CEFET/RJ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO

KAIROS - MÉTODO DE DETECÇÃO E TRATAMENTO DE DESVIO DE CONCEITO PARA PREVISÃO DE ATRASOS DE VOOS

Leonardo Mosqueira de Carvalho

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Produção e Sistemas, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ, como parte dos requisitos necessários à obtenção do título de doutor.

Orientadores:
Eduardo Soares Ogasawara
Antônio Tadeu Azevedo Gomes

Rio de Janeiro,

16 de Agosto de 2023

Kairos - Método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos

Tese de Doutorado em Engenharia de Produção e Sistemas, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ.

Leonardo Mosqueira de Carvalho

Aprovada por:

Presidente, Prof. Eduardo Soares Ogasawara, D.Sc. (orientador)

Prof. Antônio Tadeu Azevedo Gomes, D.Sc. (coorientador)

Prof. Diego Moreira de Araujo Carvalho, D.Sc.

Prof. Geraldo Bonorino Xexéo, D.Sc.

Prof. André Carlos Ponce de Leon Ferreira de Carvalho, D.Sc.

Prof. Kelly Rosa Braghetto, D.Sc.

Rio de Janeiro,

16 de Agosto de 2023

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

C331 Carvalho, Leonardo Mosqueira de
Kairos - método de detecção ao e tratamento de desvio de conceito para previsão de atrasos de voos / Leonardo Mosqueira de Carvalho. — 2023.
113f. : il. color. , enc.

Tese (Doutorado) Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, 2023.
Bibliografia : f. 102-113
Orientador: Eduardo Soares Ogasawara
Coorientador: Antônio Tadeu Azevedo Gomes

1. Aeronáutica comercial. 2. Aeronáutica – Voos – Processamento de dados. 3. Aprendizado de máquina.4. Algoritmos. I. Ogasawara, Eduardo Soares. (Orient.). II. Gomes, Antônio Tadeu Azevedo (Coorient.). III.Título.

CDD 387.70

Elaborada pela bibliotecária Tania Mello – CRB/7 nº 5507/04

DEDICATÓRIA

Dedico este trabalho a todas as pessoas que me apoiaram e me encorajaram ao longo desta jornada acadêmica. À minha família, pelo amor incondicional, apoio e compreensão durante todos os momentos desafiadores. Vocês são minha fonte de inspiração e motivação constante. Aos meus amigos e colegas, pelo incentivo, troca de conhecimentos e momentos de descontração que tornaram essa caminhada mais leve e enriquecedora. Aos meus orientadores, pela orientação, paciência, sabedoria e confiança depositada em mim. Suas contribuições foram fundamentais para o desenvolvimento deste trabalho. A todos os professores e pesquisadores que compartilharam seu conhecimento e contribuíram para minha formação acadêmica, meu sincero agradecimento. E a todos aqueles que, de alguma forma, contribuíram para a realização deste trabalho, meu profundo agradecimento. Este trabalho é dedicado a vocês.

AGRADECIMENTOS

Primeiramente, gostaria de agradecer aos meus filhos Manuela e Bernardo, ao meu enteado Pedro, e a minha linda esposa Cris pelo amor incondicional, paciência e apoio ao longo desta jornada. Vocês foram minha fonte de inspiração e motivação, e sou grato por ter vocês ao meu lado. Aos meus orientadores, Eduardo Ogasawara e Antônio Tadeu Azevedo Gomes, minha profunda gratidão. Suas orientações, conhecimento, incentivo e dedicação foram fundamentais para o meu crescimento como pesquisador. Agradeço também pela confiança depositada em mim e pelas oportunidades de aprendizado proporcionadas ao longo deste trabalho. —Aos meus amados pais, minha avó - de quem sinto imensa saudade, ao meu querido irmão, cunhada e sobrinhos lindos, expresso minha profunda gratidão por todo o amor, apoio incondicional e valores que me transmitiram ao longo da vida.
Gostaria de agradecer especialmente à Petrobras, minha empresa empregadora, pela liberação e apoio concedidos para a realização deste trabalho.
Gostaria de estender meus agradecimentos a todos os professores e funcionários do CEFET/RJ, pela qualidade do ensino, infraestrutura e suporte oferecidos ao longo do meu percurso acadêmico. O ambiente acadêmico enriquecedor e o incentivo à pesquisa foram essenciais para o desenvolvimento deste trabalho.
Por fim, agradeço ao órgão de fomento, responsável por fomentar esta pesquisa.
Muito obrigado!

RESUMO

Kairos - Método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos

Leonardo Mosqueira de Carvalho

Orientadores:
Eduardo Soares Ogasawara
Antônio Tadeu Azevedo Gomes

Resumo da Tese submetida ao Programa de Pós-graduação em Engenharia de Produção e Sistemas do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ como parte dos requisitos necessários à obtenção do título de doutor.

Este trabalho foca na detecção e tratamento de desvio de conceito em modelos de previsão de atrasos de voos no domínio da aviação. O desvio de conceito se refere a mudanças nas distribuições de dados ao longo do tempo, o que pode afetar a eficácia dos modelos de previsão. O objetivo deste estudo foi desenvolver um método que pudesse detectar e lidar com tais desvios, visando melhorar a precisão e confiabilidade dos modelos de previsão de atrasos de voos.

Foi realizada uma extensa pesquisa sobre o estado da arte em desvio de conceito e sua aplicação no campo da aviação. Os métodos existentes foram considerados geralmente genéricos e não adaptados às características específicas da aviação. Como resultado, foi proposto um novo método chamado Kairos, que utiliza análise de distribuição e janelas deslizantes para detectar e tratar desvio de conceito. Experimentos comparativos foram então realizados, comparando o método Kairos com métodos de detecção de desvio de conceito do estado da arte. Os métodos ativos de identificação e tratamento de desvio mostraram-se mais eficazes do que o método passivo.

