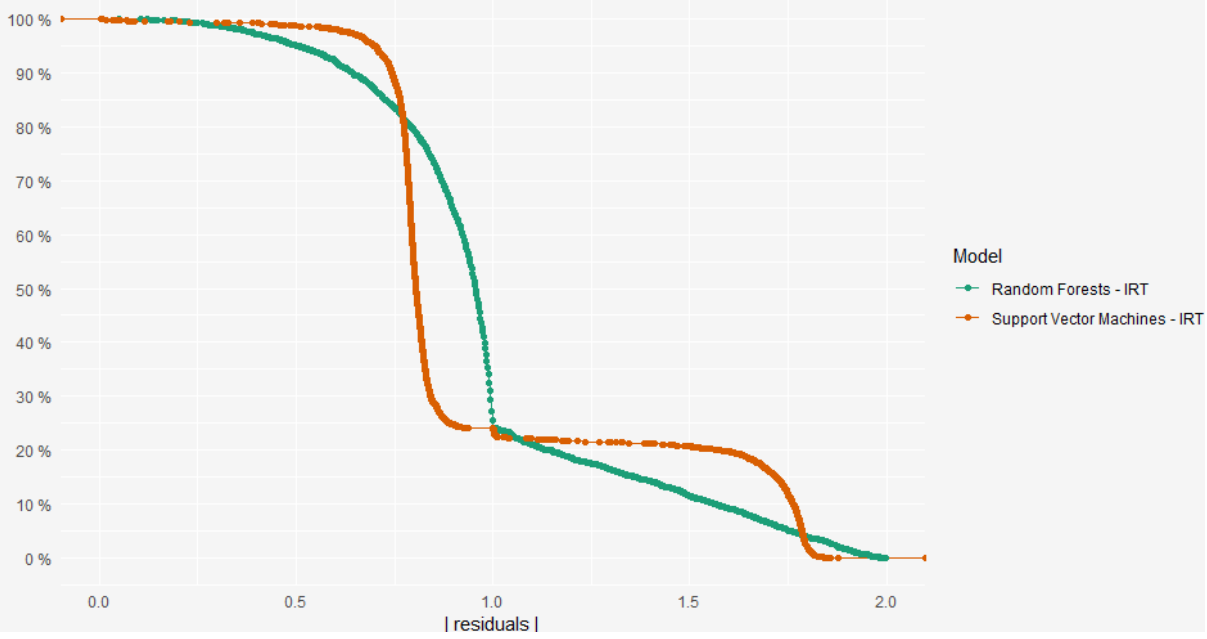
Tabela 19 – Matrizes de confusão do IRT

Algoritmo	**Correto x0**	**Correto x1**	**Incorreto x0**	**Incorreto x1**	**Acurácia**	**Intervalo de confiança (95%)**	***Kappa***
Random Forests	2341	412	160	380	83,6%	(0.8229, 0.8485)	0,5041
SVM	2459	109	42	683	77,98%	(0.7653, 0.7939)	0,167

Fonte: Elaborado pelo autor.

A Tabela 19 apresenta os resultados das matrizes de confusão dos algoritmos utilizados para classificar as respostas iniciais. A legenda “x0” significa que o *case* não violou o SLA e “x1” que o mesmo foi violado. Dito isso, conclui-se que o *Random Forests* foi o único capaz de predizer mais violações corretamente do que falsos negativos, com um coeficiente *kappa* de 0,5041, demonstrando concordância moderada. Antagonicamente, o SVM se mostrou preciso na predição dos *cases* que não violariam o SLA, mas falhou mais na recíproca contrária.

Gráfico 12 – Erros residuais dos algoritmos - IRT

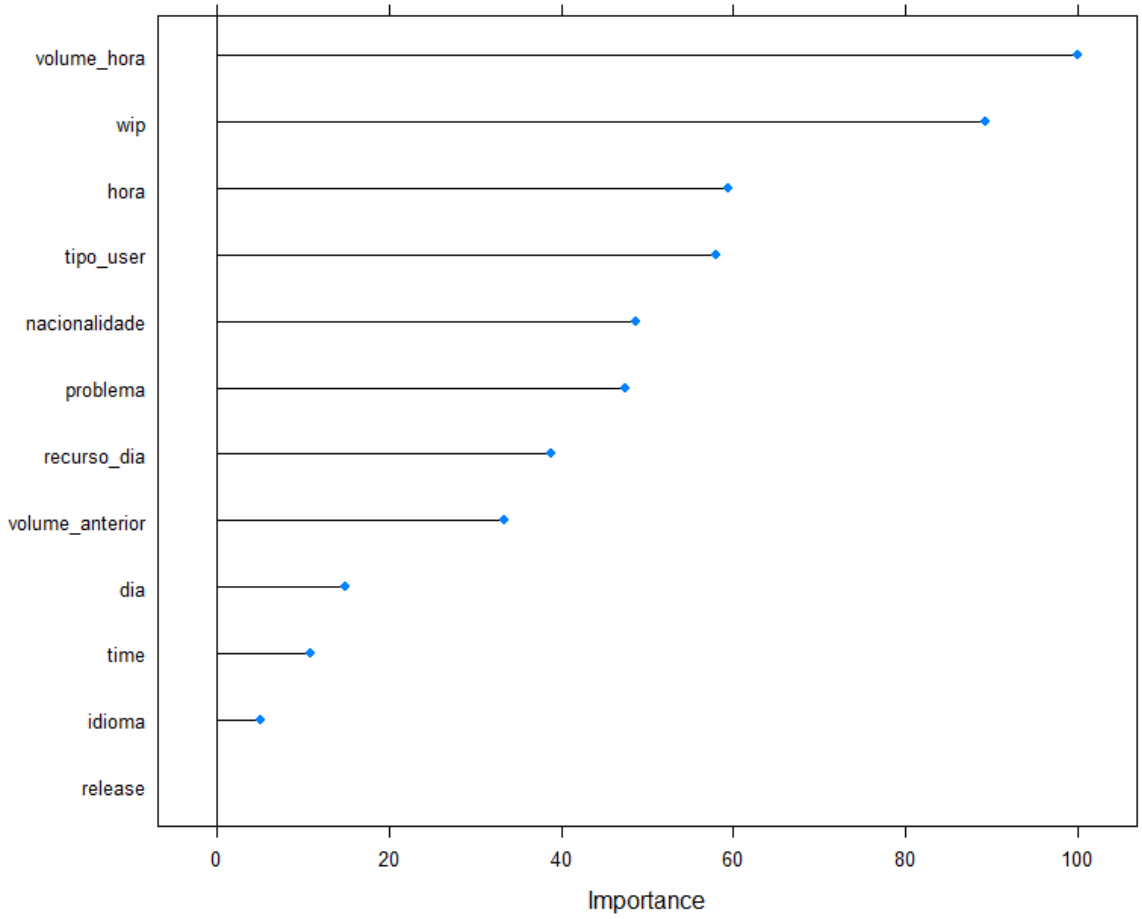


Fonte: Elaborado pelo autor.

O Gráfico 12 apresenta a evolução dos erros dos modelos ao longo da trajetória para predizer a violação do SLA. Em decorrência das classes utilizadas como variáveis de resposta, a escala do eixo x vai de 0 a 2, sendo o intervalo (0,1) os erros ao predizer cada *case* que atendeu o SLA. Consequentemente, os caminhos percorridos de 1 a 2 correspondem aos erros cometidos ao classificar um *case* como violador de SLA.

Sendo as curvas do gráfico a representação da aderência dos modelos ao predizer determinadas situações, pode-se concluir que o algoritmo SVM errou menos ao classificar as observações em conformidade com o tempo de resposta requerido, uma vez que seus caminhos aderiram mais rapidamente à tal alternativa. Por outro lado, o *Random Forests* obteve melhor precisão ao predizer a violação do SLA, tendo performance quase linear.

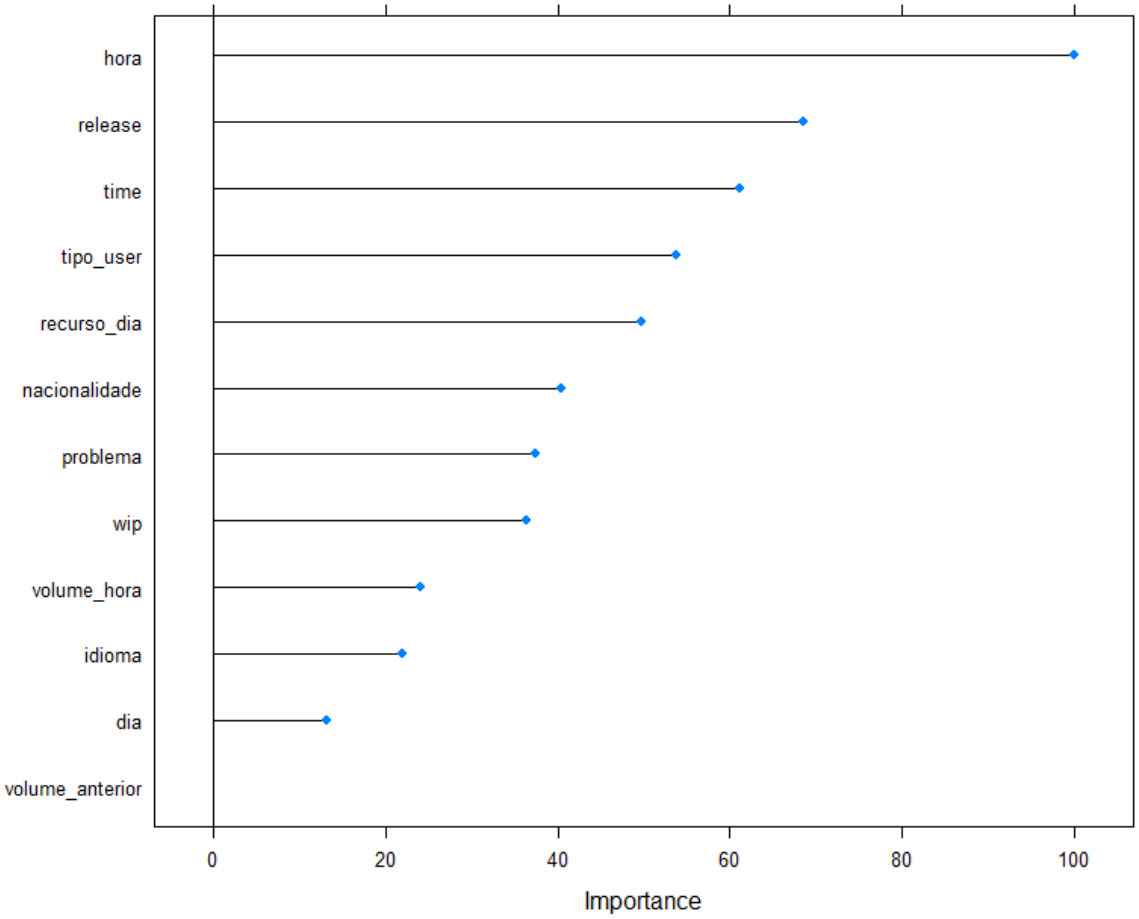
Gráfico 13 – Importância relativa das variáveis no *Random Forests* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 13 apresenta a utilização relativa das variáveis no cálculo da classificação de violação do SLA. Visualmente, destacam-se os atributos *volume_hora*, *wip*, *hora* e *tipo_user*, demonstrando mais cálculos com atributos ligados ao volume, à hora do dia e ao tipo de usuário que o algoritmo teve. A nacionalidade do atendente e o tipo de problema também foram utilizados com frequência para composição da variável de resposta. Por outro lado, a ocorrência de *release* foi descartada pelo modelo.

Gráfico 14 – Importância relativa das variáveis no SVM - IRT

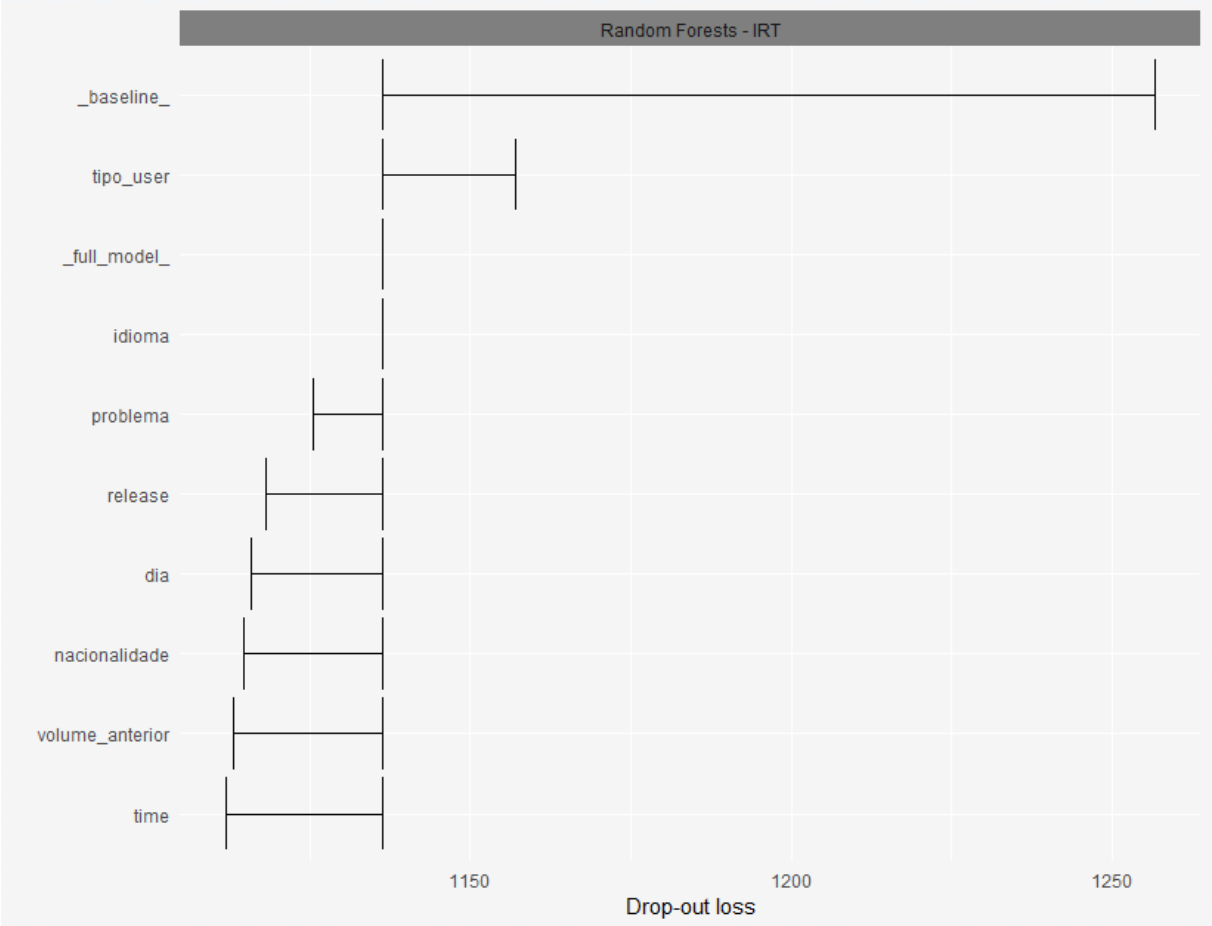


Fonte: Elaborado pelo autor.

O Gráfico 14 mostra a utilização das variáveis nos cálculos para predizer a violação do SLA com o algoritmo SVM. O atributo *hora* foi o mais utilizado, seguido do *release*, do *time* e do *tipo_user*. Tal comportamento pode ser explicado pela menor sensibilidade do algoritmo às mudanças de valores, fazendo com que classes sejam mais consideradas do que variáveis contínuas. Nesse caso, o *volume_anterior*, que

corresponde à quantidade de *cases* criados no dia anterior, não foi utilizado pelo algoritmo.

Gráfico 15 – Importância das variáveis no *Random Forests* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 15 apresenta a importância das variáveis em função das perdas de performance do modelo *Random Forests* caso determinado atributo seja retirado do modelo. Tal análise foi gerada pela função *variable_importance* com todos os argumentos *default*. Foram geradas duas variáveis auxiliares, *full_model* e *baseline*, que significam, respectivamente, a perda de referência se todo o modelo fosse desconsiderado e a soma de todas as outras.

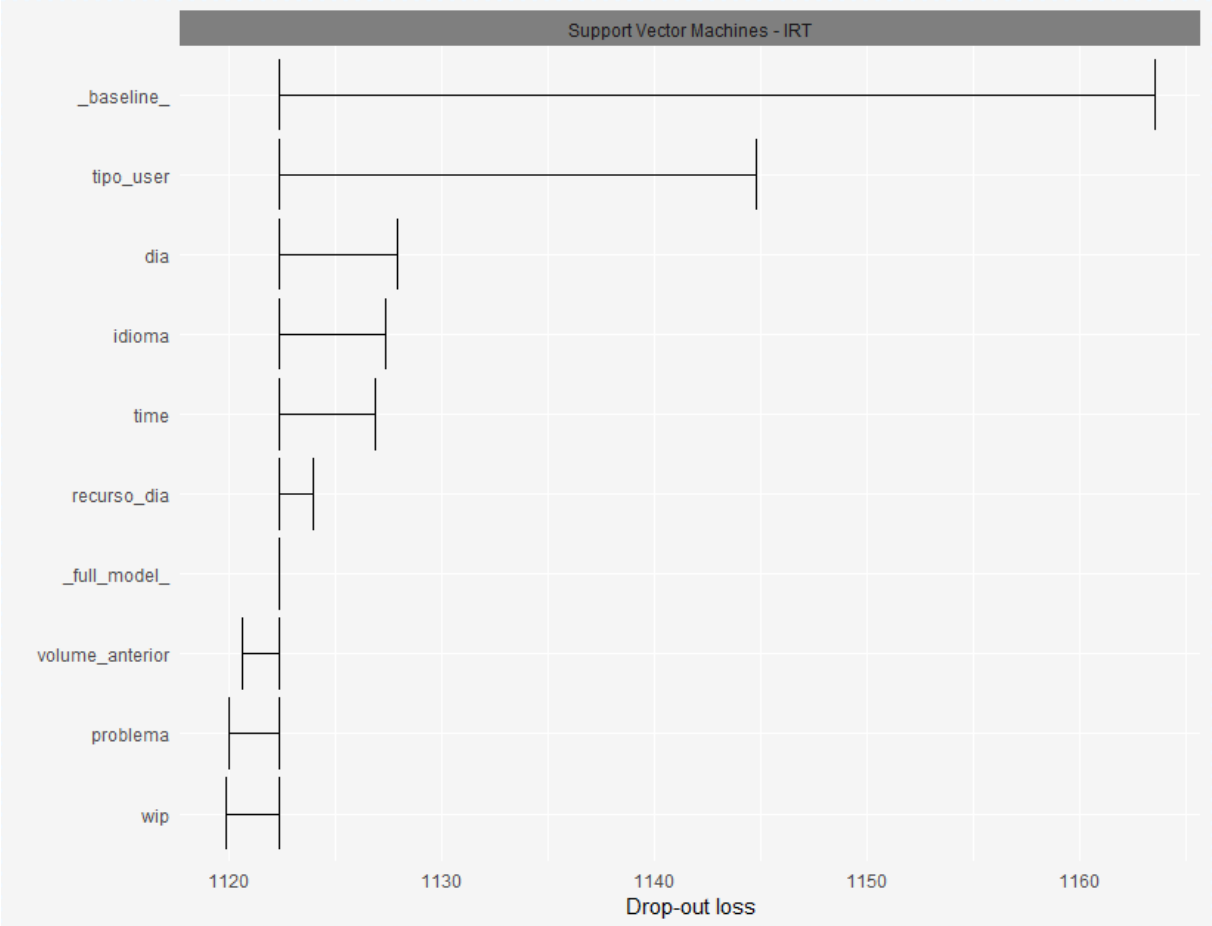
Tabela 20 – Perdas absolutas de performance no *Random Forests* – IRT

Variável	**Perdas com base no erro quadrático médio**
Tipo_user	1157,07
Idioma	1136,31
Problema	1125,44
Release	1118,23
Dia	1115,75
Nacionalidade	1114,59
Volume_anterior	1112,95
Time	1112,00
Recurso_dia	1106,32
Hora	1091,57
Volume_hora	1078,66
wip	1075,68

Fonte: Elaborado pelo autor.