Ao longo dos experimentos, o Kairos apresentou um desempenho superior aos demais métodos ativos de detecção e tratamento de desvio de conceito em termos de desempenho. Os resultados foram consistentes em diferentes conjuntos de dados e perspectivas de análise, demonstrando a eficácia do método proposto.

Palavras-chave:
Desvio de conceito, Previsão de atrasos de voos, Modelos de aprendizado de máquina, Detecção de desvio de conceito, Tratamento de desvio de conceito

Rio de Janeiro,

16 de Agosto de 2023

ABSTRACT

Kairos - Concept Drift Detection and Treatment Method for Flight Delay Prediction

Leonardo Mosqueira de Carvalho

Advisors:
Eduardo Soares Ogasawara
Antônio Tadeu Azevedo Gomes

Abstract of tesis submitted to Programa de Pós-graduação em Engenharia de Produção e Sistemas - Centro Federal de Educação Tecnológica Celso Suckow da Fonseca CEFET/RJ as partial fulfillment of the requirements for the degree of doctor.

This work focuses on the detection and treatment of concept drift in flight delay prediction models in the aviation domain. Concept drift refers to changes in the data distributions over time, which can affect the effectiveness of prediction models. The objective of this study was to develop a method that could detect and handle such drifts, aiming to improve the accuracy and reliability of flight delay prediction models.

Extensive research was conducted on the state of the art in concept drift and its application in the aviation field. Existing methods were found to be generally generic and not tailored to the specific characteristics of aviation. As a result, a new method called Kairos was proposed, which leverages distribution analysis and sliding windows to detect and treat concept drift. Comparative experiments were then conducted, comparing the Kairos method with state-of-the-art concept drift detection methods. The active methods for drift identification and treatment were found to be more effective than passive methods such as retraining.

Throughout the experiments, Kairos consistently outperformed the other active concept drift detection and treatment methods in terms of performance. The results were consistent across different datasets and analysis perspectives, demonstrating the robustness and effectiveness of the proposed method.

Key-words:

Concept drift, Flight delay prediction, Machine learning models, Concept drift detection, Concept drift treatment

Rio de Janeiro,

16 de Agosto de 2023

Sumário

I Introdução **15**

I.1 Definição do problema 18

I.2 Objetivo 19

I.3 Estrutura da tese 20

II Predição de atrasos em voos **21**

II.1 Atrasos em voos 21

II.2 Métodos utilizados para análise de atrasos em voos 23

II.3 Aprendizado de máquina 24

III Desvio de Conceito **29**

III.1 Tipos 30

III.2 Drift real e virtual 31

III.3 Detecção de desvio de conceito 31

III.4 Modos de tratamento 34

III.5 Classificação dos métodos 35

III.5.1 Análise de distribuições 36

III.5.2 Análise de Desempenho 39

III.5.3 Discussão 45

IV Trabalhos Relacionados **47**

IV.1 Trabalhos relacionados ao desvio de conceito 47

IV.1.1 Análise das distribuições 48

IV.1.2 Análise de desempenho 54

IV.2 Desvio de conceito na predição de atrasos aéreos 57

IV.3 Análise 58

V Kairos - Método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos **62**

V.1 Visão geral do Kairos 62

V.2 Etapas detalhadas 64
V.3 Formalização dos principais conceitos do método 66
V.4 Algoritmos 71
V.5 Exemplo passo a passo 74
V.6 Implementação 77

VI Experimentos 79
VI.1 Pré-processamento 79
VI.2 Bases de dados para os experimentos 82
VI.3 Configuração e estrutura dos experimentos 83
VI.4 Forma de avaliação e armazenamento 89
VI.5 Diferentes parâmetros para os experimentos 90

VII Resultados 91
VII.1 Protocolos experimental e de dados 92
VII.1.1 Protocolo experimental 92
VII.1.2 Protocolo de dados 93
VII.2 Diferença entre métodos ativos, passivo e não tratamento 94
VII.3 Diferença entre diferentes dimensões de dados 95
VII.4 Diferença entre diferentes frequências de períodos de análise 95
VII.5 Estado da Arte para cenário de aviação 97
VII.6 Comparação do Kairos ao estado da arte para cenário de aviação 98
VII.7 Evolução do método por diferença entre distribuições 98
VII.8 Comparação do Kairos em diferentes bases de dados 99
VII.9 Comparação do Kairos em diferentes frequências de tempo 101
VII.10 Análises sobre número de desvios que ocorrem nos voos 101
VII.11 Avaliação sobre tempo decorrido 104

VIII Conclusões 106

A Pesquisa prévia 110
A.1 Metodologia reduzida aplicada 110
A.2 Experimento 112
A.3 Resultados 113
A.4 Discussão e direcionamento 114
Referências 116

Lista de Figuras

II.1 Dados para treinamento da Rede Neural (a); Exemplo básico usando dados de voo (b) 26

II.2 Dados para treinamento da Random Forest (a); Exemplo básico usando Random Forest (b) 27

II.3 Visão de processo de treinamento e teste de aprendizado de máquina. 28

II.4 Visão de aplicação de um modelo de Aprendizado de Máquina. 28

III.1 Tipos de Desvio. Adaptado de Gama et al. [2014]. 31

III.2 Desvio de Conceito real e virtual. Adaptado de Gama et al. [2014]. 31

III.3 Visão de um modelo de Aprendizado de Máquina e opções de detecção ativa de desvio de conceito. (A) Detecção pela análise das distribuições, (B) Detecção pela análise de indicadores de desempenho. 32

III.4 Janelas deslizantes. 33

III.5 Primeiras linhas da base de origem com dados de voo. 33

III.6 Lista cada uma das variáveis e valores de desvio. 34

III.7 Representação de diferentes modelos sendo adotados para diferentes janelas de tempo. 35

III.8 Estado da Arte em Desvio de Conceito. 36

III.9 Modelo de avaliação do FHDDM. 44

IV.1 Principais modelos de predição citados nas publicações de trabalhos relacionados. 59

IV.2 Principais métodos de detecção de desvio de conceito citados nas publicações de trabalhos relacionados. 60

V.1 Visão geral do método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos - Kairos. 63

V.2 Esquema de geral do método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos - Kairos. 65

V.3 Maior diferença KL apurada na fase de treinamento e testes. Primeiro registro na tabela de registros. 75

V.4 Diferença KL apurada no primeiro ciclo de processamento. Menor que o limite. 75
V.5 Diferença KL apurada em novo ciclo de processamento. Ultrapassou o limite. Tabela mantida, sem detecção do desvio. 76
V.6 Diferença KL apurada em novo ciclo de processamento. Ultrapassou o limite. Tabela de registros, após desvio detectado e modelo retreinado. 77
V.7 Com o desvio, o primeiro modelo foi testado e selecionado novamente. Distância KL estava no limite. Tabela de registros se manteve, pois o modelo não foi retreinado. 77

VI.1 Esquema do fluxo do pré-processamento realizado. 80
VI.2 Treinamento e teste do método Kairos. 85
VI.3 Processamento do Kairos a partir do terceiro mês. 86
VI.4 Retreina os modelos no Kairos. 86
VI.5 Registro de modelos treinados, janelas de comparação e distâncias limite. 86
VI.6 Retreina os modelos no Kairos. 87
VI.7 Fluxo de processamento dos modelos para os métodos ADWIN, DDM, HDDM, Page Hinkley e Média. 88
VI.8 Fluxo de retreinamento dos modelos para os métodos ADWIN, DDM, HDDM, Page Hinkley e Média. 88
VI.9 Estrutura básica de avaliação. 89

VII.1 Comparação entre a linha de base, método passivo e ativos. Box plot média mensal do F1 de todos os modelos - entre 2016 a 2018 - visão de malha aérea. 95
VII.2 Comparação entre a linha de base, métodos passivo e ativos na visão aeroportos. Box plot média mensal do F1 de todos os modelos - entre 2016 a 2018 - visão aeroportos. 96
VII.3 Comparação entre a linha de base, métodos passivo e ativos na visão de companhias. Box plot média mensal do F1 de todos os modelos - entre 2016 a 2018 - visão de companhias aéreas. 96
VII.4 Comparação entre a linha de base, métodos passivo e ativos na visão malha aérea, período semanal. Box plot média semanal do F1 de todos os modelos - entre 2016 a 2018 - visão de malha aérea. 97
VII.5 Comparação os métodos passivo, ativos ADWIN, HHDM, e Kairos. Box plot do valor médio do F1 entre Kairos e métodos do Estado da Arte. 98
VII.6 Comparação consolidada dos resultados dos métodos por *media* e o Kairos. Comparação entre média mensal do F1 dos modelos - entre 2016 a 2018 - visão de malha aérea. 99

VII.7 Comparação entre o método passivo, HDDM, ADWIN e Kairos na visão aeroporto. Box plot do valor médio do F1 do Kairos e Estado da Arte - entre 2016 a 2018 - visão de aeroportos. 100

VII.8 Comparação entre HDDM, ADWIN e Kairos na visão de companhias. Box plot do valor médio do F1 do Kairos e Estado da Arte - entre 2016 a 2018 - visão de companhias aéreas. 100

VII.9 Comparação entre o método passivo, ativos HDDM e ADWIN, e Kairos na frequência de tempo semanal. Plot da média semanal do F1 de todos os modelos - entre 2016 a 2018 - visão de malha aérea. 101

VII.10 Comparação do número de desvios de conceito registrados com uso do HDDM, ADWIN e Kairos. Número de desvios por método e modelo, base mensal, na visão de malha aérea." 103

VII.11 Comparação os métodos passivo, ativos ADWIN e HHDM, e Kairos, apenas com processamento do modelo de Redes Neurais. Box plot mensal do F1 de Redes Neurais - entre 2016 a 2018 - na visão de malha aérea. 104

VII.12 Comparação do tempo de processamento decorrido registrados com uso do HDDM, ADWIN e Kairos. Tempo decorrido (em segundos), base mensal, visão de malha aérea. 105

Lista de Tabelas

III.1 Tabela comparativa dos métodos 46

V.1 Notação matemática utilizada no método. 71

A.1 Parâmetros usados para o experimento 111

Lista de Abreviações

ADWIN	Adaptive Sliding Window	38
ANAC	Agência Nacional De Aviação Civil	15, 21
BTS	Bureau Of Transportation Statistics	16
CUSUM	Cumulative Sum	37
DDM	Drift Detection Method	40
EDDM	Early Drift Detection Method	40
EUROCONTROL	European Organization For The Safety Of Air Navigation	21
EWMA	Exponentially Weighted Moving Average	40
FAA	Federal Aviation Administration	21
FHDDM	Fast Hoeffding Drift Detection Method	42
HDDM	Drift Detection Methods Based On Hoeffding's Bound	42
IATA	International Air Transport Association	16
IOT	Internet Of Things	22
MDDM	McDiarmid Drift Detection Method	42
PH	Page-Hinkley	37
RDDM	Reactive Drift Detection Method	40
SEQDRIFT	Sequential Drift	38

Capítulo I Introdução

Dentre os sistemas de transporte, o aéreo é um dos mais críticos [Wu and Truong, 2014]. Neste, o atraso é um dos indicadores de desempenho mais analisados [IATA, 2021]. Embora haja diversas maneiras de avaliar os atrasos, na maioria das vezes, um limite de tolerância para o atraso é usado, seja na saída ou na chegada. No Brasil, por exemplo, este limite é de 15 minutos.

Atrasos em voos podem ter diversos impactos negativos para passageiros, companhias aéreas e aeroportos, prejudicando todo o sistema de transporte aéreo [Sathanur et al., 2019]. Dada a incerteza de sua ocorrência, os passageiros geralmente planejam viajar muitas horas antes para seus compromissos. Eles podem ter que aumentar seus custos de viagem para chegar a tempo. Por outro lado, as companhias aéreas sofrem penalidades, multas e custos adicionais de operação, como retenção de tripulantes e aeronaves em aeroportos. Além disso, do ponto de vista da sustentabilidade, atrasos também podem causar danos ambientais, aumentando o consumo de combustível e as emissões de gases [Rebollo and Balakrishnan, 2014].