Tomando-se o *full_model* como ponto central, a variável *tipo_user* seria a única mais importante do que a combinação de todo o modelo. Isso significa que todas as outras entradas foram consideradas individualmente menos relevantes para a predição da violação do SLA do que a combinação das variáveis de entrada. Como mostrado na Tabela 20, a única variável numérica que, ao ser retirada do modelo, estaria entre as dez que mais impactam as classificações foi o *volume_anterior*, que representa a quantidade de *cases* criados no dia anterior.

Gráfico 16 – Importância das variáveis no SVM - IRT



Fonte: Elaborado pelo autor.

O Gráfico 16 apresenta as perdas do modelo SVM através dos erros quadráticos médios. Considerando-se o modelo completo como referencial, as variáveis *tipo_user*, *dia*, *idioma*, *time* e *recurso_dia* afetariam individualmente mais a performance das classificações do que a combinação de todas as outras. Nesse algoritmo o atributo *volume_anterior* também foi a variável numérica mais importante para as predições.

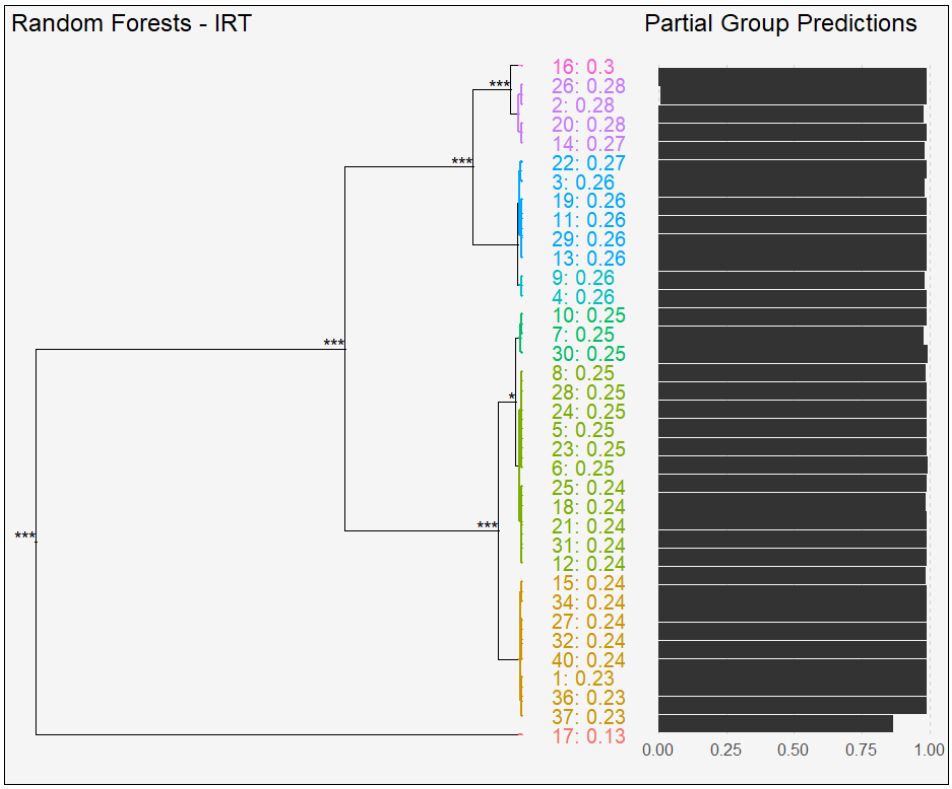
Tabela 21 – Perdas absolutas de performance no SVM – IRT

Variável	Perdas com base no erro quadrático médio
Tipo_user	1144,77
Dia	1127,93
Idioma	1127,34
Time	1126,86
Recurso_dia	1124,00
Volume_anterior	1120,67
Problema	1120,04
wip	1119,91
Release	1119,64
Volume_hora	1116,58
Nacionalidade	1115,17
hora	1104,94

Fonte: Elaborado pelo autor.

A Tabela 21 apresenta os valores absolutos das possíveis perdas que cada variável poderia causar na performance do modelo SVM. Nesse modelo, a amplitude de tais perdas é 51,06% menor do que no anterior, evidenciando o fato de que o algoritmo SVM é menos sensível às mudanças e tem mais dificuldade do que o *Random Forests* para aderir às oscilações das variáveis.

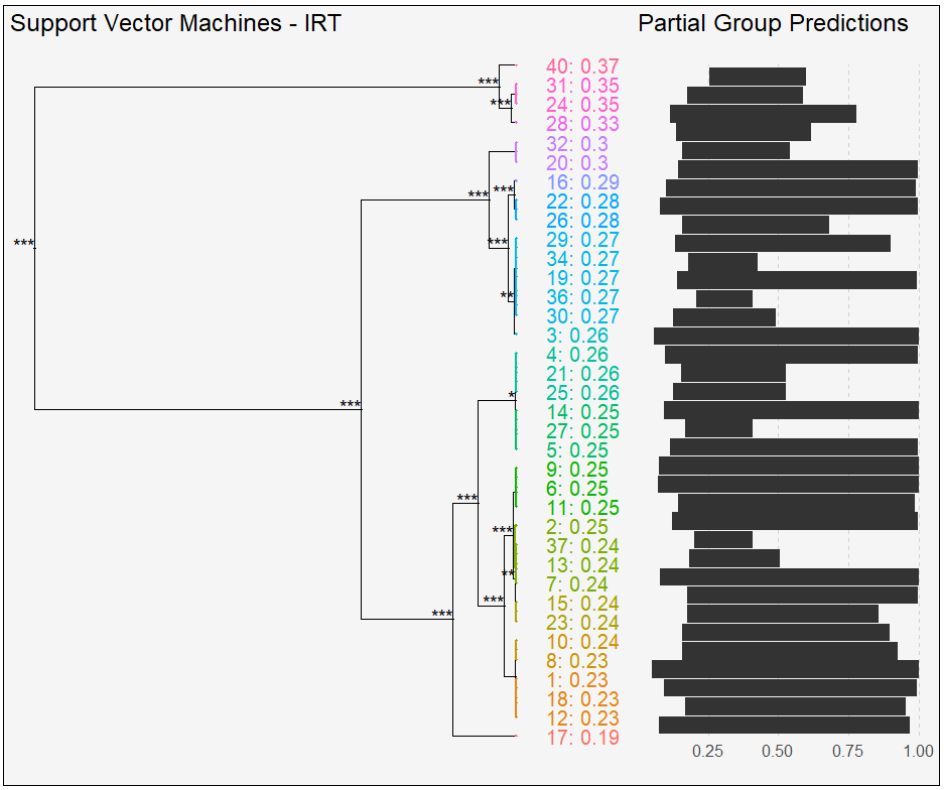
Gráfico 17 – Comportamento da variável *problema* - *Random Forests* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 17 mostra o agrupamento dos 40 tipos de problema que foram considerados pelo algoritmo. Foram formados 8 grupos de acordo com a contribuição relativa para compor a classificação final (taxa de sucesso). A coluna *Partial Group Predictions* corresponde à capacidade de cada problema em diferenciar a violação do SLA (1.00) da não violação (0.00). A maior taxa de sucesso foi de 30% e todos os problemas compuseram a variável de resposta de maneira semelhante, mostrando dificuldades ao discriminar o resultado. Excepcionalmente, o problema 17 apresentou maior capacidade discriminatória mas com baixa taxa de sucesso de 13%, o que indica a predominância de falsos positivos.

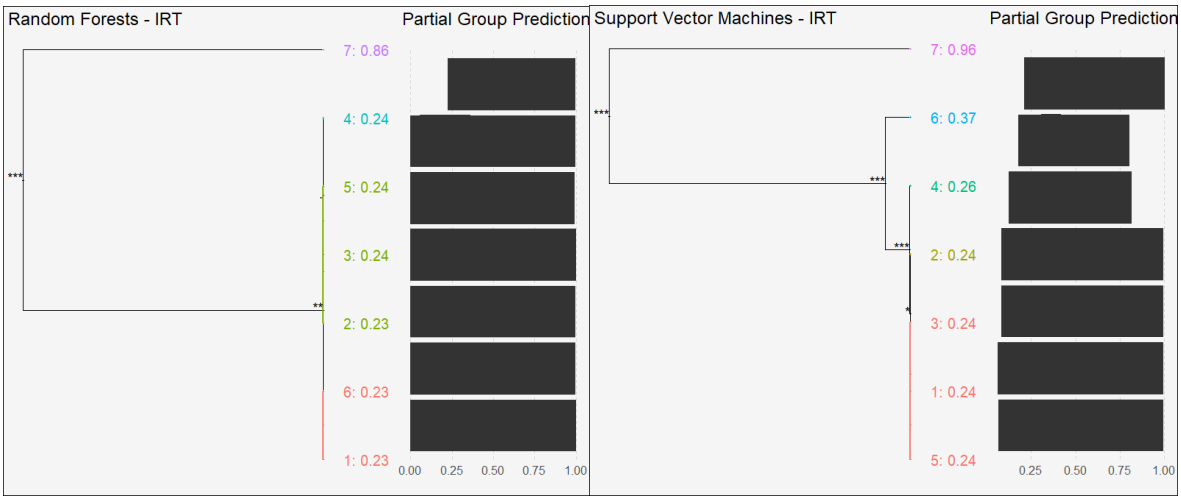
Gráfico 18 – Comportamento da variável *problema* – SVM – IRT



Fonte: Elaborado pelo autor.

O Gráfico 18 apresenta o comportamento da variável *problema* ao predizer a violação do SLA com o algoritmo SVM. Visualmente, pode-se concluir que os tipos de problema foram importantes para as classificações do modelo, visto que houveram várias entradas capazes de discriminar os valores 0.00 e 1.00. Foram formados 17 grupos e o problema que menor taxa de sucesso também foi o 17. Antagonicamente ao *Random Forests*, no qual a entrada 40 foi uma das piores classificadoras, o SVM obteve taxa de sucesso de 37% ao utilizar tal variável para a predição.

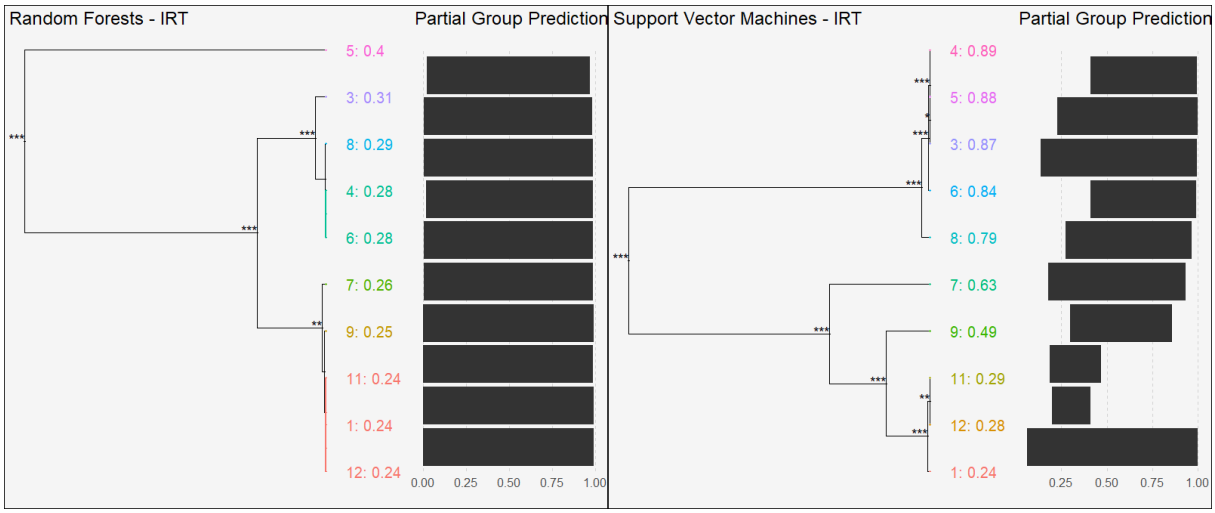
Gráfico 19 – Comportamento da variável *tipo_user* – IRT



Fonte: Elaborado pelo autor.

O Gráfico 19 apresenta um dendograma da variável *tipo_user* ao classificar a violação do SLA. O *Random Forests* formou 4 grupos e o tipo de usuário 7 obteve maior taxa de sucesso, já que os outros tipos de usuários foram pouco capazes de distinguir as classes. No SVM, foram formados 5 grupos e as entradas 7, 6 e 4 foram as mais determinantes para que o modelo classificasse a resposta. Assim como no outro algoritmo, os outros tipos de problema não foram capazes de distinguir a violação do SLA.

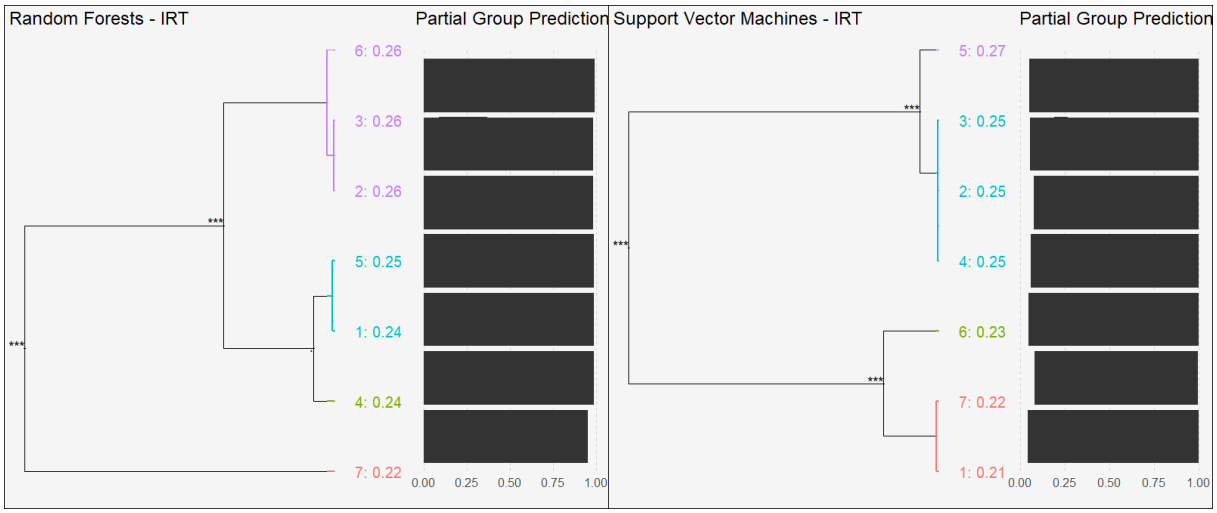
Gráfico 20 – Comportamento da variável *idioma* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 20 mostra o comportamento da variável *idioma* na composição das classificações. O *Random Forests* agrupou as entradas em 7 categorias e apenas o idioma 5 apresentou uma separação mais clara entre os valores 0.00 e 1.00, evidenciando a razão pela qual a variável foi considerada importante, uma vez que tal entrada se diferencia das outras. O SVM criou 10 grupos e utilizou os idiomas de forma mais discricionária que o outro algoritmo, de modo que apenas as entradas 1, 7, 9 11 e 12 apresentaram taxa de sucesso abaixo de 70%.

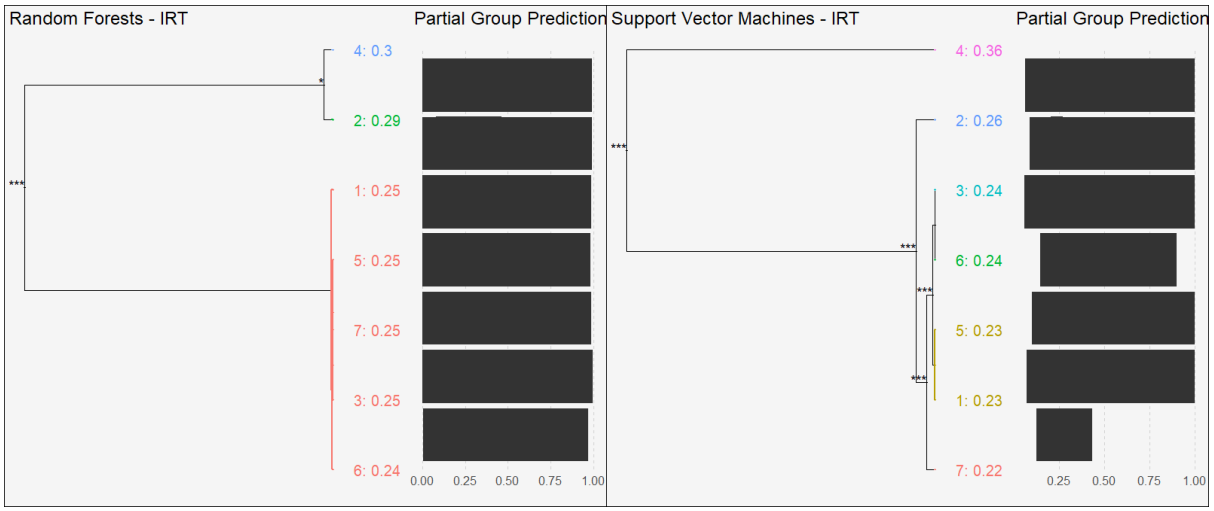
Gráfico 21 – Comportamento da variável *dia* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 21 mostra o dendograma da variável *dia* ao classificar a violação do SLA. A performance de ambos os modelos foi semelhante, com 4 grupos criados e baixa capacidade dos algoritmos em separar os *cases* que violaram ou não o tempo de resposta acordado. Tal situação também pode ser evidenciada pela menor amplitude que os modelos apresentaram e pela baixas taxas de sucesso de 26% com o *Random Forests* e 27% com o SVM

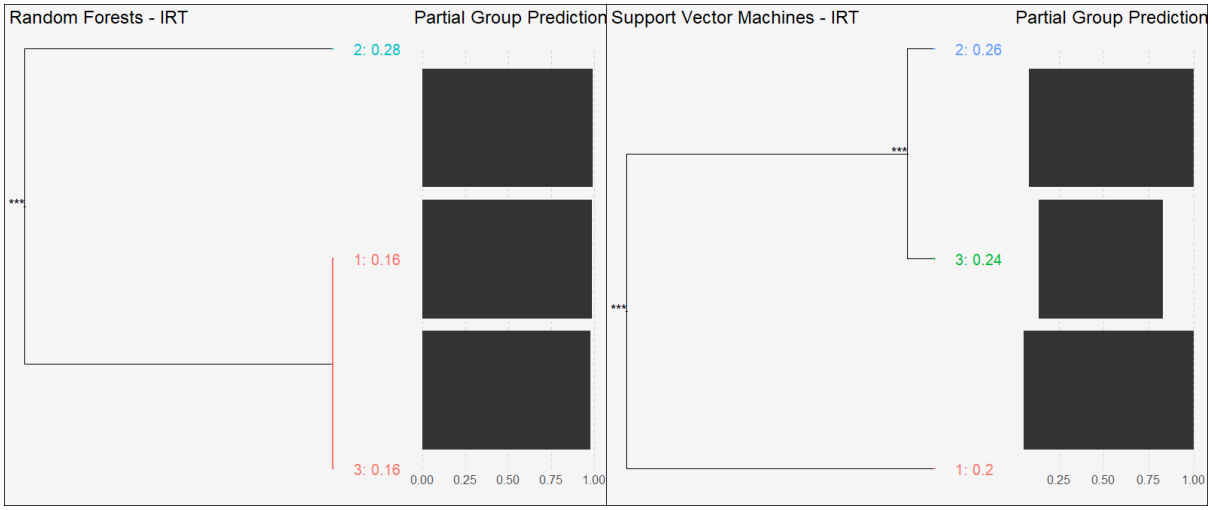
Gráfico 22 – Comportamento da variável *nacionalidade* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 22 apresenta a contribuição da variável *nacionalidade* na composição das respostas finais dos modelos. O *Random Forests* formou 4 grupos e a maior taxa de sucesso foi 0,3, mas o *Partial Group Prediction* do algoritmo evidencia a dificuldade que o mesmo tem em distinguir as duas classes. O SVM formou 7 grupos e a nacionalidade 4 obteve taxa de sucesso 0,36, levemente mais alta que no modelo anterior. Entretanto, as outras entradas do SVM não obtiveram bons resultados, evidenciando que a capacidade discricionária apresentada gerou muitas predições incorretas.