Em relação às características que levam a atrasos de voos, o sistema aéreo brasileiro é muito semelhante aos principais sistemas do mundo. Podemos citar o clima como uma das principais variáveis associadas ao indicador de atraso de voo [Sternberg et al., 2016]. Contudo, existem vários outros fatores que adicionam complexidade às análises, como eventos pontuais, contextos sociais e políticos, sazonalidade, e até outros aspectos como, por exemplo, os que pudemos observar com a pandemia de COVID-19, que foi motivo de disrupção no sistema [EUROCONTROL, 2021].

Segundo a Agência Nacional de Aviação Civil (ANAC), no mercado de voos domésticos no Brasil, de janeiro a dezembro de 2020, foram transportados 45,2 milhões de passageiros domésticos. Isso representa uma queda acumulada de 52,5% em relação ao mesmo período de 2019. Em dezembro de 2020, houve 80,3% menos passageiros que o mesmo mês em 2019, tendo o total de passageiros em 408 mil [ANAC, 2021].

O mercado de voos internacionais, com partidas e chegadas no Brasil, acumulou quedas de 71% e 62,6% entre demanda e oferta, de janeiro a dezembro de 2020 se comparado a 2019. Durante todo o ano de 2020, foram transportados 6,75 milhões de passageiros no mercado internacional. Este número é 72% menor do que foi registrado no acumulado dos 12 meses do ano anterior.

Em dezembro de 2019, cerca de 79% dos voos chegaram sem atraso, com uma média geral de atrasos em minutos de 38,6 minutos. Em época ainda pré-pandêmica, em fevereiro de 2020, vemos como esses números ainda são próximos, com cerca de 84% dos voos no horário e um tempo médio

de atrasos de 37 minutos. Em maio de 2021, cerca de 92,8% dos voos chegou no horário, com uma média de 30 minutos de atraso. Com a redução de voos e passageiros em período de pandemia, existe uma clara mudança de patamares de voos que chegam em tempo e de média de atrasos [ANAC, 2021].

Dentre os últimos dados comparativos liberados para 2019, após dois anos seguidos de crescimento em percentuais de voos atrasados, houve pequena queda em 2019 de 3% em relação a 2018. Porém, muito maiores que os níveis registrados em 2016. Ainda em 2019, no mercado doméstico, foram realizados 804,9 mil voos (redução de 1,4%) e transportados 95,3 milhões de passageiros (alta de 1,8%). Houve queda de 1,7% na oferta e aumento de 1,14% na demanda, resultando em um aumento no aproveitamento das aeronaves, de 82,0% para 84,3% [ANAC, 2021]. Uma das poucas localidades onde a capacidade cresceu menos que a demanda foi a América Latina. Enquanto a demanda global aumentou 3,4% e a capacidade aumentou 5,4%, na América Latina a demanda aumentou 5,8% enquanto a capacidade aumentou apenas 3,4%, segundo o International Air Transport Association (IATA) [IATA, 2021].

Verificamos novamente um ano atípico, e contrariando qualquer expectativa, quando apuramos que as companhias aéreas dos EUA transportaram 557 milhões de passageiros a menos em 2020 do que em 2019, queda de 60% ano a ano. Para todo o ano de 2020, de janeiro a dezembro, as companhias aéreas dos EUA transportaram 369 milhões passageiros. Valor abaixo de 927 milhões em 2019 e 889 milhões em 2018. O tráfego de passageiros em 2020 foi o mais baixo em companhias aéreas dos EUA desde meados da década de 1980 [BTS, 2019]. Verificamos como 2020 foi um ano atípico quando vemos que seu patamar de atrasos em voos ficou em 9%, algo nunca registrado pelo Bureau of Transportation Statistics (BTS) [BTS, 2019]. O BTS estima que as companhias aéreas dos EUA transportaram mais passageiros em 2019 do que em qualquer ano anterior. Para todo o ano de 2019, de janeiro a dezembro, as companhias aéreas transportaram o maior total com base em registros comparáveis desde 2003. O número de passageiros em 2019 aumentou 4,1% em relação a 2018, um aumento de 9,0% em relação a 2017 e um aumento de 12,3% em relação a 2016.

Segundo o EUROCONTROL [2021], em termos de atraso geral, o atraso médio por voo no 1º trimestre de 2021 diminuiu em 1 minuto para 8,4 minutos por voo, enquanto os níveis de tráfego anuais estão cerca de 58% em comparação ao 1º trimestre de 2020. Espera-se que os níveis voltem ao patamar de 2019 apenas em 2024. No segundo trimestre de 2021, existe um alto desempenho na pontualidade, com 87,1% dos voos chegando dentro de 15 minutos, ou antes do horário programado de chegada. Tal desempenho é atribuído à diminuição de voos e à possibilidade de partidas antes do previsto, chegando, por sua vez, muitas vezes antes de seu tempo estimado para chegada. O motivador clima continua sendo um fator relevante para o atraso médio por voo. Essa contribuição aumentada de atraso de tempo de 1 minuto por voo foi impulsionada por eventos climáticos no

aeroporto (ventos fortes e neve) em janeiro e fevereiro de 2021.

A aplicação de modelos de aprendizado de máquina entram neste cenário como instrumento de avaliação e de tomada de decisões de operação, trajeto, escala, dentre outras, visando a correção ou melhoria do sistema aéreo. Por esta razão, muitos modelos preditivos foram desenvolvidos para este fim [Rong et al., 2015; Kim et al., 2016; Yu et al., 2019]. Os modelos de aprendizado de máquina se baseiam em dados, que no caso da aviação comercial, podem vir de diversas bases diferentes, como dados das operações de aeroportos, controles de tráfego, entidades reguladoras, companhias de aviação, e diversas outras como monitoramento e informações gerais sobre o clima.

Na aviação há um grande fluxo de dados relacionados a sistemas de monitoramento, operação e controles de voo. Há uma abundância de dados que podem orientar análises e decisões que possibilitam resolver ou mitigar as causas dos atrasos. Esses dados são a base para estruturação de modelos de análise de dados. Tais modelos são utilizados subsequentemente para prover informações preditivas e prescritivas a partir das novas informações entrantes. Entretanto, a assertividade desses modelos depende da manutenção da relação entre as novas informações entrantes e os dados que deram base ao modelo treinado. Nos casos onde essa relação muda para cenários que não fizeram parte do treinamento, o modelo pode não ser mais válido.

O Desvio de Conceito é a denominação dada a essa perda de aderência do modelo preditivo aos dados usados como base para o treinamento. A mudança nos dados em relação aos dados utilizados para o treinamento ocorre de tal forma que o modelo não consegue mais manter o mesmo nível de desempenho. Essas mudanças podem estar, por exemplo, relacionadas à escala do sistema (número de voos, passageiros e capacidade do aeroporto). Mesmo quando não há mudança na escala, a relação entre as variáveis do sistema de voo (variáveis de entrada) e atrasos (variável de saída) pode mudar [Iwashita and Papa, 2019].

O desvio de conceito pode ser caracterizado como Abrupto, Gradual, Incremental ou Recorrente [Iwashita and Papa, 2019]. O desvio abrupto é aquele em que existe uma disruptura brusca, onde um contexto é instantaneamente substituído por outro. O desvio gradual apresenta a mistura de dois contextos, um anterior e um mais novo, em determinado período, até que existe uma substituição pelo novo. No desvio incremental, as diferenças acontecem gradualmente, sendo percebidas apenas em longos períodos. O desvio recorrente se caracteriza pela variação entre contextos, onde existe uma alternância com certo grau de regularidade, como os ciclos das estações, ou datas específicas como o Dia dos Pais, Dia das Mães e Natal [Lu et al., 2019].

A literatura indica que a existência de desvio de conceito pode impactar os modelos preditivos da forma que apresentem queda de eficácia com o tempo ou até se tornem totalmente inadequados por um período, ou na sua totalidade, dependendo das mudanças ocorridas [Gama et al., 2014; Webb et al., 2016; Iwashita and Papa, 2019]. Para resolver este problema, na aviação, alguns estudos

testaram algoritmos que retreinam os modelos se um desvio ocorre, ou que podem se adaptar às mudanças, como redes neurais recorrentes [Khamassi and Sayed-Mouchaweh, 2014; Pesaranghader and Viktor, 2016; Kim et al., 2016]. Entretanto, o tema ainda se apresenta aberto e há lacunas a serem estudadas.

I.1 Definição do problema

Inicialmente, podemos dividir os modos de tratamento do desvio de conceito em dois, passivos e ativos. No modo passivo, os métodos são adaptados em intervalos regulares, independentemente de as mudanças terem ocorrido. Os exemplos incluem métodos que ponderam os exemplos de acordo com sua idade e métodos que usam janelas de tempo de tamanho fixo. Outro exemplo de uma forma passiva é o de modelos combinados, consistindo na existência prévia de mais de um modelo, gerando um modelo combinado. Desta forma, a possível perda de desempenho de um dos modelos pode ser mitigada pela combinação. No modo ativo, métodos apenas alteram o modelo de decisão após a detecção de um desvio. Eles são usados em conjunto com um modelo de detecção. Sempre que um novo lote de dados åé introduzido, a detecção de desvio é aplicada. Se nenhum desvio for detectado, o modelo treinado anteriormente ainda será usado. No entanto, se o desvio for detectado, um novo modelo é construído usando novas distribuições de dados.

Os modelos de aprendizado de máquina são geralmente estruturados com dados históricos que dão suporte a treinamento e teste. No entanto, há mudanças de contexto que tornam os resultados dos modelos menos precisos ao longo do tempo [Lu et al., 2019]. Essa mudança de contexto é inerente ao cenário de voo e, portanto, é crucial para prever atrasos de voos. Essa complexidade deve ser trazida aos modelos de tratamento, de forma ativa ou passiva, para serem mantidos com um nível mínimo de confiança, mitigando problemas operacionais e possibilitando melhorias no sistema de planejamento de voo, rotas e operações gerais. Diante deste cenário, verificamos que a adoção da detecção e tratamento do desvio de conceito mostra claro potencial de complementação aos modelos de aprendizagem de máquina para atrasos e voos [Kim et al., 2016].

As principais perguntas deste trabalho podem ser traduzidas em: "Qual é o impacto da detecção e tratamento de desvio de conceito na predição dos atrasos de voos na aviação e como isso se compara à abordagem tradicional sem considerar o desvio de conceito? Quais são as vantagens dos métodos ativos em relação aos passivos na detecção e tratamento do desvio de conceito nos atrasos de voos? Além disso, existe ganho na aplicação de um novo método ativo para detecção e tratamento de desvios de conceito no contexto de atrasos aéreos, que trate suas características, como comportamentos sazonais?"

Essas perguntas da tese visam investigar o impacto da detecção e tratamento de desvio de

conceito nos atrasos de voos na aviação, considerando a lacuna de estudos que até o momento têm se concentrado principalmente na abordagem tradicional, que não considera o desvio de conceito. Além disso, a pesquisa analisará as vantagens dos métodos ativos em relação aos passivos na detecção e tratamento do desvio de conceito nos atrasos de voos. O objetivo é explorar como o desvio de conceito pode afetar os atrasos de voos e se a utilização de métodos ativos pode proporcionar melhorias significativas na previsão e gestão dos atrasos em comparação com a abordagem tradicional. Essa análise permitirá identificar e compreender as vantagens dos métodos ativos em termos de detecção precoce, adaptação dinâmica e tomada de decisão em tempo real, visando à minimização dos atrasos e otimização do desempenho operacional na aviação.