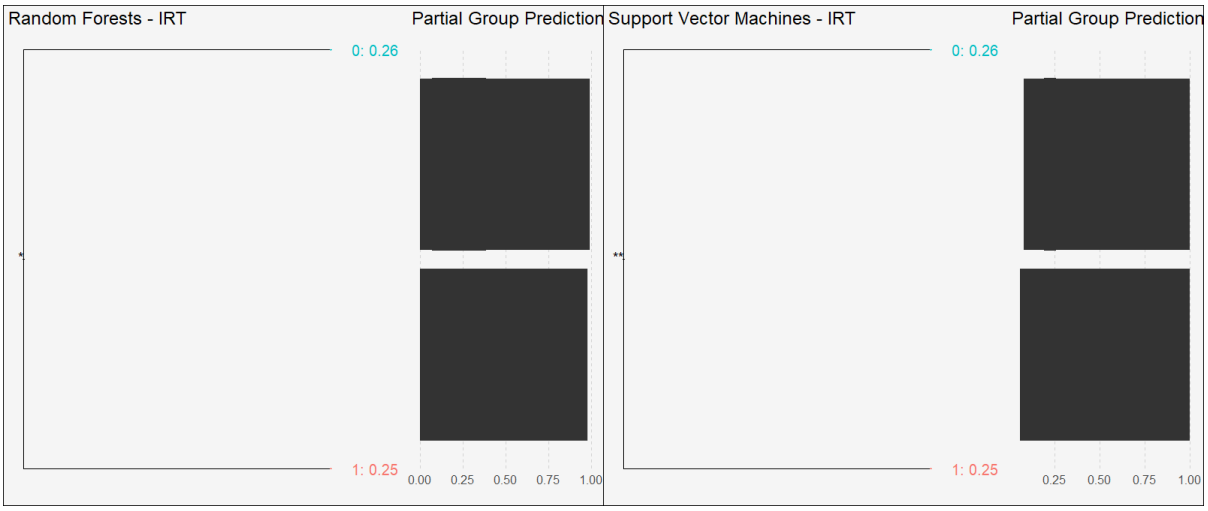
Gráfico 23 – Comportamento da variável *time* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 23 apresenta as taxas de sucesso da variável *time* ao predizer a violação do SLA. O algoritmo *Random Forests* dividiu as entradas em 2 grupos e o time 2 obteve o maior sucesso, com 0,28. Mesmo com tal resultado, o *Partial Group Prediction* evidenciou que não ouve muita distinção entre os valores 0.00 e 1.00. O SVM obteve taxas de sucesso predominantemente mais altas e, considerando-se que o time 3 foi considerado discricionário mas classificou apenas 24% das observações corretamente, pode-se concluir que o algoritmo interpretou a entrada incorretamente.

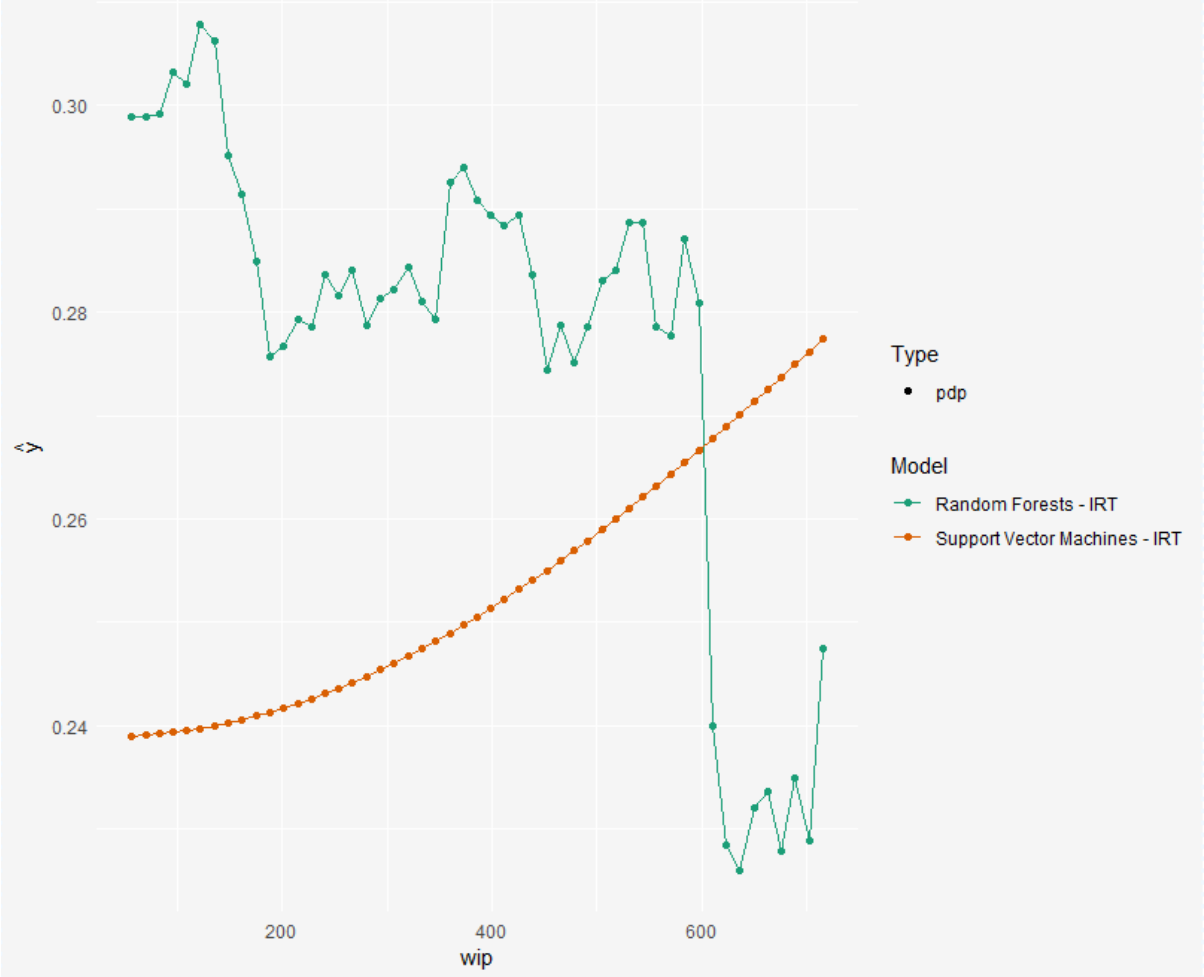
Gráfico 24 – Comportamento da variável *release* – IRT



Fonte: Elaborado pelo autor.

O Gráfico 24 mostra como a variável *release* se comportou ao classificar as violações do SLA. Ambos os modelos tiveram taxas de sucesso idênticas e foram pouco capazes de distinguir os valores 0.00 e 1.00. Consequentemente, a variável *release* classificou muitas observações incorretamente uma vez que foi muito utilizada e importante no algoritmo SVM (Gráfico 14 e Gráfico 16).

Gráfico 25 – Dependência da variável *wip* - IRT



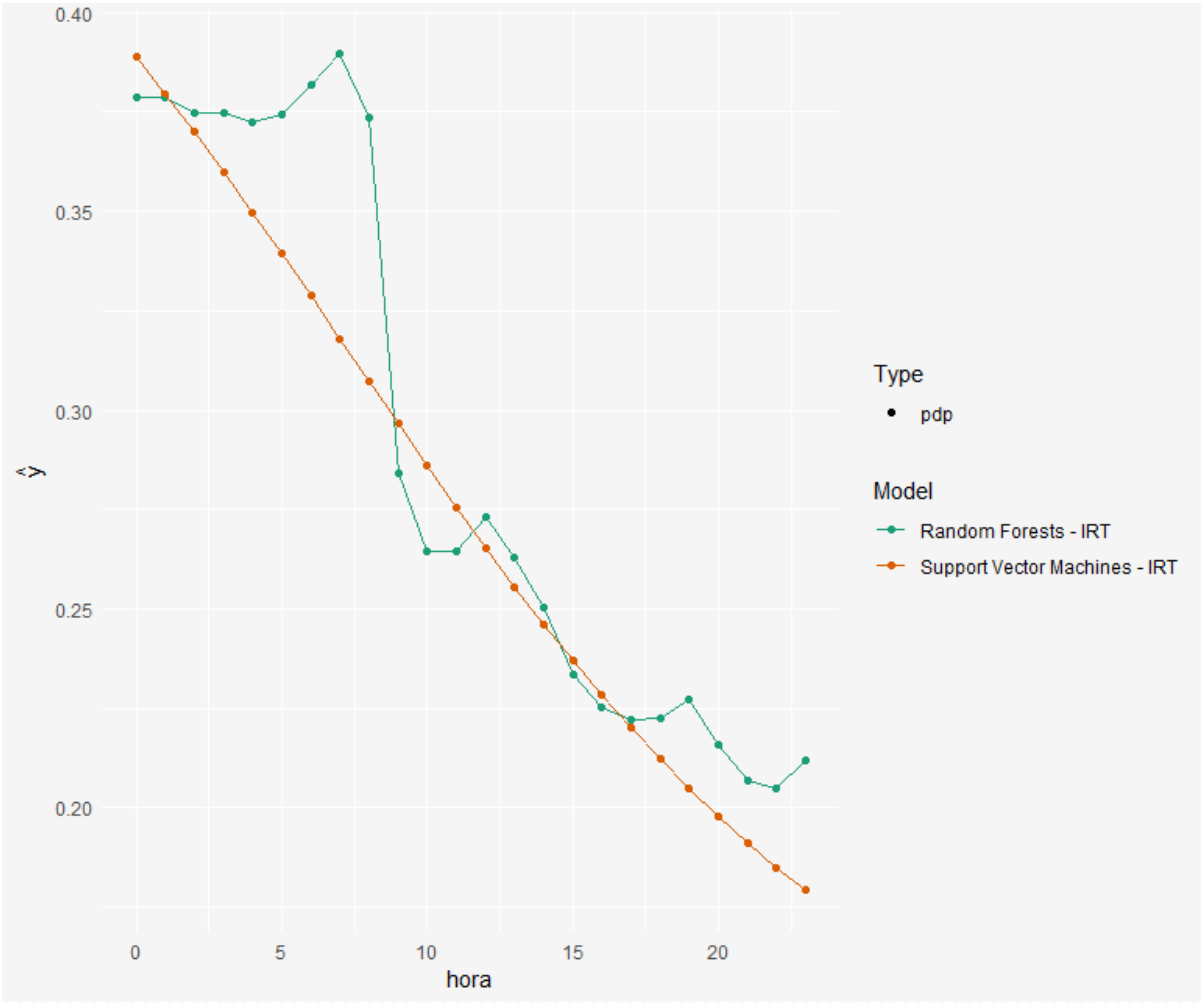
Fonte: Elaborado pelo autor.

O Gráfico 25 apresenta a variação da dependência da variável de resposta em função do *wip*. Para fins de interpretação, a variável em questão é mais importante para as predições à medida que o valor de y aumenta. O algoritmo *Random Forests* identificou que quando há em torno de 100 *cases* em processamento, a dependência da classificação aumenta em relação à variável *wip*, ao passo que entre 200 e 600 *cases* ela é menor e acima de 600 é quase nula. Isso mostra que se a demanda aumenta muito, o resultado final tende a depender cada vez menos dela.

O algoritmo SVM apresentou comportamento diferente do *Random Forests*. Ele precisou de mais *cases* para adequar as curvas e, ao aprender que a partir de 200 *cases* a dependência da resposta é maior do que o patamar que estava sendo assumido, adotou valores maiores de y até se cruzar com o *Random Forests*. Dada a

maior dificuldade de ajuste às variações contínuas do SVM, a curva não acompanhou o movimento descendente do outro algoritmo a partir dos 600 *cases*.

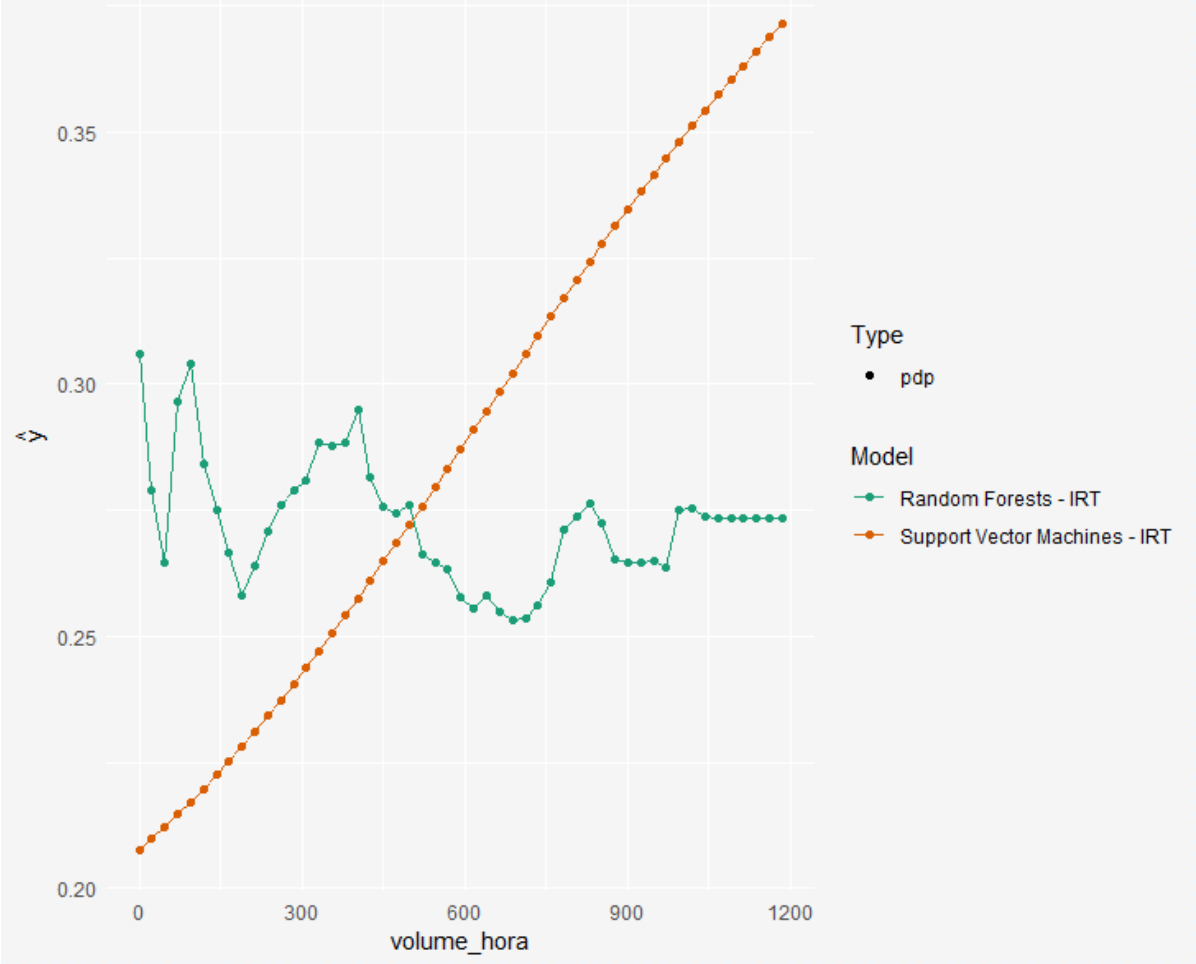
Gráfico 26 – Dependência da variável *hora* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 26 mostra a dependência da classificação da violação do SLA em função da hora em que um *case* foi aberto. Os algoritmos se comportaram de maneira semelhante e a variável *hora* tende a perder importância nas predições ao longo do dia. Até em torno das 8 horas da manhã, o *Random Forests* atribuiu à classificação em questão mais de 0,35 de importância do horário de criação do *case*. A partir desse ponto, os dois algoritmos interpretam que a dependência diminui muito, pois a hora já não é mais relevante para identificar a violação do SLA quando ela assume valores maiores do que 8.

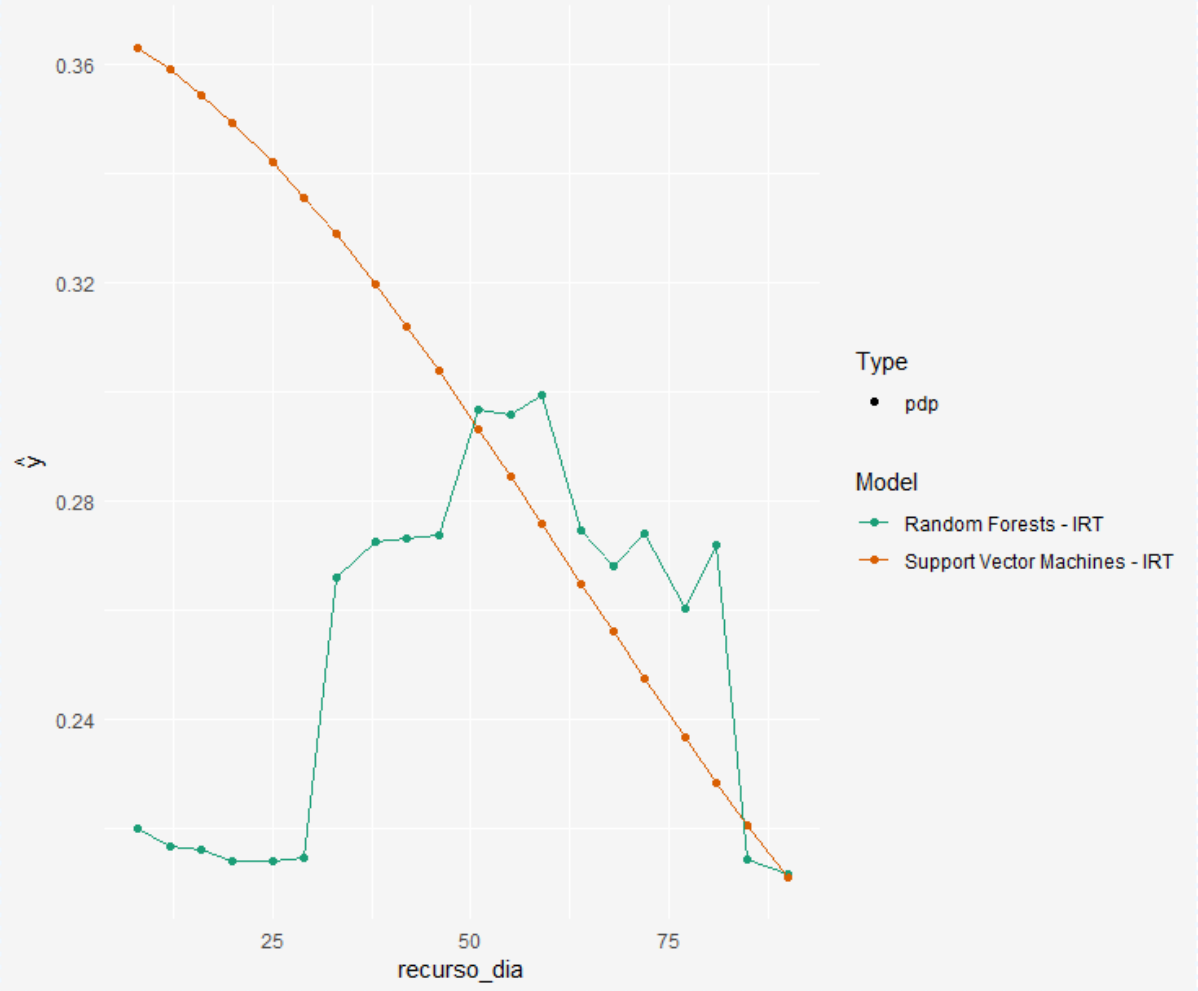
Gráfico 27 – Dependência da variável *volume_hora* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 27 apresenta a evolução da dependência do volume de *cases* criados na última hora em relação à classificação da violação do SLA. Assim como no gráfico 25, o SVM assumiu patamares de dependências muito baixos enquanto a variável independente era baixa. Dessa forma, sua curva assumiu grandes amplitudes para poder se ajustar às predições e não conseguiu captar as relações não lineares que o *Random Forests* captou. Esse último evidenciou que a variável *volume_hora* praticamente não interfere na predição, tanto pelos baixos valores de y, quanto pela pequena variabilidade apresentada.