I.2 Objetivo

O objetivo desta tese é desenvolver um modelo de detecção de desvio para uso na área de aviação em complementação aos modelos de predição de atrasos em voos. Verificamos casos limitados de aplicação de formas tradicionais de resolução de desvio de conceito, como aprendizado incremental, onde existe um ciclo de troca e incremento de contextos, e o retreinamento sem identificação de desvio, o qual chamamos de tratamento passivo neste trabalho. Em complemento, a maioria dos modelos de detecção de desvio propostos são de aplicação genérica, não existindo indicação de melhor aplicação por área de atuação.

O método proposto para a detecção de desvio de conceito é ativo e prévio, e ocorre através da análise das distribuições de dados entrantes no sistema de predição. O modelo é ativo no sentido de ser um modelo que tratará eventos à medida que identificar mudanças e prévio no sentido de ocorrer em momento anterior ao processamento do modelo de aprendizado de máquina que efetuará a predição. A hipótese é que o desvio de conceito pode ser identificado analisando diferentes contextos e distribuições dos dados usados como base para os modelos de predição, identificando previamente nesses modelos qual seria o melhor modelo já existente para seu processamento ou direcionar para um novo treinamento.

A novidade e contribuição no desenvolvimento desse trabalho está no fato de estarmos explorando um novo método de detecção de desvio de conceito, e em momento e forma diferentes dos métodos que hoje formam o estado da arte nesta disciplina. O modelo proposto objetiva atender a lacunas hoje não tratadas de forma ótima pelos demais modelos do estado da arte, como os desvios de conceito recorrentes. Por fim, o modelo proposto também visa trazer uma forma de detecção menos reativa, com menor necessidade de reprocessamento, buscando tratamento de cenários de dados em tempo real.

I.3 Estrutura da tese

Além deste capítulo de introdução, esta tese está subdividida em demais seis capítulos. No Capítulo II é feita uma explanação da área de aviação, dos modelos e métodos utilizados para estudo e predição de atrasos em voos. O Capítulo III apresenta o Desvio de Conceito, seus tipos, aplicações e estado da arte. O Capítulo IV descreve trabalhos relacionados da disciplina de Desvio de Conceito e sua aplicação, tanto abrangentemente, como na área de aviação. O Capítulo V estrutura o modelo de detecção de desvio, o detalha e apresenta como serão efetuados os cálculos e comparações. O Capítulo VI segue com o detalhamento da estrutura do modelo de detecção de desvio, e apresenta com maior nível de profundidade como foram efetuados os cálculos e comparações. Seguido pelo Capítulo VII, com os resultados obtidos e análises feitas a partir dos experimentos. E finalmente, a tese é finalizada com o Capítulo VIII, de Conclusões.

Adicionalmente, segue o Apêndice A, que descreve o histórico dos experimentos prévios feitos até a qualificação para embasar parte da tese, onde apresentamos passos e resultados preliminares que nos levaram às hipóteses correntes.

Capítulo II Predição de atrasos em voos

II.1 Atrasos em voos

A aviação comercial é um complexo sistema distribuído de transporte e responsável pelo transporte de milhares de pessoas todos os dias ao longo do globo. Este sistema demanda alto nível de gerenciamento para manutenção da sua eficiência. A aviação, por sua vez, influencia diretamente em várias outras áreas de atuação e na economia. Envolve diversas partes e atores, como os próprios passageiros, mas também pilotos, equipes de bordo e de operação aeroportuária [Ball et al., 2007].

Um dos problemas mais comuns no setor de aviação comercial é o atraso em voos, que merece atenção e análise de indicadores em qualquer área ou entidade que participe da operação ou controle do setor. Podemos definir o atraso em voos como um problema: (i) de classificação (regressão logística), na identificação de existência ou não de atraso; ou (ii) de regressão, por exemplo, para estimativa do tempo de atraso [Wieland, 1997].

Os atrasos podem ser verificados em diversos níveis e por diversas causas. Os passageiros, por exemplo, seguem seus próprios itinerários com cada um o seu planejamento. A modalidade de transporte aéreo pode ser apenas um dos passos nesse planejamento. Companhias aéreas têm toda a carga de panejamento de capacidade, horários para aeronaves, pilotos e comissários de bordo. As partes deste sistema incluem operações de terminais, aeroportos, pistas e gerenciamento do espaço aéreo [Britto et al., 2012].

Todos esses atores e variáveis são suscetíveis a diferentes tipos de atrasos em vários níveis. Alguns exemplos incluem problemas mecânicos, condições climáticas, atrasos no solo, controle de tráfego aéreo, filas de pista e restrições de capacidade [Xu et al., 2008]. Pilotos, comissários de bordo e aeronaves podem ter horários diferentes devido aos descansos exigidos por lei, deveres e planos de manutenção para aeronaves. Assim, qualquer interrupção no sistema pode influenciar nos voos subsequentes da mesma companhia aérea [Deshpande and Arikan, 2012]. Além disso, distúrbios podem causar congestionamento no espaço aéreo ou mesmo em outros aeroportos, criando filas e atrasando alguns voos de outras companhias aéreas [Xu et al., 2005; Rodríguez-Sanz et al., 2019].

Desta forma, estudar os atrasos nos voos é um assunto essencial para companhias aéreas, aeroportos, provedores de serviços de navegação aérea, entidades de controle como a ANAC [ANAC, 2021], a Federal Aviation Administration (FAA) [FAA, 2017] e o European Organization for the Safety of Air Navigation (EUROCONTROL) [EUROCONTROL, 2021].

Diferentes fontes podem induzir atrasos e afetar aeroportos, companhias aéreas ou toda a malha aérea, sendo importante distinguir um atraso originário de atrasos subsequentes ou propagados. Um atraso originário é causado por um evento gerador, que pode resultar da operação do aeroporto, tripulação, entre vários outros [Abdelghany et al., 2004].