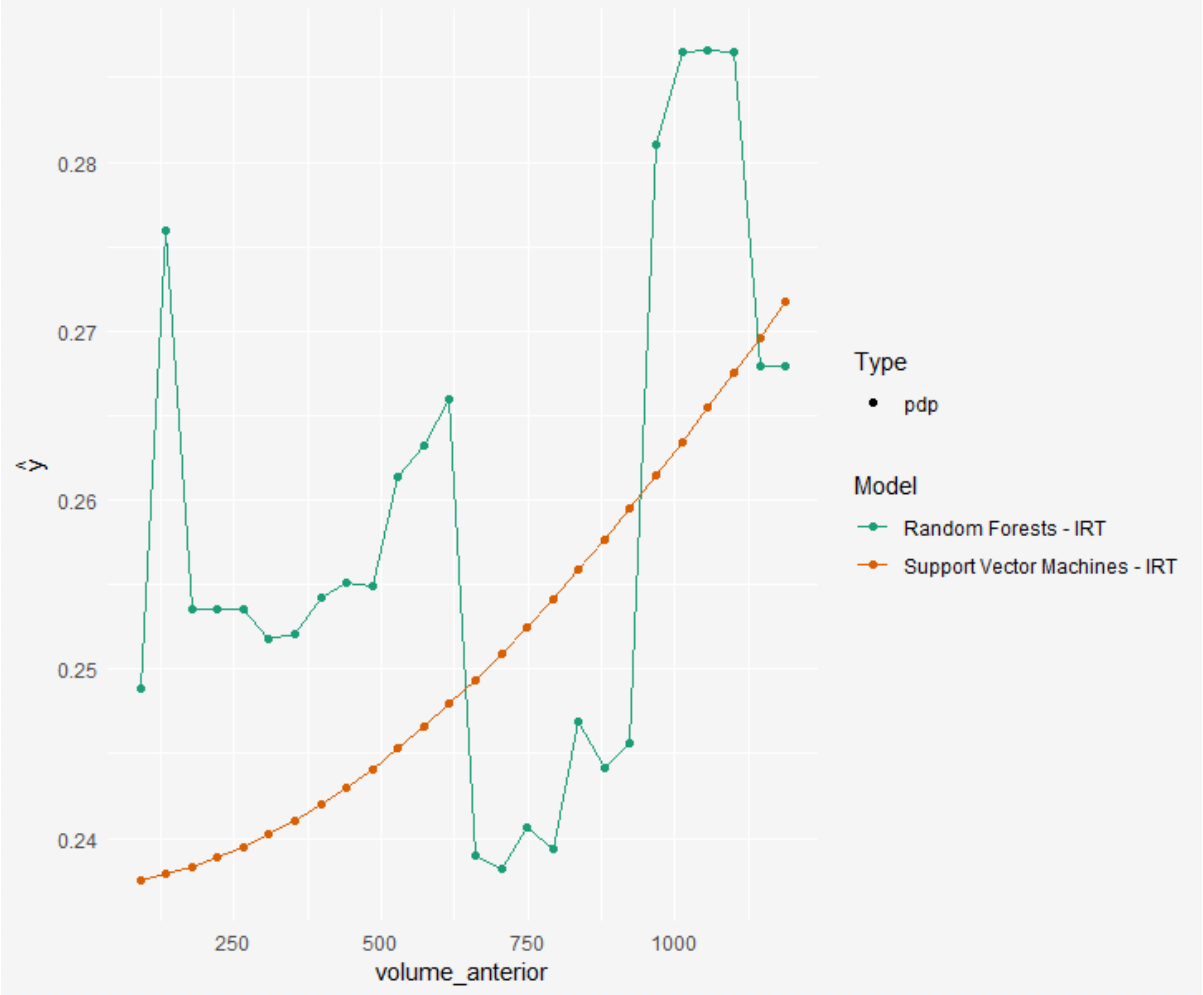
Gráfico 28 – Dependência da variável *recurso_dia* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 28 apresenta a dependência da quantidade de recursos trabalhando na fila de *cases* por dia em relação à possível violação do SLA. Ao inicialmente assumir pontos distantes grandes amplitudes para ajustar suas curvas à dependência, o algoritmo SVM apresentou comportamento semelhante ao descrito anteriormente. Antagonicamente, o *Random Forests* conseguiu identificar as variabilidades que a dependência da variável de resposta tem em relação aos recursos do dia. Dessa forma, pode-se afirmar que a classificação da violação do SLA foi mais dependente da variável em questão quando essa assumiu valores entre 40 e 75.

Gráfico 29 – Dependência da variável *volume_anterior* - IRT



Fonte: Elaborado pelo autor.

O Gráfico 29 mostra a dependência da variável de reposta em relação à quantidade de *cases* criados no dia anterior. Nesse caso, o algoritmo SVM identificou dependências próximas do *Random Forests* uma vez que o patamar e a variabilidade dos dois algoritmos são parecidos. Considerando o aumento de dependência entre os 750 e os 1000 *cases* criados, o SVM foi capaz de captar tal variação de dependência ao aumentar a amplitude da curva. Diferentemente, o *Random Forests* foi sensível às variações e apresentou grande variabilidade, com destaque para o aumento da dependência quando, no dia anterior, tenham sido criados mais de 800 *cases*.

4.4.3 Classificação do tempo de fechamento

O tempo de fechamento dos *cases* é um dos componentes da satisfação dos clientes ao contatarem o help desk. Não obstante, a classificação da violação do SLA depende de variáveis diferentes do tempo de atravessamento de um *case*. Dessa maneira, as observações que tiveram *lead times* maiores que um dia (1440 minutos) foram representadas pela variável “x1”, ao passo que os *cases* que demoraram menos de um dia para serem fechados foram codificados como “x0”. Os algoritmos utilizados foram os mesmos da classificação do SLA.

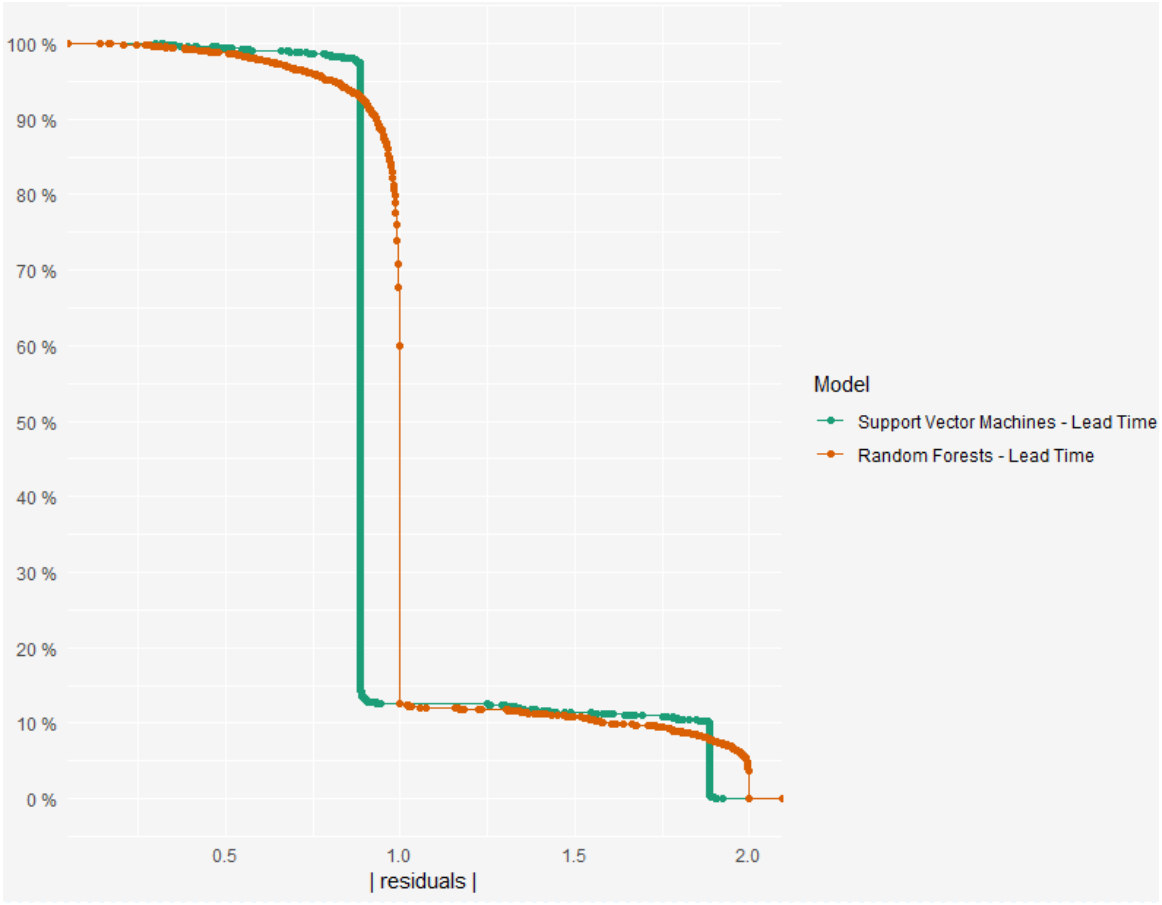
Tabela 22 – Matriz de confusão do tempo de atravessamento

Algoritmo	**Correto x0**	**Correto x1**	**Incorreto x0**	**Incorreto x1**	**Acurácia**	**Intervalo de confiança (95%)**	***Kappa***
Random Forests	2815	74	65	339	87,73%	(0.8656, 0.8883)	0,2188
SVM	2861	37	19	376	88,00%	(0.8685, 0.891)	0,1318

Fonte: Elaborado pelo autor.

A Tabela 22 apresenta os resultados das predições realizadas com os dados de teste e treinamento. Ambos os algoritmos apresentaram baixos coeficientes *kappa*, indicando que foram pouco capazes de identificar as relações entre as duas amostras. A acurácia dos modelos foi semelhante, mas tal métrica foi predominantemente composta pela predição correta dos *lead times* menores que um dia. Por outro lado, os algoritmos não obtiveram boas performances ao classificar a variável x1.

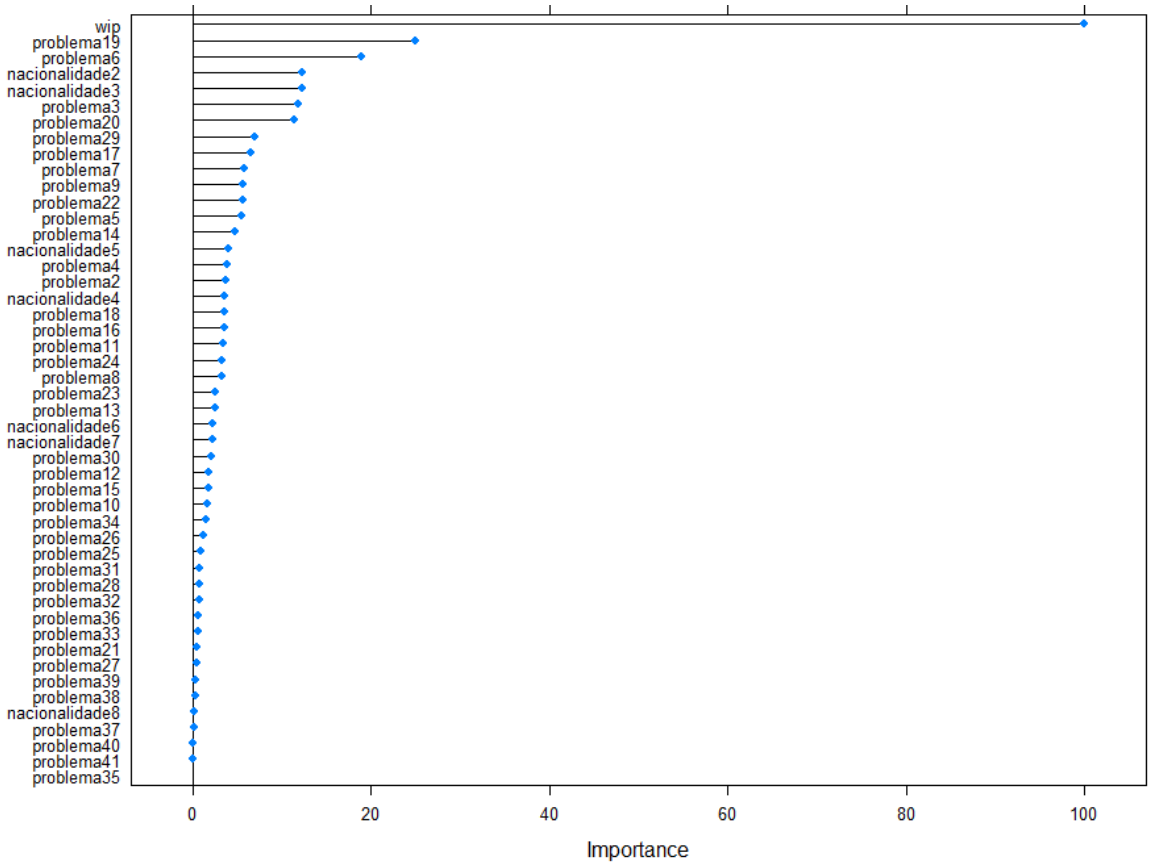
Gráfico 30 – Erros residuais dos modelos - Fechamento



Fonte: Elaborado pelo autor.

O Gráfico 30 mostra os erros residuais que os algoritmos *Random Forests* e SVM apresentaram ao classificar o tempo de atravessamento dos *cases*. O SVM apresentou curvas predominantemente retas, o que indica a dificuldade do algoritmo em predizer as diferentes classificações. O *Random Forests* também foi pouco capaz de ter suas curvas adaptadas às predições, mesmo sendo mais efetivo que o outro modelo.

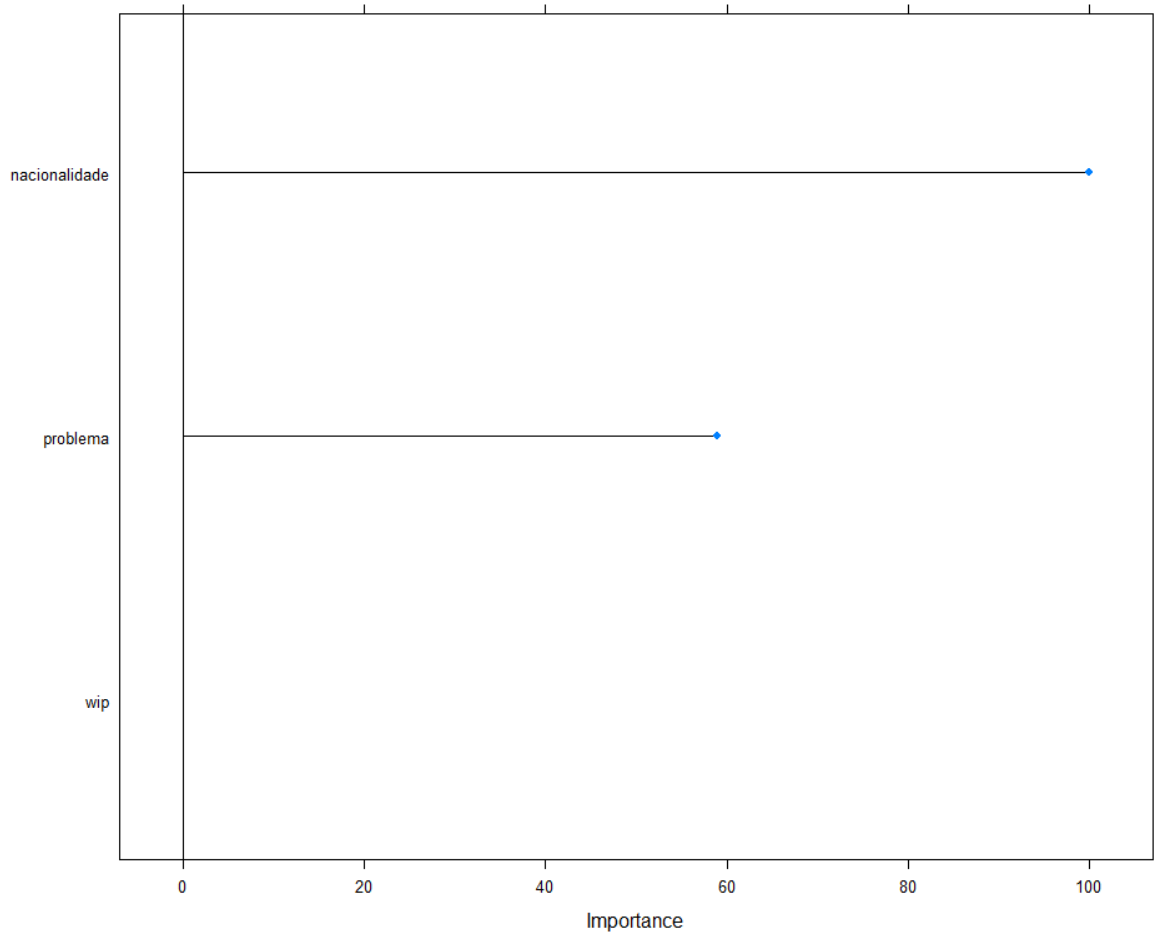
Gráfico 31 – Importância relativa das variáveis no *Random Forests* - Fechamento



Fonte: Elaborado pelo autor.

O Gráfico 31 apresenta as variáveis mais utilizadas pelo *Random Forests* para classificar o tempo de fechamento dos *cases*. Em decorrência da pequena quantidade de entradas, o modelo transformou o *problema* e a *nacionalidade* em variáveis binárias, as quais assumem os valores 1 ou 0 ao, respectivamente, terem determinada entrada como atributo de um *case* ou não. O *wip* foi a variável mais utilizada para calcular as respostas do algoritmo, seguida pelo *problema*.

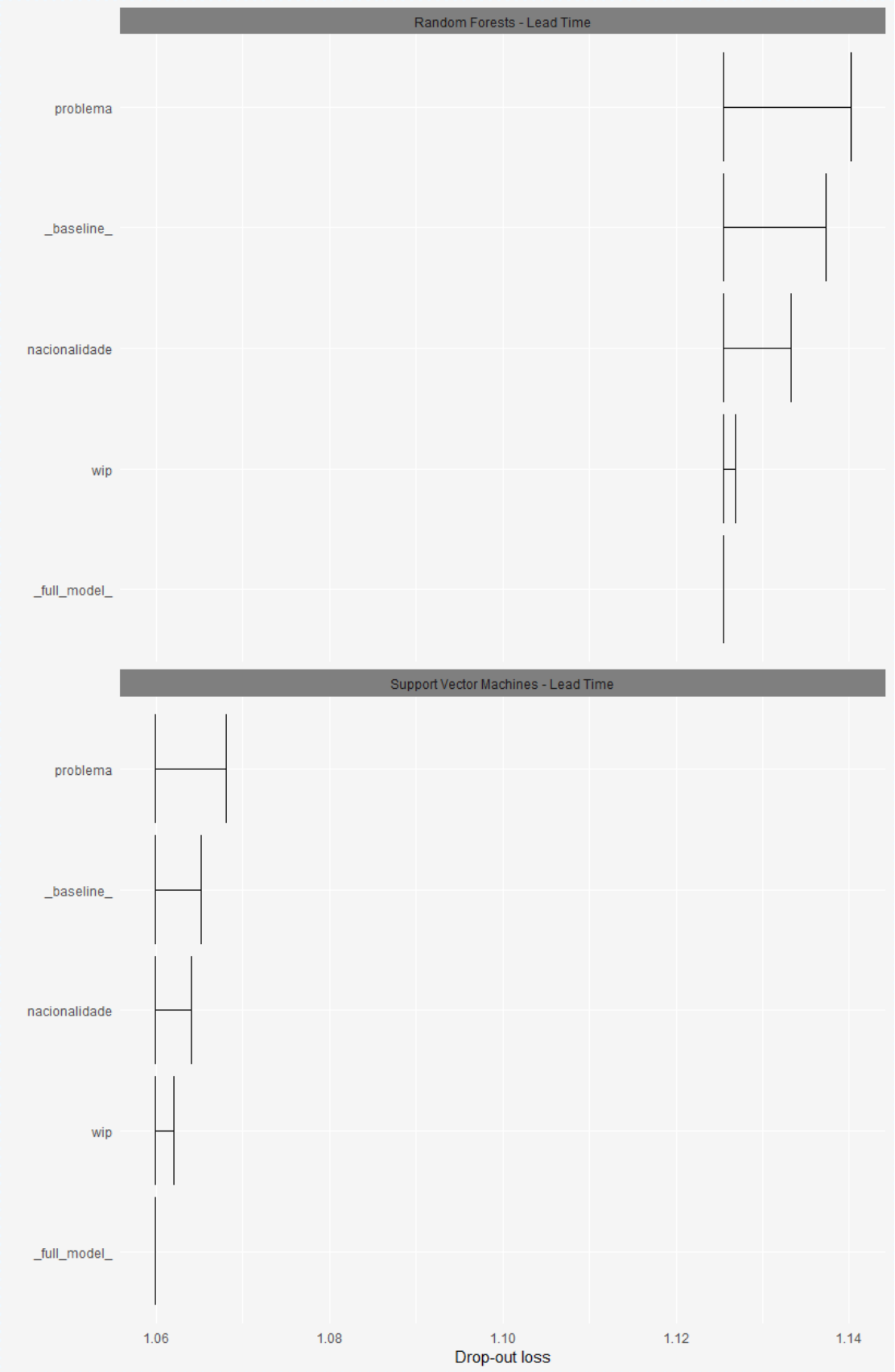
Gráfico 32 – Importância relativa das variáveis no SVM - Fechamento



Fonte: Elaborado pelo autor.

O Gráfico 32 apresenta a utilização das variáveis no cálculo do SVM ao predizer a classificação do tempo de fechamento dos *cases*. Pode-se concluir que, ao contrário do modelo anterior, a variável *wip* não foi utilizada nas predições, cabendo à *nacionalidade* e ao *problema* comporem os cálculos das classificações. A nacionalidade do atendente foi o atributo mais usado para calcular as respostas.

Gráfico 33 – Importância das variáveis - Fechamento



Fonte: Elaborado pelo autor.

O Gráfico 33 apresenta o erro quadrático médio que cada algoritmo teria a mais caso alguma variável fosse desconsiderada pelos modelos. Os valores estão apresentados considerando o *full_model* como referencial e, diferentemente da predição de violação do SLA, todas as variáveis se retiradas uma a uma do modelo acarretariam em perdas de performance maiores do que todo o conjunto de entradas. O *Random Forests* considerou a variável *problema* como a mais importante e o SVM, a *nacionalidade*.

Tabela 23 – Perdas absolutas de performance no *Random Forests* - Fechamento

Variável	**Perdas com base no erro quadrático médio**
problema	1,140
nacionalidade	1,133
wip	1,127

Fonte: Elaborado pelo autor.

A Tabela 23 mostra as perdas absolutas que o algoritmo *Random Forests* teria se cada variável fosse eliminada. O tipo de problema que originou o *case* seria a entrada que mais penalizaria o modelo. Antagonicamente à classificação de violação do SLA, a perda do modelo com a remoção de qualquer variável seria pequena, indicando que o algoritmo não obteve sucesso ao predizer se um *case* demoraria mais de um dia para ser fechado. Sendo assim, as três variáveis foram relevantes para identificar quando o atendimento é encerrado em menos de um dia.

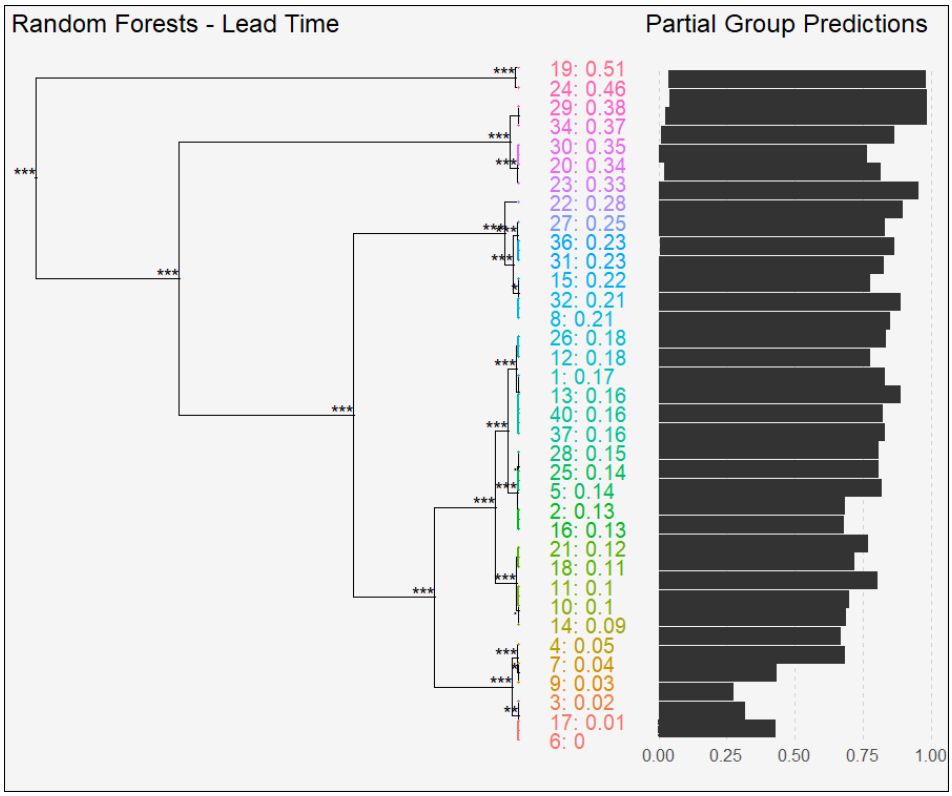
Tabela 24 - Perdas absolutas de performance no SVM - Fechamento

Variável	**Perdas com base no erro quadrático médio**
problema	1,068
nacionalidade	1,064
wip	1,062

Fonte: Elaborado pelo autor.

A Tabela 24 apresenta as perdas de performance do algoritmo SVM se cada variável fosse removida do modelo. Da mesma forma que no *Random Forests*, as diferenças seriam pequenas dada a dificuldade do SVM em classificar o tempo de atravessamento dos *cases*. Dito isso, a importância das variáveis é mais relevante para predizer se o atendimento será encerrado em menos de um dia.

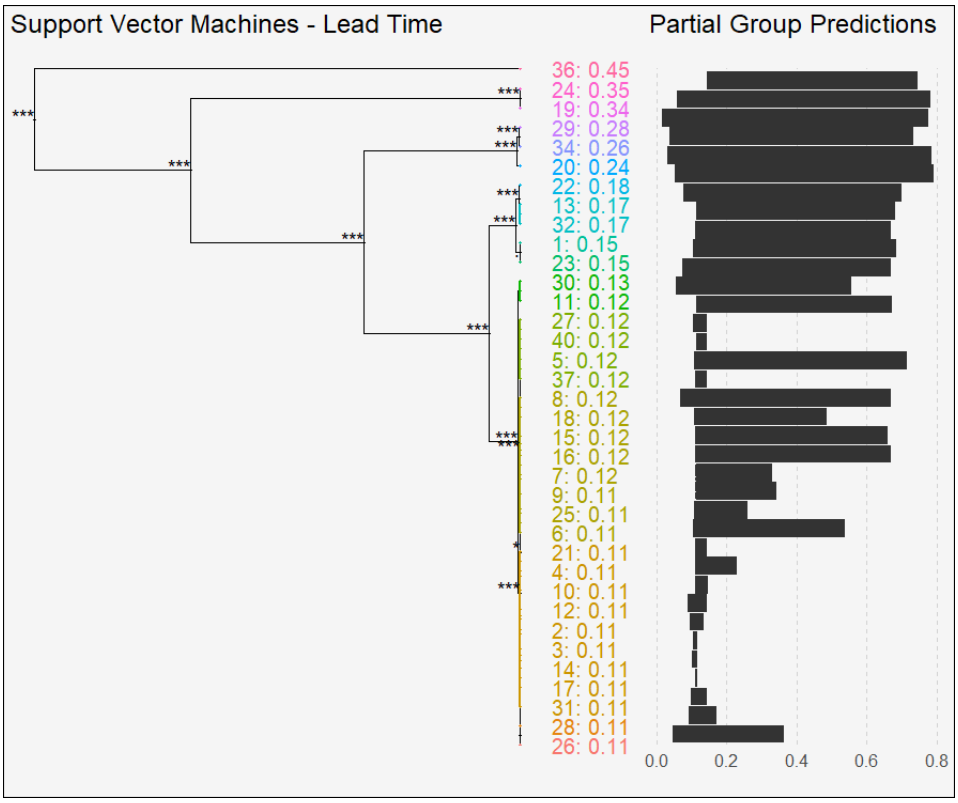
Gráfico 34 – Comportamento da variável *problema* – *Random Forests* - Fechamento



Fonte: Elaborado pelo autor.

O Gráfico 34 apresenta o comportamento da variável *problema* ao compor a classificação do tempo de fechamento dos *cases*. Os tipos de problema 19 e 24 tiveram as maiores taxas de sucesso nas predições, com 0,51 e 0,46, respectivamente. Por outro lado, as entradas 14, 4, 7, 9, 3 17 e 6 apresentaram menos de 10% de predições acertadas. Além disso, 36 tipos de problema foram utilizados para classificar o tempo de atravessamento como menor que um dia (0.00), mas o algoritmo não identificou nenhum atributo capaz de predizer tempos maiores que um dia (1.00).

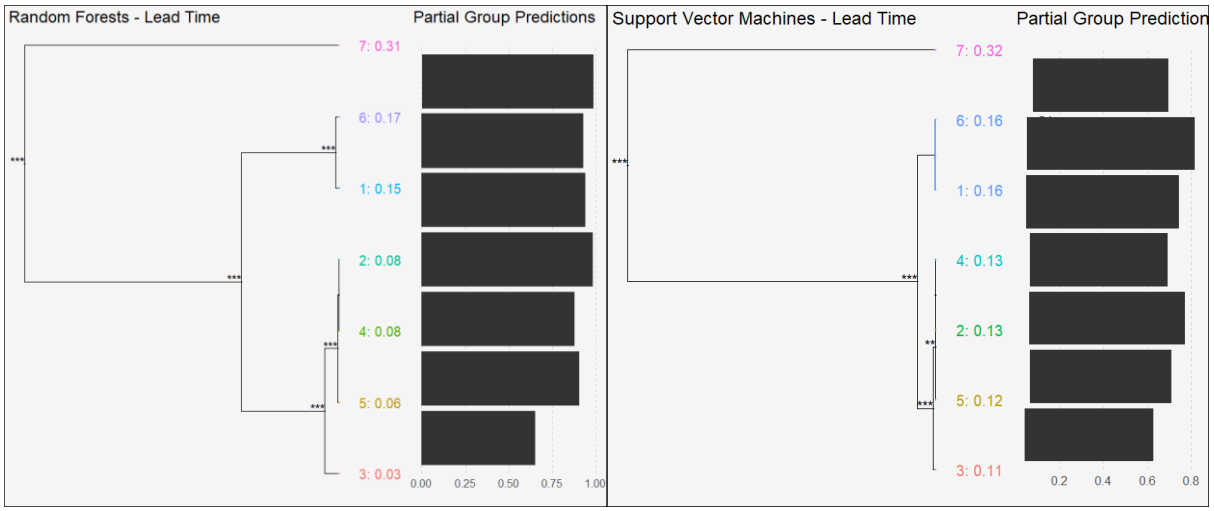
Gráfico 35 - Comportamento da variável *problema* – SVM - Fechamento



Fonte: Elaborado pelo autor.

O Gráfico 35 mostra como o SVM utilizou os tipos de problema para classificar o tempo de fechamento dos *cases*. Diferentemente do outro algoritmo, esse modelo apresentou várias entradas com baixo poder discricionário, exemplificadas pelos problemas 17, 14, 3, 2, 12, 10, 21, 37, 40 e 27. O tipo de problema 36 foi isoladamente o que obteve maior taxa de sucesso, com 45% das predições corretas. Assim como no *Random Forests*, o modelo enfrentou dificuldades ao tentar chegar na classificação 1.00, tendo como resultado *Partial Group Predictions* menores do que 0.8.

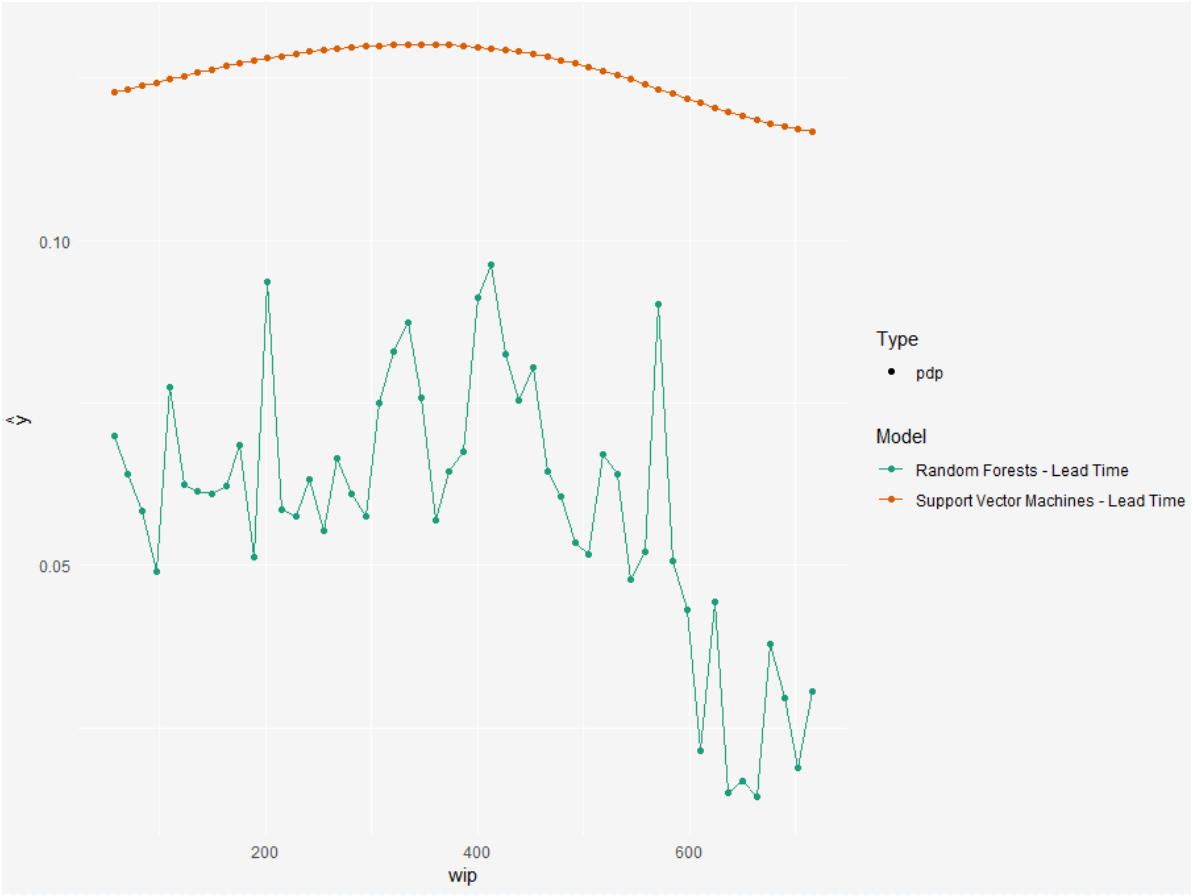
Gráfico 36 – Comportamento da variável *nacionalidade* - Fechamento



Fonte: Elaborado pelo autor.

O Gráfico 36 apresenta o comportamento da variável *nacionalidade* ao compor a variável de resposta dos modelos. As taxas de sucesso apresentadas pelo SVM foram maiores que do *Random Forests*, com a nacionalidade do atendente representada pelo número 7 assumindo, respectivamente, 0,32 e 0,31. Da mesma forma, a nacionalidade 3 foi a pior classificadora em ambas as situações, acertando 11% das predições com o SVM e 3% com o *Random Forests*. Os dois algoritmos formaram 7 clusters no dendograma.

Gráfico 37 – Dependência parcial da variável *wip* - Fechamento



Fonte: Elaborado pelo autor.

O Gráfico 37 mostra a variação da dependência do tempo de fechamento dos *cases* em relação à variável *wip*. Em decorrência da característica do algoritmo SVM, ele se ajustou menos às variações do que o *Random Forests*, mas ambos foram sensíveis à diminuição da dependência a partir dos 400 *cases* em processamento. Sendo a amplitude das curvas, em média, baixa, pode-se concluir que a variável de resposta depende mais do *wip* quando esse assume valores até 400 *cases*, visto que acima disso o algoritmo considerou a demanda tão alta que se tornou irrelevante para a classificação.

Quadro 16 – Síntese da importância das variáveis

Ordem de Importância	**IRT – RF**	**IRT – SVM**	**Fechamento – RF**	**Fechamento – SVM**
1	*Tipo_user*	*Tipo_user*	*Problema*	*Problema*
2	*Idioma*	*Dia*	*Nacionalidade*	*Nacionalidade*
3	*Problema*	*Idioma*	*Wip*	*Wip*

Fonte: Elaborado pelo autor.

O Quadro 16 apresenta uma síntese dos resultados de todas análises de importância de variáveis realizadas na presente pesquisa. O tipo de usuário que contata o help desk, representado pela variável *tipo_user*, é o atributo mais importante para predizer se um *case* violará ou não o SLA. Em ambos os algoritmos, o tipo 7 foi determinante para as predições ao alcançar, respectivamente no *Random Forests* e no SVM, taxas de sucesso de 86% e 96%.