Podemos diferenciar momentos ou níveis em que podem ocorrer atrasos. Por exemplo, o atraso na partida corresponde à diferença entre o horário original de partida para decolagem e o tempo real de partida. Dami and Yahaghizadeh [2018] propuseram uma forma baseada em Internet of Things (IOT) de prever eventos de atraso de partida.

O atraso na chegada é outra forma de verificar o atraso. Os estudos referentes à chegada discutem previsões, características e causas de atrasos que resultaram na diferença. Uma vez entendido que o voo começou sem demora, é possível destacar alguns fatores que levam ao atraso de chegada. Geralmente inclui o tempo na rota ou destino, bem como problemas de tráfego que podem afetar e causar o atraso de chegada após uma saída pontual [Abdel-Aty et al., 2007]. Chakrabarty [2019] analisou o atraso na chegada de voo usando quatro algoritmos supervisionados de aprendizagem de máquina. Além disso, Rodríguez-Sanz et al. [2019] definiram uma metodologia para prever e avaliar as condições operacionais do sistema de chegada do aeroporto.

Como um atraso na chegada ou partida pode gerar atrasos subsequentes, torna-se necessário também efetuar análise de propagação de atrasos. Nela, o objetivo é entender como ocorre essa propagação a partir de um atraso originário. Esse cenário acontece quando ocorrem atrasos que impactam outros voos da mesma companhia aérea, como reações em cadeia. Envolve previsão, decisões de custo-benefício, realocação de recursos, re-planejamento de rotas, relocação operacional, entre outros [Fleurquin et al., 2014, 2013]. Rodriguez-Sanz et al. [2018] estudaram o quão estáveis e confiáveis os transportadores podem ser para se recuperar da propagação de atrasos. Os atrasos também podem ser apresentados em diferentes níveis, como atrasos nos aeroportos, comumente associados à alocação de tripulantes, gerenciamento de bagagem, qualidade dos serviços, táxi, estacionamento, *service desk* e atrasos de passageiros [Xu et al., 2008].

Na visão da capacidade aeroportuária e do sistema, é essencial analisar os investimentos em infraestrutura para alcançar um equilíbrio competitivo entre oferta e demanda. A análise da mudança de capacidade desencadearia um conjunto complicado de ajustes e interações entre a demanda de passageiros, passagem aérea, frequência de voo, tamanho da aeronave e atrasos de voo, levando a uma mudança no equilíbrio. Chandramouleeswaran and Tran [2018] ilustram um procedimento para prever futuros atrasos nos aeroportos com base em recursos de tempo, congestionamento da rede e condições climáticas.

Outro nível em que se pode evidenciar atrasos é o nível de companhia aérea, onde é possível selecionar alguns fatores, como alocação de equipes, desembaraços, abastecimento e a porcentagem

de uso de aeronaves. Essa perspectiva filtra o estudo na visão de uma ou mais companhias aéreas, onde seus resultados visam identificar problemas, formas de resolução ou como melhorar o sistema [Rebollo and Balakrishnan, 2014; Schultz and Reitmann, 2019a]. Deshpande and Arikan [2012] analisaram atrasos das companhias aéreas, considerando o tempo de bloqueio programado alocado para os voos. Em um nível maior, podemos analisar o atraso na malha aérea, constituída de todos os itens que a formam, inclusive a rede de rotas. Du et al. [2018] mostraram que, no sistema aéreo, apenas alguns aeroportos impactam outros aeroportos em propagação de atrasos, e que todo o sistema pode ser afetado.

II.2 Métodos utilizados para análise de atrasos em voos

Nesta seção, apresentamos os diversos tipos de métodos utilizados para análise de atrasos em voos, seguindo uma linha do tempo. Para isso, utilizaremos como base a taxonomia apresentada por [Carvalho et al., 2021], onde os métodos podem ser classificados como: Análise Estatística, Classificação, Regressão, Análise de Redes, Agrupamentos (clustering), Mineração de Padrões e Aprendizado de Máquina. Para este último dedicaremos uma seção específica.

Modelos baseados em análises estatísticas são os modelos historicamente mais utilizados, que podem incluir análise de correlação, Monte-Carlo, testes paramétricos, testes não paramétricos e análise multivariada (MVA). Tais métodos são normalmente baseados em dados numéricos históricos para chegar a conclusões sobre o que ocorreu. Podemos ver um exemplo em Abdel-Aty et al. [2007], onde calcularam a média diária de atrasos para detectar correlações para entender as principais causas de atrasos no Aeroporto Internacional de Orlando.

Seguindo um pouco na linha do tempo, ainda se identifica o uso de métodos estatísticos. No entanto, esses trabalhos já incluem modelos de árvores condicionais e de decisão. Eles começam a analisar eventos causais, seus potenciais riscos, prováveis riscos residuais. Então, a partir dessa análise, permitem que os processos de tomada de decisão se concentrem em problemas com as probabilidades e impactos mais significativos [Post et al., 2006; Tu et al., 2008; Zonglei et al., 2008]. Classificação é o processo de criação de um modelo (classificador) a partir de dados. A partir de um conjunto de dados com um atributo categórico (rótulo de classe), um modelo é construído a partir de um subconjunto dos outros atributos do conjunto de dados para prever o rótulo de classe. A classificação é conhecida como aprendizagem supervisionada. Dois padrões de métodos estatísticos de classificação de aprendizagem comumente utilizados incluem árvores de decisão [Thiagarajan et al., 2017] e Naive Bayes [Choi et al., 2016; Yanying et al., 2019].

Encontramos também modelos usando teoria dos grafos para modelar a rede de comunicação de rotas entre aeroportos. Além disso, a partir dessa rede, implementar técnicas para resolver

problemas práticos. Por exemplo, a analise da proximidade entre os aeroportos conforme as rotas disponíveis, identificar gargalos ou pontos de concentração entre rotas [Liu and Willsky, 2013; Xu et al., 2015; Dai et al., 2018].