A classificação do tempo de fechamento dos *cases* maior que um dia apresentou a mesma ordem de importância das variáveis em ambos os algoritmos. Além disso, as tabelas Tabela 23 e Tabela 24 mostram que as perdas de performance dos modelos são semelhantes para a remoção individual de todas as entradas. Adicionalmente, a variável *problema*, considerada a mais importante, apresentou taxas máximas de sucesso de 51% e 45% nos algoritmos *Random Forests* e SVM, respectivamente.

5 DISCUSSÃO DOS RESULTADOS E IMPLICAÇÕES GERENCIAIS

A análise anterior apresentou resultados que são relevantes para compor o estudo detalhado acerca dos processos. Sendo assim, surge a necessidade de realizar uma discussão sobre as contribuições que o presente estudo oferece à comunidade acadêmica. Primeiramente, serão apresentadas as contribuições do *process mining* e, posteriormente, dos modelos preditivos.

A pesquisa contribuiu para a academia com a utilização de duas técnicas que são frequentemente aplicadas separadamente. A partir da revisão sistemática da literatura realizada, foi verificada uma escassez de estudos com aplicações reais de *process mining* para não apenas desenvolver novos algoritmos e mostrar que a modelagem de processos tradicional é falha, mas também para que os resultados de tal análise possam servir de inputs para outros métodos analíticos.

Além da utilização dos resultados do *process mining* para compor outras análises, a pesquisa também contribuiu no âmbito das diferenças de importâncias e comportamentos entre as variáveis que explicam o atendimento ao SLA e as que classificam o tempo de atravessamento dos *cases*. Dessa forma, o estudo foi capaz de comprovar que, com o mesmo conjunto de entradas, a diferença de importância das variáveis existe ao ponto de reduzir a performance do segundo modelo preditivo.

Dumas e Maggi (2014) e Van der Aalst (2016) apresentaram pesquisas afirmando que as metodologias tradicionais de BPM como coletas manuais de dados, entrevistas e *workshops* podem falhar ao considerar apenas os processos ideais e não evidenciar as exceções. Além disso, os autores convergem para a ideia de que as análises de processos baseadas em dados elevam o patamar do BPM. Sendo assim, a presente pesquisa contribui analisando o fluxo em questão com base nos dados extraídos diretamente de diversos relatórios do CRM.

Van der Aalst, La Rosa e Santoro (2016) estudaram os diferentes componentes do BPM para discutir acerca do real objetivo desses estudos, que é a melhoria dos processos. Eles afirmaram que não há valor em estudar o *process mining* para desenvolver algoritmos precisos se o processo não for analisado ou melhorado. Dessa maneira, o estudo contribuiu com a utilização dos resultados do *process mining* para servirem de entradas em modelos preditivos.

Turner et al. (2012) pesquisaram as ferramentas comerciais que suportam a aplicação de *process mining* e apontaram que há mais de 40 softwares com tais

capacidades. Além disso, os autores concluíram que a otimização de processos de negócio com base em dados é um campo promissor, pois possibilita a redução de custos e modelagens customizadas mais rapidamente. Assim, o presente estudo contribuiu com a aplicação do *process mining* na plataforma R, que não foi considerada pelos autores, e com as análises dos comportamentos das variáveis para a compreensão dos impactos das mesmas no sistema.

Os estudos de Jiménez (2017) e De Vries et al. (2017) aplicaram as técnicas de descoberta de processos e verificação de conformidade para visualizar e identificar desvios nos fluxos estudados. Tais pesquisas se limitaram a identificar o algoritmo de *process mining* que mais se adequou aos dados disponíveis. Sendo assim, o presente estudo contribuiu com a combinação da descoberta de processos com as análises preditivas.

Chuchaimongkhon, Porouhan e Premchaiswadi (2017) aplicaram o *process mining* em um call center bancário mas apenas apresentaram os resultados e os modelos. Dessa forma, a presente pesquisa contribuiu com análises descritivas do processo e modelos preditivos que, ao serem analisados, foram capazes de apontar as variáveis mais importantes para identificar a violação do SLA.

Den Hertog (2008) pesquisou abordagens preditivas que foram aplicadas em *event logs*. Considerando-se que o estudo do autor se limitou a encontrar o melhor algoritmo para determinada situação, a presente pesquisa contribuiu com a análise da importância das variáveis de modelos preditivos sob uma perspectiva de negócio. Ou seja, um estudo analisou o quanto um algoritmo explicou um *event log* e outro como os inputs ao longo das predições, de modo a gerar entendimento acerca de como o processo é impactado.

No estudo de Polato et al. (2018), foram realizadas predições para identificar a próxima atividade de um *case* e o tempo restante previsto para o encerramento de tal. Diferentemente da presente pesquisa, os autores aplicam modelos preditivos diretamente no *event log* e não explicam a seleção das variáveis e a análise das mesmas. Sendo assim, este estudo contribuiu com a seleção prévia de variáveis para que as conclusões do *process mining* pudessem ser testadas à luz de outros métodos.

Ainslie et al. (2017) aplicaram redes neurais para que o atendimento aos SLA's pudessem ser preditos. Os autores identificaram as variáveis de entrada, as avaliaram com base em um algoritmo de *machine learning*, mas os resultados do estudo foram focados apenas na aderência dos algoritmos e em como melhorar a precisão. O

presente estudo contribuiu com a análise detalhada do comportamento de cada variável, sejam elas contínuas ou classificatórias. Além disso, foi feito um comparativo entre os impactos nas violações do SLA e no tempo de atravessamento dos *cases*.

O mesmo estudo de Ainslie et al. (2017) também não aponta qual o SLA que está sendo analisado. A única informação sobre tal é que o acordo entre as partes que define o período de tempo em que determinadas atividades precisam ser executadas é chamado de SLA. A falta de indicação temporal serviu de oportunidade para a outra contribuição da presente pesquisa, a qual compara a importância das variáveis e a dependência das respostas ao classificar IRT's acima de 120 minutos e fechamentos com mais de 1440 minutos. Finalmente, Ainslie et al. (2017) ainda sugeriram futuras análises com o algoritmo SVM, o qual foi implementado na presente pesquisa mas não apresentou boa aderência aos dados, dada a sua dificuldade em se adaptar às observações com alta variabilidade.

Dadas as diversas lacunas apresentadas, a contribuição acadêmica da presente pesquisa é aplicação do *process mining* e de técnicas preditivas para comprovar que as variáveis que impactam no atendimento ao SLA não são as mesmas que explicam o tempo de atravessamento dos *cases*. Em relação ao processo analisado, os tipos de usuários foram considerados determinantes para o IRT pois as complexidades dos problemas que cada um enfrenta são distintas e, visto que a resposta inicial requer um encaminhamento inicial à resolução do *case*, determinados usuários podem possuir maiores probabilidades de ter o SLA violado.

A variável *idioma* também foi considerada uma das mais importantes para determinar o cumprimento do SLA. Sendo o help desk um processo global, os *cases* podem ser submetidos em qualquer idioma, gerando a necessidade de contato com atendentes específicos distribuídos globalmente e consequente retardo do tempo inicial de resposta.

Considerando-se que as variáveis *idioma* e *tipo_user* apresentaram taxas de sucesso superiores à 50% ao predizer a violação do SLA, elas podem embasar a definição do cenário de menor probabilidade de cumprimento do acordo. Dito isso, o modelo pode ser utilizado como justificativa para a implementação de um marcador visual, com o objetivo de destacar os *cases* que são abertos pelo tipo de usuário 7 e nos idiomas 3, 4, 5, 6 ou 8, visto que essa combinação acarreta a maior probabilidade de violação do SLA.

A predição do tempo de atravessamento maior que um dia não obteve resultados satisfatórios ao analisar o mesmo conjunto de variáveis utilizado para classificar o cumprimento ou violação do SLA. Considerando-se que o atendimento ao SLA depende mais de esforços internos da empresa analisada, é mais provável que se possa predizer o tempo da resposta inicial de um *case* dadas as condições sob as quais ele foi aberto, do que o tempo de fechamento do mesmo, visto que esse depende de interações com os clientes.

Sendo o tempo de fechamento de um *case* dependente do engajamento dos clientes, fatores como a não leitura de e-mails, períodos de férias, conhecimento do sistema ou desinteresse por parte dos usuários podem retardar a conclusão dos atendimentos. Todas essas questões são identificadas à medida que o atendente interage com os clientes, de modo a não ser possível mensurá-las logo na abertura dos *cases* se consideradas as informações disponíveis atualmente.

O trabalho contribuiu com a organização estudada pois, além de aplicar o *process mining* para detalhar e analisar o processo sob diversas óticas, utilizou modelos preditivos para que, com base nos resultados e na performance das predições, as variáveis pudessem ser avaliadas tanto em relação à importância, quanto à dependência da resposta. Tal análise possibilitou a identificação dos idiomas e tipos de usuário que, combinados, acarretam maiores probabilidades dos *cases* violarem o SLA.

Não obstante, atualmente os tempos do processo são exclusivamente monitorados através de relatórios diários, o que acarreta ações de melhorias predominantemente corretivas e por tentativa e erro. Com o presente trabalho, tanto o *process mining* quanto o estudo das variáveis podem prover suporte operacional à gestão.

No *process mining* a quantidade de caminhos possíveis ao realizarem as quatro últimas atividades é o primeiro ponto de destaque. A gestão pode se beneficiar dessa informação para revisar os processos e planejar treinamentos com o objetivo de padronizar as atividades, de modo a facilitar e agilizar futuras análises e ações de melhorias.

Outra informação relevante é que o *process mining* evidenciou que a maioria dos *cases* são fechados em até 1000 minutos. Como na situação dos caminhos anteriormente apresentada, isso mostra a importância de haver ações de padronização e compartilhamento de conhecimento acerca do processo, para que a

performance da fila, que possui alto volume, possa ser melhorada de modo que cada recurso que pegue um *case* já saiba as atividades a realizar.

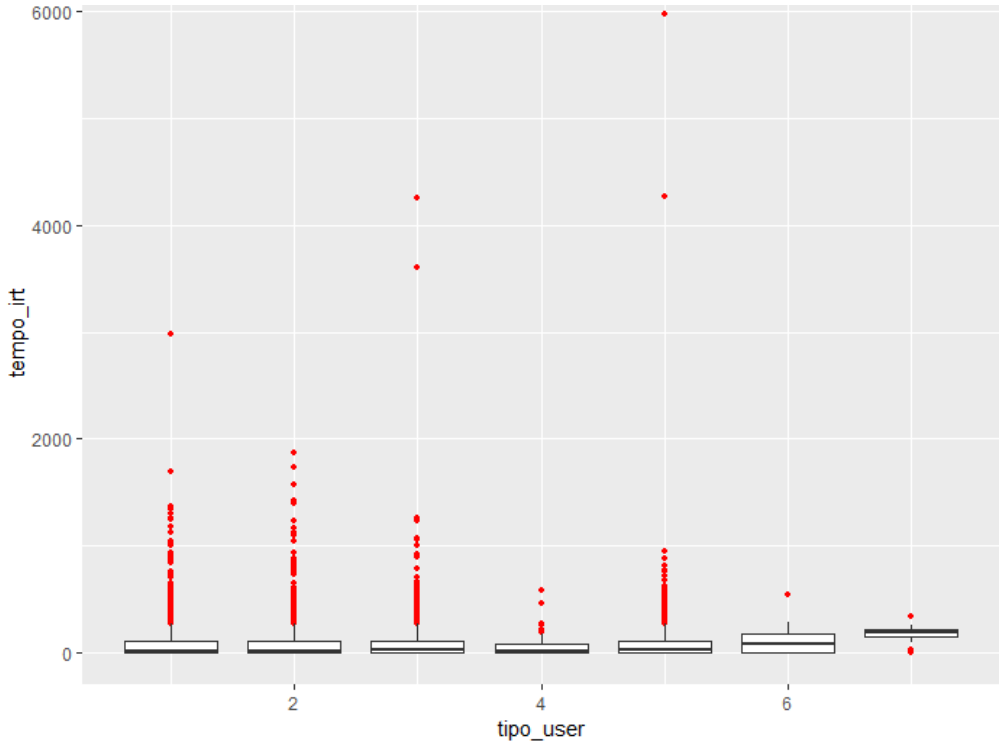
A demanda do processo se mostrou variável ao longo dos dias e das horas. Tal variação impacta negativamente a performance subsequente do sistema, de modo que o volume da segunda-feira traz consequências ao tempo de atravessamento pelo resto da semana. Isso evidencia que se a demanda do início da semana for controlada, os outros dias sofreriam menos com o estoque em processamento. Dessa forma, aumentar o número de atendentes na segunda-feira e garantir a disponibilidade dos mesmos com dedicação exclusiva à fila são ações de melhorias que poderiam ser analisadas pela gestão.

Considerando-se os resultados dos modelos preditivos, as variáveis *hora* e *volume_hora,* que foram apresentadas como potencialmente impactantes sob a ótica do *process mining*, não acarretariam perdas significantes na capacidade dos modelos em classificar as violações do SLA e o tempos de atravessamento. Sendo assim, a alocação de recursos ao longo do dia não precisa ser modificada mesmo com os picos de demanda, pois os clientes estão sendo atendidos dentro do acordado sob o ponto de vista das horas do dia.

As classificações realizadas através de algoritmos de *machine learning* mostraram que as variáveis que impactam o atendimento ao SLA não são as mesmas que definem se um *case* demorará menos de um dia para ser fechado. Tal conclusão serve de subsídio para a gestão tratar de modo diferente as ações de melhoria para os tempos de fechamento e inicial de resposta. São dois cenários diferentes e impactados de maneiras distintas, de modo que um único plano de melhoria para ambos os indicadores não seria adequado.

A análise de resultados mostrou que a variável mais importante para predizer se um *case* violará o SLA é a *tipo_user*. Sendo assim, o foco de atuação da gestão para garantir que os atendentes respondam os clientes em até 120 minutos deveria ser entender as relações entre o tipo de usuário que requisita o suporte e a performance do sistema.

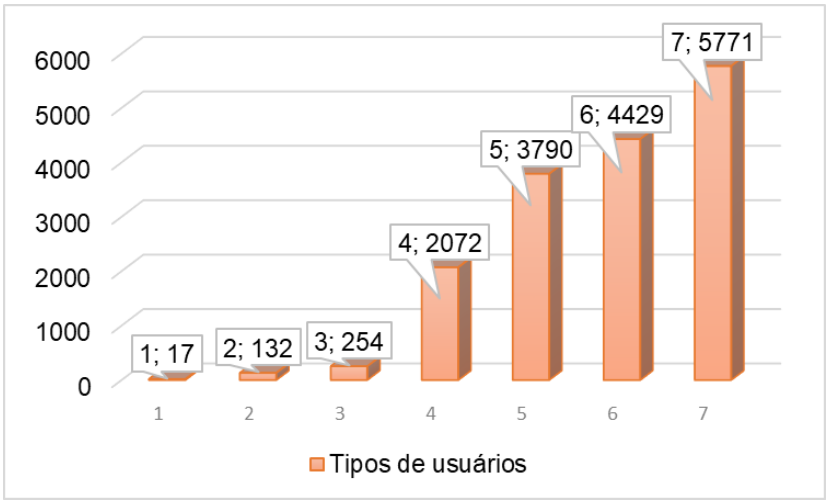
Gráfico 38 – Tempo de resposta por tipo de usuário – IRT



Fonte: Elaborado pelo autor.

O Gráfico 38 apresenta os tempos da primeira resposta aos clientes em função do tipo de usuário. É possível concluir que a combinação das análises preditivas com o *process mining* é válida, uma vez que os usuários 1 e 2 possuem mais outliers do que os outros. Dessa forma, a pesquisa contribui para a gestão com a proposta de separar os recursos em times que atendam os tipos de usuário separadamente.

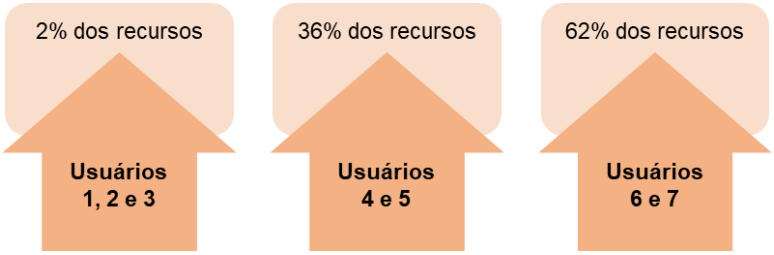
Gráfico 39 – Volume de *cases* por tipo de usuário



Fonte: Elaborado pelo autor.

Complementarmente à análise dos outliers, o Gráfico 39 apresenta a demanda por tipo de usuário. Os usuários que mais demoram a receber a primeira reposta são os que menos criam *cases*. Por outro lado, os tempos de resposta mais estáveis foram verificados nos usuários 6 e 7. Considerando-se a proporção da demanda e o agrupamento dos tipos de usuários de acordo com a incidência de outliers, a Figura 21 apresenta uma proposta de redistribuição de atendentes.

Figura 21 – Redistribuição dos recursos



Fonte: Elaborado pelo autor.

A Figura 21 apresenta a proposta de segmentação do atendimento por tipo de usuário. Desse modo, a gestão pode ter recursos específicos alocados para garantir o atendimento ao SLA de acordo com a demanda. Sendo assim, treinamentos e melhorias do processo poderão ser realizados de acordo com os segmentos, especializando a mão de obra e, consequentemente, diminuindo os tempos de atravessamento dos *cases* em função do maior conhecimento técnico dos atendentes.

Quadro 17 – Síntese das implicações gerenciais do estudo

Melhoria	**Descrição**	**Impacto**
Identificação visual dos *cases* com maior probabilidade de violação do SLA	Implementação de um marcador visual nos *cases* que combinem o tipo de usuário 7 e os idiomas 3, 4, 5, 6 ou 8	Os atendentes poderão priorizar esses *cases* uma vez que será possível a identificação do maior risco de violação do SLA
Ajuste da capacidade do sistema nas segundas-feiras	Aumento do número de atendentes e recomendação para que não sejam agendadas reuniões, treinamentos e ligações nas segundas-feiras	Aumento da disponibilidade dos recursos e menor estoque em processamento pelo resto da semana

Melhoria	Descrição	Impacto
Segmentação dos atendentes de acordo com os tipos de usuários	Divisão dos atendentes com base na demanda de cada grupo de usuários	Especialização dos recursos de modo a acelerar os atendimentos e capacidade adequada à demanda de cada segmento

Fonte: Elaborado pelo autor.

O Quadro 17 mostra a síntese das implicações gerenciais da presente pesquisa. Foram apresentados três pontos de melhoria, os quais perpassam pela reorganização dos recursos e pela identificação visual dos *cases* com maior probabilidade de violação de SLA. Com isso, a empresa estudada poderá evitar multas e aumentar a satisfação dos clientes uma vez que, mesmo com as ações sendo direcionadas ao atendimento do SLA, as melhorias propostas possibilitam o aumento do conhecimento dos atendentes acerca dos seus focos de atuação.

6 CONSIDERAÇÕES FINAIS

O presente trabalho teve o objetivo de analisar os dados de um processo de help desk, de modo a predizer sob quais condições o SLA é cumprido ou não, bem como apontar as variáveis que melhor explicam esses cenários. Dessa forma, foram realizadas revisões da literatura que identificaram a carência de aplicação do *process mining* em conjunto com técnicas preditivas com o objetivo de contribuir para um problema de negócio, sem o foco da acurácia dos algoritmos.

O objetivo principal foi decomposto em objetivos específicos, sendo eles: modelar o processo atual com base no *event log*; identificar as variáveis que possam impactar na performance do sistema; avaliar o impacto de tais variáveis no atendimento ao SLA e analisar o comportamento das mesmas frente ao tempo de atravessamento dos *cases*.

Sendo assim, o primeiro objetivo específico foi atingido com base na literatura de Van der Aalst (2016), a qual possibilitou que fossem seguidos cinco passos para a aplicação do *process mining*. A utilização de tal técnica proveu entendimento acerca do processo estudado e subsidiou o atendimento ao segundo objetivo específico, que foi atingido com a listagem das variáveis a serem coletadas para posterior análise.

Os dois últimos objetivos específicos foram atingidos baseados na literatura de Johnson e Kuhn (2013). Os dados coletados foram avaliados à luz das técnicas de análise de correlação, *recursive feature selection* e do algoritmo *Boruta*. As saídas desse processo serviram de entrada para os algoritmos *Random Forests* e *Support Vector Machines*, para que as situações de violação do SLA e tempo de atravessamento maior que um dia pudessem ser classificadas de modo a gerarem explicações acerca do comportamento das variáveis ao identificar tais cenários.

Analisados os resultados, a contribuição acadêmica da pesquisa é a aplicação do *process mining* em conjunto com técnicas preditivas para comprovar que as variáveis que impactam no atendimento ao SLA não são as mesmas que explicam o tempo de atravessamento dos *cases.*

As implicações gerenciais da pesquisa foram consequências das contribuições à comunidade acadêmica. O fato das mesmas variáveis não explicarem os diferentes tempos analisados comprova que as ações de melhoria e as análises dos processos não podem ser generalizada para ambos os casos. Os tempos de resposta e

atravessamento precisam ser submetidos à ações de melhorias distintas, pois suas variáveis de impacto não são as mesmas.

Não obstante, o *process mining*, foi capaz de identificar que, sendo *tipo_user* e *idioma* as variáveis mais importantes para classificar violações do SLA, há um padrão nos outliers dos tempos de resposta em função do tipo de usuário. Sendo assim, foi sugerida segmentação do atendimento com base na proporção da demanda verificada nos dados analisados, de modo a facilitar o atendimento ao SLA e, consequentemente, a satisfação dos clientes.

Mesmo com todas as contribuições apresentadas, o estudo possui limitações. O *process mining* não considerou os tempos de ciclo das atividades pois as investigações não são realizadas no sistema em que os dados foram extraídos. Considerando que cada evento é o registro de uma interação pontual com o CRM, a atividade *edit* agrupou diversas outras, como a escolha do problema, o tipo de cliente, o nome da companhia do cliente, o idioma e o canal de comunicação do *case*. Outra limitação são os dados monetários, pois os diversos contratos e a diferenciação entre usuário e cliente contratante não foram consideradas no trabalho.

Dadas as limitações, o *process mining* pode ser deficitário se aplicado sozinho, pois muitas interações podem não ser discriminadas em decorrência de agrupamentos, como na atividade *edit*. Isso também evidencia que, mesmo com a facilidade das modelagens, o *process mining* ainda precisa da supervisão de especialistas do processo para garantir que os dados de entrada sejam adequados e interpretar os modelos sabendo das suas limitações.

As futuras pesquisas podem avaliar as variáveis que impactam o tempo de atravessamento de um *case*. Para tal, sugere-se a coleta de dados como a criticidade dos atendimentos, necessidade de escalações e o cliente que requer suporte. Outra sugestão é analisar o impacto que os telefonemas e reuniões possuem no sistema, agregando mais variáveis do processo e do recurso, como a disponibilidade.

Como extensão da pesquisa nos mesmos moldes da realizada, recomenda-se o teste de outros algoritmos para classificar a violação do SLA e os tempos de fechamento, como redes neurais, as quais possuem diversos parâmetros de *tuning* e podem ser adaptadas aos mais variados desafios de aprendizado supervisionado.

REFERÊNCIAS

AINSLIE, R. et al. Predicting Service Levels Using Neural Networks. In: BRAMER, M.; PETRIDIS, M. (Eds.). Lecture Notes in Computer ScienceCham: Springer International Publishing, 2017. v. 10630p. 411–416.

BIECEK, P. DALEX: explainers for complex predictive models. [s. l.], v. 1, p. 1–14, 2018. Disponível em: <https://arxiv.org/abs/1806.08915>

BRAMER, M. **Principles of Data Mining**. [s.l: s.n.]. Disponível em: <http://link.springer.com/10.1007/978-1-4471-4884-5>

BREIMAN, L. Random forests. **Machine Learning**, [s. l.], v. 45, n. 1, p. 5–32, 2001.

BUCKLEY, P.; MAJUMDAR, R. **The services powerhouse: Increasingly vital to world economic growth**. 2018. Disponível em: <https://www2.deloitte.com/insights/us/en/economy/issues-by-the-numbers/trade-in-services-economy-growth.html>.

CHEN, M. S.; HAN, J.; YU, P. S. Data mining: An overview from a database perspective. **IEEE Transactions on Knowledge and Data Engineering**, [s. l.], v. 8, n. 6, p. 866–883, 1996.

CHUCHAIMONGKHON, C.; POROUHAN, P.; PREMCHAISWADI, W. A study to investigate time durations of a call center customer service using transition systems. **International Conference on ICT and Knowledge Engineering**, [s. l.], p. 93–97, 2017. Disponível em: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85010789914&doi=10.1109%2FICTKE.2016.7804104&partnerID=40&md5=86e23fda5345b525c64c3cc38a421edb>

DAVENPORT, T. H. Competing on analytics. **Harvard Business Review**, [s. l.], n. January, 2006.

DE VRIES, G.-J. et al. Towards Process Mining of EMR Data - Case Study for Sepsis Management. In: PROCEEDINGS OF THE 10TH INTERNATIONAL JOINT CONFERENCE ON BIOMEDICAL ENGINEERING SYSTEMS AND TECHNOLOGIES 2017, **Anais**... : SCITEPRESS - Science and Technology Publications, 2017. Disponível em: <http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0006274405850593>

DEN HERTOG, S. N. **Case prediction in BPM systems estimating the**

completion time of individual cases. 2008. Eindhoven University of Technology, [s. l.], 2008.

DIMITRIADOU, A. E. et al. The e1071 Package. [s. l.], n. November 2009, 2009.

DRESCH, A.; LACERDA, D. P.; ANTUNES, J. A. V. **Design science research: método de pesquisa para avanço da ciência e tecnologia.** Porto Alegre: Bookman, Livro eletrônico., 2015.

DRESCH, A.; LACERDA, D. P.; JÚNIOR, J. A. V. A. **Design Science Research: A Method for Science and Technology Advancement**. Porto Alegre: Bookman, Livro eletrônico., 2015.

DUMAS, M. et al. **Fundamentals of Business Process Management**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. Disponível em: <http://dspace.fudutsinma.edu.ng/jspui/bitstream/123456789/358/1/Fundamentals of Business Process Management.pdf>

DUMAS, M.; MAGGI, F. M. **Evidence-based business process management**. [s.l.] : Elsevier Inc., 2014. v. 1 Disponível em: <http://dx.doi.org/10.1016/B978-0-12-799959-3.00017-3>

EISENHARDT, M. Building Theories from Case. **Academy of Management Review**, [s. l.], v. 14, n. No: 4, p. 532–550, 1989.

GIL, A. C. **Como Elaborar Projetos de Pesquisa**. [s.l: s.n.]. Disponível em: <http://www.ie.ufrj.br/intranet/ie/userintranet/hpp/arquivos/031120162924_AntonioCarlosGil_ComoElaborarProjetosdePesquisa_EditoraAtlasCopia.pdf>

GRIGORI, D. et al. Business Process Intelligence. **Computers in Industry**, [s. l.], v. 53, n. 3, p. 321–343, 2004.

HAIR, J. F. H. et al. **Essentials of Business Research Methods**. [s.l: s.n.].

HAND, D. et al. **Principles of data mining**. [s.l: s.n.]. v. 30 Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/17604416>

HELO, P.; GUNASEKARAN, A.; RYMASZEWSKA, A. Managing Service Delivery. In: [s.l: s.n.]. p. 49–56.

HERMANN, M.; PENTEK, T.; OTTO, B. Design Principle for Industrie 4.0 Scenarios: A Literature Review. **Hawaii International Conference on System Sciences (HICSS)**, [s. l.], n. 01, p. 16, 2016.

JANSSENSWILLEN, G.; DEPAIRE, B. BupaR: Business process analysis in R. **CEUR Workshop Proceedings**, [s. l.], v. 1920, p. 2–6, 2017.

JIMÉNEZ, H. G. **Applying Process Mining to the Academic Administration**

Domain. 2017. PUC-Rio, [s. l.], 2017.

KUHN, M. Building Predictive Models in R Using the caret Package. **Journal of Statistical Software**, [s. l.], v. 28, n. 5, p. 159–160, 2008. Disponível em: <http://www.jstatsoft.org/v28/i05/>

KUHN, M.; JOHNSON, K. **Applied Predictive Modeling**. New York, NY: Springer New York, 2013. v. 5 Disponível em: <http://dx.doi.org/10.1016/j.is.2015.04.004>

LACERDA, D. P.; RODRIGUES, L. H.; SILVA, A. C. Da. Uma abordagem de avaliação de processos baseados no mundo dos custos para processos no mundo dos ganhos em instituições de ensino superior. **Gestão & Produção**, [s. l.], v. 16, n. 4, p. 584–597, 2009. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0104-530X2009000400008&lang=pt>

LEITNER, P. et al. Runtime Prediction of Service Level Agreement Violations for Composite Services. In: **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**. [s.l: s.n.]. v. 6275 LNCSp. 176–186.

MANS, R. et al. Application of Process Mining in Healthcare – A Case Study in a Dutch Hospital. **Communications in Computer and Information Science**, [s. l.], v. 127, p. 224–237, 2011. Disponível em: <http://www.scopus.com/inward/record.url?eid=2-s2.0-79952525388&partnerID=tZOtx3y1>

MENDLING, J. et al. Challenges of smart business process management: An introduction to the special issue. **Decision Support Systems**, [s. l.], v. 100, p. 1–5, 2017. Disponível em: <http://dx.doi.org/10.1016/j.dss.2017.06.009>

MIGUEL, P. A. C. Estudo de caso na engenharia de produção: estruturação e recomendações para sua condução. **Produção**, [s. l.], v. 17, n. 1, p. 216–229, 2007.

NORAMBUENA, B. K.; ZEPEDA, V. V. Minería de procesos de software: Una revisión de experiencias de aplicación. **RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao**, [s. l.], n. 21, p. 51–66, 2017.

POLATO, M. et al. Time and activity sequence prediction of business process instances. **Computing**, [s. l.], v. 100, n. 9, p. 1005–1031, 2018. Disponível em: <https://doi.org/10.1007/s00607-018-0593-x>

POSPÍŠIL, M. et al. Process Mining in a Manufacturing Company for Predictions

and Planning. **International Journal on Advances in Software**, [s. l.], v. 6, n. 3 & 4, p. 283–297, 2013.

PRAVILOVIC, S.; APPICE, A.; MALERBA, D. Process mining to forecast the future of running cases. **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**, [s. l.], v. 8399 LNAI, p. 67–81, 2014.

REIJERS, H. Case prediction in BPM systems: a research challenge. **Journal of the Korean Institute of Industrial Engineers**, [s. l.], v. 33, n. 1, p. 1–10, 2006.

ROSEMANN, M.; VOM BROCKE, J. The Six Core Elements of Business Process Management. In: **Handbook on Business Process Management 1**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 107–122.

SENDEROVICH, A. et al. Queue mining for delay prediction in multi-class service processes. **Information Systems**, [s. l.], v. 53, p. 278–295, 2015. Disponível em: <http://dx.doi.org/10.1016/j.is.2015.03.010>

SIHA, S. M.; SAAD, G. H. Business process improvement: empirical assessment and extensions. **Business Process Management Journal**, [s. l.], v. 14, n. 6, p. 778–802, 2008. Disponível em: <http://www.emeraldinsight.com/doi/10.1108/14637150810915973>

SILVA, E. L.; MENEZES, E. M. Metodologia da Pesquisa e Elaboração de Dissertação - 4a edição. **Portal**, [s. l.], n. January 2005, p. 138p, 2005.

THE WORLD BANK GROUP. **World Development Indicators: Structure of output**. 2017. Disponível em: <http://wdi.worldbank.org/table/4.2>.

TIWARI, A.; TURNER, C. J.; MAJEED, B. A review of business process mining: state-of-the-art and future trends. **Business Process Management Journal**, [s. l.], v. 14, n. 1, p. 5–22, 2008. Disponível em: <http://www.emeraldinsight.com/doi/10.1108/14637150810849373>

TURNER, C. J. et al. Process mining: from theory to practice. **Business Process Management Journal**, [s. l.], v. 18, n. 3, p. 493–512, 2012. Disponível em: <http://www.emeraldinsight.com/doi/10.1108/14637151211232669>

VAN DER AALST, W. Process Mining: Overview and Opportunities. [s. l.], v. 99, n. 99, 2012. Disponível em: <http://arxiv.org/abs/1710.03346>

VAN DER AALST, W. Spreadsheets for business process management: Using process mining to deal with “events” rather than “numbers”? **Business Process Management Journal**, [s. l.], v. 24, n. 1, p. 105–127, 2018.

VAN DER AALST, W. M. P. et al. Workflow mining: A survey of issues and approaches. **Data and Knowledge Engineering**, [s. l.], v. 47, n. 2, p. 237–267, 2003.

VAN DER AALST, W. M. P. **Process Mining**. [s.l: s.n.]. v. 5 Disponível em: <http://www.springerlink.com/index/10.1007/978-3-642-19345-3>

VAN DER AALST, W. M. P.; LA ROSA, M.; SANTORO, F. M. Business process management: Don't forget to improve the process! **Business and Information Systems Engineering**, [s. l.], v. 58, n. 1, p. 1–6, 2016.

VAN DER AALST, W. M. P.; SCHONENBERG, M. H.; SONG, M. Time prediction based on process mining. **Information Systems**, [s. l.], v. 36, n. 2, p. 450–475, 2011. Disponível em: <http://dx.doi.org/10.1016/j.is.2010.09.001>

VAN DER AALST, W. M. P.; WEIJTERS, T.; MARUSTER, L. Workflow Mining: Discovering Process Models from Event Logs. **IEEE TKDE**, [s. l.], v. 16, n. 9, p. 1128–1142, 2004.

VAN ECK, N. J.; WALTMAN, L. Software survey: VOSviewer, a computer program for bibliometric mapping. **Scientometrics**, [s. l.], v. 84, n. 2, p. 523–538, 2010.

WESKE, M. **Business Process Management**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. Disponível em: <http://link.springer.com/10.1007/978-3-540-73522-9>

WICKHAM, H. Tidy Data. **Journal of Statistical Software**, [s. l.], v. 59, n. 10, 2014. Disponível em: <http://www.jstatsoft.org/v59/i10/>

YIN, R. K. **Case Study Research: Design and Methods**. [s.l.] : Sage Publications, 1994. v. 2 Disponível em: <http://linkinghub.elsevier.com/retrieve/pii/S0002822310014811>

APÊNDICE A – CÓDIGO PARA AGREGAÇÃO DE ATIVIDADES (VBA)

```
Public Sub percorrer()
Dim atual As Long, prox As Long, activity As String, roww As Long, neww As String, newvalue
As String, oldvalue As String
Range("K2").Select
Do Until IsEmpty(ActiveCell)
roww = ActiveCell.Row
activity = Range("K" & roww).Value
newvalue = Range("M" & roww).Value
oldvalue = Range("L" & roww).Value
   Select Case activity
      Case Is = "Action Requested"
         Range("N" & roww).Select
         ActiveCell.Value = "edit"
         Range("K" & roww).Select

      Case Is = "Issue"
         Range("N" & roww).Select
         ActiveCell.Value = "edit"
         Range("K" & roww).Select

      Case Is = "Contact Name"
         Range("N" & roww).Select
         ActiveCell.Value = "edit"
         Range("K" & roww).Select

      Case Is = "First Call Resolution?"
         Range("N" & roww).Select
         ActiveCell.Value = "edit"
         Range("K" & roww).Select

      Case Is = "Severity"
         Range("N" & roww).Select
         ActiveCell.Value = "edit"
         Range("K" & roww).Select
```

```
Case Is = "Subject"
    Range("N" & roww).Select
    ActiveCell.Value = "edit"
    Range("K" & roww).Select

Case Is = "Request Type"
    Range("N" & roww).Select
    ActiveCell.Value = "edit"
    Range("K" & roww).Select

Case Is = "Hours Worked"
    Range("N" & roww).Select
    ActiveCell.Value = "edit"
    Range("K" & roww).Select

Case Is = "Priority"
    Range("N" & roww).Select
    ActiveCell.Value = "edit"
    Range("K" & roww).Select

Case Is = "Parent Case"
    Range("N" & roww).Select
    ActiveCell.Value = "edit"
    Range("K" & roww).Select

Case Is = "Created."
    Range("N" & roww).Select
    ActiveCell.Value = "create"
    Range("K" & roww).Select

Case Is = "Case Owner"
    If newvalue <> "Global Internal Transfer" And newvalue <> "Customer Support" And
newvalue <> "PMO Support" And newvalue <> "Customer Support_EU" And newvalue <>
"PMO Support_B" And newvalue <> "Production Engineering" And newvalue <> "Supplier
CAA" And newvalue <> "PMO Support_EU" Then
        Range("N" & roww).Select
        ActiveCell.Value = "grab"
```

```
        ElseIf oldvalue = "Integration" Or (oldvalue = "" And (newvalue = "Customer Support" Or newvalue = "PMO Support" Or newvalue = "Customer Support_EU" Or newvalue = "PMO Support_B" Or newvalue = "PMO Support_EU")) Then
            Range("N" & roww).Select
            ActiveCell.Value = "queued"
        Else
            Range("N" & roww).Select
            ActiveCell.Value = "reassigned"
        End If

      Case Is = "Status"
        If newvalue = "Closed_Duplicate" Or newvalue = "Closed_User Request" Or newvalue = "Closed" Then
            Range("N" & roww).Select
            ActiveCell.Value = "closed"

        ElseIf (oldvalue = "Awating Internal Response" And newvalue = "In Progress") Or (oldvalue = "client_reopen" And newvalue = "Awating Client Response") Or (oldvalue = "In Progress" And newvalue = "Awating Client Response") Then
            Range("N" & roww).Select
            ActiveCell.Value = "change_status"

        ElseIf oldvalue = "reassigned" And newvalue = "In Progress" Then
            Range("N" & roww).Select
            ActiveCell.Value = "restart_working"

        Else
            Range("N" & roww).Select
            ActiveCell.Value = newvalue
        End If

   End SelectRange("K" & roww).Select
ActiveCell.Offset(1, 0).Select
Loop

End Sub
```

APÊNDICE B – GERAÇÃO DO *EVENT LOG* (R)

```
library(bupaR)
library(tidyverse)
library(lubridate)

mining <- read.csv("C:\\Users\\EDUARDO\\Desktop\\Programming\\R\\projeto_r
\\dados\\log_tratado.csv", sep = ";", stringsAsFactors = FALSE)

#Verificar estrutura do Data Frame
str(mining)

#Padronizar timestamps
mining$timestamp <- mdy_hm(mining$timestamp, locale="English")

#Definir fatores
mining$case_number <- as.factor(mining$case_number)
mining$resource <- as.factor(mining$resource)
mining$adjustment <- as.character(mining$adjustment)
mining$status <- as.factor(mining$status)
mining$history_id <- as.factor(mining$history_id)

#Substituir caracteres especiais
str_replace(mining$adjustment, "-", "_")

#Padronizar o número de caracteres
mining$resource <- str_pad(mining$resource, width = 2, side = "left", pad
= "0")
mining$case_number <- str_pad(mining$case_number, width = 4, side = "left"
, pad = "0")

mining <- mining %>% group_by(case_number) %>% arrange(case_number, timest
amp) #Ordenar o Data Frame

#Definição do event log
logg <- eventlog(eventlog = mining, case_id = "case_number",
                 activity_instance_id = "history_id",
                 activity_id = "adjustment",
                 timestamp = "timestamp",
                 lifecycle_id = "status",
                 resource_id = "resource")

#Verificar estrutura do event log
str(logg)
```

APÊNDICE C – MODELOS DESCRITIVOS (R)

```r
#Verificar estrutura do event log
str(logg)

#Resultados descritivos
n_cases(logg)
n_activities(logg)
logg %>% start_activities()
logg %>% end_activities()
n_events(logg)
n_resources(logg)
n_traces(logg)
logg %>% group_by_activity %>% n_cases
ungroup_eventlog(logg)
logg %>% start_activities()
logg %>% end_activities()

#Gráfico: Tempo de atravessamento
logg %>% throughput_time(level="case", units = "mins") %>% plot()

#Matriz de predeceção
processmapR::precedence_matrix(logg) %>% plot

#Mapas do processo
process_map(logg)
logg %>%  filter_endpoints(start_activities = "create",
                           end_activities = "closed") %>% process_map(rank
dir = "TB",
                                                                      type
= performance())
logg %>%  filter_endpoints(start_activities = "create",
                           end_activities = "closed") %>% process_map(rank
dir = "TB")

#Caminhos
trace_explorer(logg)
logg %>% group_by_case() %>% first_n(7) %>% trace_explorer(cov = 0.8491)
ungroup_eventlog(logg)

logg %>% group_by_case() %>% last_n(4) %>% trace_explorer(cov = 0.8491)
ungroup_eventlog(logg)

#Gráficos de pontos
dotted_chart(x = "relative", logg, units = "mins", sort = "duration")
dotted_chart(x = "absolute", logg, units = "mins", sort = "start")
dotted_chart(x = "relative", logg, units = "mins", sort = "start")

#Graficos de atividades
logg$adjustment <- as.factor(logg$adjustment)
logg %>% activity_frequency(level = "activity") %>% plot()
logg %>% activity_presence() %>% plot()
```

```
#Gráfico: Tempo de atravessamento por dia
Dia_da_semana <- ifelse(day(logg$timestamp)==23,"Domingo",
                        (ifelse(day(logg$timestamp)==24, "Segunda-feira",
                            (ifelse(day(logg$timestamp)==25, "Terça-feira",
                                (ifelse(day(logg$timestamp)==26, "Quarta-fei
ra",
                                    (ifelse(day(logg$timestamp)==27, "Quint
a-feira",
                                        (ifelse(day(logg$timestamp)==28,"Se
xta-feira","Sábado")))))))))
logg$Dia_da_semana <- Dia_da_semana
dia_unico <- logg$adjustment == "create"
logg$dia <- ifelse(dia_unico == "TRUE",1, 0)

argument_1 <- logg  %>% throughput_time(level = "case", units = "mins")

novo <- logg[logg$dia==1,]
novo$Tempo_de_atravessamento_mins <- argument_1[,2]
novo$Dia_da_semana <- as.factor(novo$Dia_da_semana)

novo$Dia_da_semana <- ordered(novo$Dia_da_semana, levels = c("Domingo",
                                                    "Segunda-feira",
                                                    "Terça-feira",
                                                    "Quarta-feira",
                                                    "Quinta-feira",
                                                    "Sexta-feira",
                                                    "Sábado"))

novo %>% ggplot(aes(x=Dia_da_semana,
                    y=Tempo_de_atravessamento_mins),
                    xlab = "Dia da semana",
                    ylab = "Tempo de atravessamento (minutos)")
+ geom_boxplot(outlier.colour="red", outlier.size=2) + stat_summary(fun.y
= mean, geom = "point", shape = 23, size = 4)

#Gráfico: Cases por hora
novo$hour <- hour(novo$timestamp)
novo %>% group_by(hour) %>% count(hour) %>% plot(type= "h", ylab="Quantida
de de cases", xlab= "Hora")

#Gráfico: Tempo de atravessamento por hora
novo$hour <- as.factor(novo)
novo %>% group_by(hour) %>% ggplot(aes(x=hour,y=Tempo_de_atravessamento_mi
ns), xlab= "Hora") + geom_boxplot(outlier.colour="red", outlier.size=2) +
stat_summary(fun.y = mean, geom = "point", shape = 23, size = 4)
```

APÊNDICE D – ANÁLISE DE CORRELAÇÃO (R)

```r
# IRT: Descobre correlações maiores que 0.75
descrCor <- cor(train_irt)
corrplot(descrCor, type ="upper" )
summary(descrCor[upper.tri(descrCor)])
highlyCorDescr <- findCorrelation(descrCor, cutoff = .75)

# Plota o gráfico das variáveis correlacionadas
variaveis_correlacionadas <- train_irt[,+highlyCorDescr]
str(variaveis_correlacionadas)
pvalor <- cor(variaveis_correlacionadas)
corrplot(pvalor, method = "number")

# Plota o gráfico das variáveis descorrelacionadas
train_irt <- train_irt[,-(12:13)]
descrCor2 <- cor(train_irt)
summary(descrCor2[upper.tri(descrCor2)])

# FECHAMENTO: Descobre correlações maiores que 0.75
descrCor_atravessamento <- cor(train_atravessamento)
corrplot(descrCor_atravessamento, type ="upper" )
summary(descrCor_atravessamento[upper.tri(descrCor)])
highlyCorDescr_atravessamento <- findCorrelation(descrCor_atravessamento,
cutoff = .75)
head(highlyCorDescr_atravessamento)

# Plota o gráfico das variáveis correlacionadas
variaveis_correlacionadas_atravessamento <- train_atravessamento[,+highlyC
orDescr_atravessamento]
str(variaveis_correlacionadas_atravessamento)
pvalor_atravessamento <- cor(variaveis_correlacionadas_atravessamento)
corrplot(pvalor_atravessamento, method = "number")

# Plota o gráfico das variáveis descorrelacionadas
train_atravessamento <- train_atravessamento[,-(11:12)]
descrCor2_atravessamento <- cor(train_atravessamento)
summary(descrCor2[upper.tri(descrCor2_atravessamento)])
```

APÊNDICE E – SELEÇÃO DAS VARIÁVEIS (R)

```
library(caret)
library(Boruta)
library(tidyverse)

# Garante que não há dados faltantes
data <- na.omit(data)

# Divide os dados_irt entre treinamento e teste
set.seed(123)
trainRowNumbers <- createDataPartition(dados_irt$irt, p=0.8, list=FALSE)
train_irt <- dados_irt[trainRowNumbers,]
test_irt <- dados_irt[-trainRowNumbers,]
y = train_irt$irt
train_irt <- train_irt[,-(12:13)]
descrCor2 <- cor(train_irt)
summary(descrCor2[upper.tri(descrCor2)])

#Boruta package
train_irt$irt <- as.factor(train_irt$irt)
train_irt$nacionalidade <- as.factor(train_irt$nacionalidade)
train_irt$dia <- as.factor(train_irt$dia)
train_irt$time <- as.factor(train_irt$time)
train_irt$tipo_user <- as.factor(train_irt$tipo_user)
train_irt$problema <- as.factor(train_irt$problema)
train_irt$idioma <- as.factor(train_irt$idioma)
train_irt$hora <- as.factor(train_irt$hora)
train_irt$release <- as.factor(train_irt$release)
train_irt$volume_anterior <- as.integer(train_irt$volume_anterior)
train_irt$volume_hora <-as.integer(train_irt$volume_hora)
train_irt$wip <- as.integer(train_irt$wip)
train_irt$recurso_dia <- as.integer(train_irt$recurso_dia)

set.seed(111)
selecao_boruta <- Boruta(irt~., data = train_irt, doTrace = 2)
print(selecao_boruta)
plot(selecao_boruta, cex.axis=.7, las=2, xlab="", main="Variable Importanc
e")

# Variable Selection
parametros_selecionados_boruta = getSelectedAttributes(seleção_boruta)
tabela_resultados_boruta = arrange(cbind(attr=rownames(attStats(seleção_bo
ruta)), attStats(seleção_boruta)),desc(medianImp))

# Recursive feature selection
set.seed(100)
options(warn=-1)
subsets_5 <- c(1:4, 6, 8, 10, 12)

ctrl <- rfeControl(functions = rfFuncs,
                   method = "cv",
                   number = 10,
```

```
                    verbose = FALSE)

cv5_12var_10folds_5subsets <- rfe(x=train_irt[, 2:13], y=train_irt$irt,
                                  sizes = subsets_5,
                                  rfeControl = ctrl)

plot(cv5_12var_10folds_5subsets)

# Ajuste com as novas variáveis
head(test_irt)
test_ajustado <- test_irt[,-c(12:13)]
test_irt$recurso <- NULL
head(test_ajustado)
head(train_irt)
train_ajustado = train_irt
head(train_ajustado)
```

APÊNDICE F – MODELOS PREDITIVOS (R)

```r
library(caret)
library(e1071)

#Random Forests
set.seed(277)
train_rf = train_ajustado
test_rf = test_ajustado
train_rf$irt <- as.factor(train_rf$irt)
train_rf$nacionalidade <- as.factor(train_rf$nacionalidade)
train_rf$time <- as.factor(train_rf$time)
train_rf$dia <- as.factor(train_rf$dia)
train_rf$tipo_user <- as.factor(train_rf$tipo_user)
train_rf$problema <- as.factor(train_rf$problema)
train_rf$idioma <- as.factor(train_rf$idioma)
train_rf$release <- as.factor(train_rf$release)
train_rf$hora <- as.integer(train_rf$hora)
train_rf$recurso_dia <- as.integer(train_rf$recurso_dia)
train_rf$volume_anterior <- as.integer(train_rf$volume_anterior)
train_rf$volume_hora <- as.integer(train_rf$volume_hora)
train_rf$wip <- as.integer(train_rf$wip)
levels(train_rf$irt) <- make.names(levels(factor(train_rf$irt)))

test_rf$irt <- as.factor(test_rf$irt)
test_rf$nacionalidade <- as.factor(test_rf$nacionalidade)
test_rf$time <- as.factor(test_rf$time)
test_rf$dia <- as.factor(test_rf$dia)
test_rf$tipo_user <- as.factor(test_rf$tipo_user)
test_rf$problema <- as.factor(test_rf$problema)
test_rf$idioma <- as.factor(test_rf$idioma)
test_rf$release <- as.factor(test_rf$release)
test_rf$hora <- as.integer(test_rf$hora)
test_rf$recurso_dia <- as.integer(test_rf$recurso_dia)
test_rf$volume_anterior <- as.integer(test_rf$volume_anterior)
test_rf$volume_hora <- as.integer(test_rf$volume_hora)
test_rf$wip <- as.integer(test_rf$wip)
levels(test_rf$irt) <- make.names(levels(factor(test_rf$irt)))

control_rf_irt <- trainControl(method = 'cv', number = 5, savePredictions
= 'final', classProbs = T, summaryFunction = twoClassSummary)

model_rf_irt = train(irt~., data=train_rf, method = 'rf', trControl = cont
rol_rf_irt, tuneLength = 6)
model_rf_irt
fitted_rf_irt <- predict(model_rf_irt)
predicted_rf_irt <- predict(model_rf_irt, test_rf)
confusionMatrix(reference = test_rf$irt, data = predicted_rf_irt, mode='ev
erything')
var_imp_rf_irt <- varImp(model_rf_irt)
plot(var_imp_rf_irt, main="Importância das variáveis com Random Forests -
IRT")
```

```
#Support Vector Machines
train_svm_irt = train_ajustado
test_svm_irt = test_ajustado
train_svm_irt$irt <- as.factor(train_svm_irt$irt)
train_svm_irt$nacionalidade <- as.factor(train_svm_irt$nacionalidade)
train_svm_irt$time <- as.factor(train_svm_irt$time)
train_svm_irt$dia <- as.factor(train_svm_irt$dia)
train_svm_irt$tipo_user <- as.factor(train_svm_irt$tipo_user)
train_svm_irt$problema <- as.factor(train_svm_irt$problema)
train_svm_irt$idioma <- as.factor(train_svm_irt$idioma)
train_svm_irt$release <- as.factor(train_svm_irt$release)
train_svm_irt$hora <- as.integer(train_svm_irt$hora)
train_svm_irt$recurso_dia <- as.integer(train_svm_irt$recurso_dia)
train_svm_irt$volume_anterior <- as.integer(train_svm_irt$volume_anterior)
train_svm_irt$volume_hora <- as.integer(train_svm_irt$volume_hora)
train_svm_irt$wip <- as.integer(train_svm_irt$wip)
levels(train_svm_irt$irt) <- make.names(levels(factor(train_svm_irt$irt)))

test_svm_irt$irt <- as.factor(test_svm_irt$irt)
test_svm_irt$nacionalidade <- as.factor(test_svm_irt$nacionalidade)
test_svm_irt$time <- as.factor(test_svm_irt$time)
test_svm_irt$dia <- as.factor(test_svm_irt$dia)
test_svm_irt$tipo_user <- as.factor(test_svm_irt$tipo_user)
test_svm_irt$problema <- as.factor(test_svm_irt$problema)
test_svm_irt$idioma <- as.factor(test_svm_irt$idioma)
test_svm_irt$release <- as.factor(test_svm_irt$release)
test_svm_irt$hora <- as.integer(test_svm_irt$hora)
test_svm_irt$recurso_dia <- as.integer(test_svm_irt$recurso_dia)
test_svm_irt$volume_anterior <- as.integer(test_svm_irt$volume_anterior)
test_svm_irt$volume_hora <- as.integer(test_svm_irt$volume_hora)
test_svm_irt$wip <- as.integer(test_svm_irt$wip)
levels(test_svm_irt$irt) <- make.names(levels(factor(test_svm_irt$irt)))

model_svm_irt = train(irt~., data=train_svm_irt, method = 'svmRadial', trC
ontrol = control_rf_irt, tuneLength = 6)
model_svm_irt
fitted_svm_irt <- predict(model_svm_irt)
predicted_svm_irt <- predict(model_svm_irt, test_svm_irt)
confusionMatrix(reference = test_svm_irt$irt, data = predicted_svm_irt, mo
de='everything')
var_imp_svm_irt <- varImp(model_svm_irt)
plot(var_imp_svm_irt, main="Importância das variáveis com Support Vector M
achines - IRT")
```

APÊNDICE G – ANÁLISE DAS VARIÁVEIS (R)

```r
library(DALEX)

# Criação de função e definição dos vetores y
p_fun <- function(object, newdata){predict(object, newdata=newdata, type="
prob")[,2]}

ytest_rf <- as.numeric(test_rf$irt)
ytest_svm <- as.numeric(test_svm_irt$irt)

# Explainers
explainer_rf_irt <- DALEX::explain(model = model_rf_irt, label = "Random F
orests - IRT", data = test_rf[,2:13], y=ytest_rf, predict_function = p_fun
)
explainer_svm_irt <- DALEX::explain(model = model_svm_irt, label = "Suppor
t Vector Machines - IRT", data = test_svm_irt[,2:13], y=ytest_svm, predict
_function =  p_fun)

# Model Performance
mp_rf_irt <- model_performance(explainer_rf_irt)
plot(mp_rf_irt)
mp_svm_irt <- model_performance(explainer_svm_irt)
plot(mp_svm_irt)

# Importância das variáveis
vi_rf_irt <- variable_importance(explainer_rf_irt, type = "raw")
vi_svm_irt <- variable_importance(explainer_svm_irt, type = "raw")
plot(vi_rf_irt, vi_svm_irt)

# Variable Response - RF
vr_rf_nacionalidade_irt <- variable_response(explainer_rf_irt, type = "fac
tor", variable = "nacionalidade")
plot(vr_rf_nacionalidade_irt)
vr_rf_time_irt <- variable_response(explainer_rf_irt, type = "factor", var
iable = "time")
plot(vr_rf_time_irt)
vr_rf_dia_irt <- variable_response(explainer_rf_irt, type = "factor", vari
able = "dia")
plot(vr_rf_dia_irt)
vr_rf_tipo_irt <- variable_response(explainer_rf_irt, type = "factor", var
iable = "tipo_user")
plot(vr_rf_tipo_irt)
vr_rf_problema_irt <- variable_response(explainer_rf_irt, type = "factor",
variable = "problema")
plot(vr_rf_problema_irt)
vr_rf_idioma_irt <- variable_response(explainer_rf_irt, type = "factor", v
ariable = "idioma")
plot(vr_rf_idioma_irt)
vr_rf_release_irt <- variable_response(explainer_rf_irt, type = "factor",
variable = "release")
plot(vr_rf_release_irt)
vr_rf_hora_irt <- variable_response(explainer_rf_irt , type = "pdp", varia
```

```
ble = "hora")
vr_rf_volumehora_irt <- variable_response(explainer_rf_irt , type = "pdp",
variable = "volume_hora")
vr_rf_recurso_irt <- variable_response(explainer_rf_irt , type = "pdp", va
riable = "recurso_dia")
vr_rf_wip_irt <- variable_response(explainer_rf_irt , type = "pdp", variab
le = "wip")
vr_rf_volumeanterior_irt <- variable_response(explainer_rf_irt , type = "p
dp", variable = "volume_anterior")

# Variable Response - SVM
vr_svm_nacionalidade_irt <- variable_response(explainer_svm_irt, type = "f
actor", variable = "nacionalidade")
plot(vr_svm_nacionalidade_irt)
vr_svm_time_irt <- variable_response(explainer_svm_irt, type = "factor", v
ariable = "time")
plot(vr_svm_time_irt)
vr_svm_dia_irt <- variable_response(explainer_svm_irt, type = "factor", va
riable = "dia")
plot(vr_svm_dia_irt)
vr_svm_tipo_irt <- variable_response(explainer_svm_irt, type = "factor", v
ariable = "tipo_user")
plot(vr_svm_tipo_irt)
vr_svm_problema_irt <- variable_response(explainer_svm_irt, type = "factor
", variable = "problema")
plot(vr_svm_problema_irt)
vr_svm_idioma_irt <- variable_response(explainer_svm_irt, type = "factor",
variable = "idioma")
plot(vr_svm_idioma_irt)
vr_svm_release_irt <- variable_response(explainer_svm_irt, type = "factor"
, variable = "release")
plot(vr_svm_release_irt)
vr_svm_wip_irt <- variable_response(explainer_svm_irt, type = "pdp", varia
ble = "wip")
vr_svm_hora_irt <- variable_response(explainer_svm_irt, type = "pdp", vari
able = "hora")
vr_svm_volumehora_irt <- variable_response(explainer_svm_irt, type = "pdp"
, variable = "volume_hora")
vr_svm_recurso_irt <- variable_response(explainer_svm_irt, type = "pdp", v
ariable = "recurso_dia")
vr_svm_volumeanterior_irt <- variable_response(explainer_svm_irt, type = "
pdp", variable = "volume_anterior")

# Gráficos de dependência parcial
plot(vr_rf_wip_irt, vr_svm_wip_irt, color = "_label_", alpha = 0.7, size_p
oints = 6)
plot(vr_rf_hora_irt, vr_svm_hora_irt, color = "_label_", alpha = 0.7, size
_points = 6)
plot(vr_rf_volumehora_irt, vr_svm_volumehora_irt, color = "_label_", alpha
= 0.7, size_points = 6)
plot(vr_rf_recurso_irt, vr_svm_recurso_irt, color = "_label_", alpha = 0.7
, size_points = 6)
plot(vr_rf_volumeanterior_irt, vr_svm_volumeanterior_irt, color = "_label_
", alpha = 0.7, size_points = 6)
```