Outro método largamente utilizado é o agrupamento, que reúne técnicas para agrupar dados por métricas de similaridade. Os objetos em um *cluster* são semelhantes, mas diferem de objetos de outros agrupamentos [Shalev-Shwartz and Ben-David, 2013]. Nessa abordagem, é desconhecido um rótulo de classe, caracterizando a abordagem de aprendizagem não supervisionada. Estudos incluem Woodburn and Ryerson [2014], que aplicaram análises de agrupamento para simular como a implantação de uma quinta pista no Aeroporto Internacional de Hartsfield-Jackson Atlanta, na Geórgia, pode influenciar a previsibilidade das chegadas de voos.

A análise de regressão estuda a relação entre uma variável denominada variável dependente e outras variáveis denominadas variáveis independentes. A relação entre elas é representada por um modelo matemático e, havendo correlação entre as variáveis, pode fornecer previsão de comportamento futuro. Um dos modelos de regressão, o modelo de regressão linear simples, define uma relação linear entre a variável dependente e uma variável independente. Se várias variáveis independentes estiverem envolvidas, isso será chamado de modelo de regressão linear múltipla [Han et al., 2012]. Modelos de regressão são comumente usados para construir um modelo de regressão múltipla do sistema de voo [Post et al., 2006; Nayak and Zhang, 2011; Zhang and He, 2010]. Ding [2017] propõe um método para modelar usando regressão múltipla para prever o atraso de chegada. Vários outros estudos criaram modelos de regressão linear para previsão de atrasos [Scarpel and Pelicioni, 2018; Cheng and Xing, 2011; Nigam and Govinda, 2018; Wang et al., 2018; Diana, 2011; Cheng et al., 2016; Wang et al., 2016; Tejasviram et al., 2016; Xu et al., 2008].

Mais recentemente, estudos começaram a aparecer usando métodos de aprendizagem de máquina para simuladores e previsão de eventos [Chandramouleeswaran and Tran, 2018; Moreira et al., 2018; Feuser Fernandes and Müller, 2019]. Esses modelos podem utilizar um ou mais dos métodos mencionados até agora, mas se caracterizam como métodos adaptativos e que se baseiam em treinamento e teste dos modelos que se adéquam a diversos contextos e suas mudanças.

II.3 Aprendizado de máquina

Aprendizado de Máquina é o estudo de métodos que constroem um modelo a partir de dados. Ele usa dados de amostra, conhecidos como um conjunto de treinamento, para construir modelos de previsão. Modelos de aprendizagem de máquina tratam parâmetros por um conjunto de validação, e a avaliação real produzida usando o conjunto de teste. Nesta categoria, por exemplo, estão todas as Redes Neurais, como redes neurais recorrentes, redes neurais em cascata e aprendizagem

profunda (Deep Learning, como CNN e LSTM) [Pouyanfar et al., 2018].

Existem dois tipos principais de redes neurais artificiais: redes neurais artificiais de feedforward e feedback. A rede neural feedforward é uma rede na qual os neurônios de uma camada são conectados apenas aos neurônios da próxima camada, e eles não formam um ciclo. No feedforward, os sinais viajam em apenas uma direção em direção à camada de saída. As redes neurais de feedback contêm fluxos de comunicação em ambas as direções, introduzindo loops na rede. O comportamento pode mudar com o tempo, causado pelos diferentes ciclos. Rede neural de feedback, também conhecida como redes neurais recorrentes. A camada oculta onde ocorre o processamento por meio da aplicação de funções matemáticas e ajustes de peso precisa ser treinada para os resultados serem adequados aos dados de entrada [Schmidhuber, 2015].

Ai et al. [2019] utilizaram um modelo de Deep Learning (DL) - um LSTM convolucional (Conv-LSTM) visando prever os atrasos de um aeroporto na malha aérea. Choi et al. [2016] usaram a mineração de dados e algoritmos de aprendizagem de máquina supervisionado para prever atrasos nas companhias aéreas causados por condições climáticas adversas por meio de processamento e análise de big data. Além disso, [Schultz and Reitmann, 2019b] usaram modelos de rede neural convolucional (CNN) e rede neural recorrente (RNN) para apoiar o suporte operacional de decisões dos aeroportos. Estudos recentes incluem Schultz and Reitmann [2019b], que utilizou uma abordagem de aprendizado de máquina para prever o embarque de aeronaves, implementando e treinando uma rede neural de memória longa (LSTM) recorrente. Em seu modelo, os autores forneceram dados detalhados sobre a movimentação de passageiros e processos operacionais (espera no corredor, armazenamento de bagagem de mão ou assentos usados, por exemplo). Além disso, o aprendizado de máquina também inclui técnicas como máquina vetorial de suporte, tomada de decisão difusa, máquina de aprendizagem extrema e aprendizado de reforço [Shalev-Shwartz and Ben-David, 2013; Dong et al., 2020].

Um bom exemplo dos métodos usados neste grupo, Redes Neurais é inspirado no cérebro humano para realizar uma determinada tarefa ou funções. Rede Neural (ou Rede Neural Artificial) consegue aprender por meio de exemplos. O processo de aprendizagem ocorre por meio de um conjunto de unidades de entrada e saída conectadas em que cada conexão possui um peso associado a ela. Na fase de aprendizagem, a rede aprende ajustando os pesos para prever o rótulo de classe correto das entradas fornecidas. Ele segue o caminho não linear e processa as informações em paralelo em todos os nós [Shalev-Shwartz and Ben-David, 2013].

A Figura II.1.a mostra um exemplo hipotético, com dados simples de seis voos. Para cada um desses seis voos, a tabela contém o Tempo até Partida (TTD - tempo original em minutos marcado para a partida) e o Tempo desde a Chegada (TSA - tempo em minutos desde que a aeronave chegou ao solo). São informações que, ao olharmos os dados de entrada, podemos inferir o resultado, que

não serve para treinar a camada de processamento.

id	time_to_departure	time_since_arriving	status_on_departure
1	20	5	Delayed
2	10	30	Delayed
3	15	15	Delayed
4	20	50	On time
5	80	50	On time
6	30	80	On time

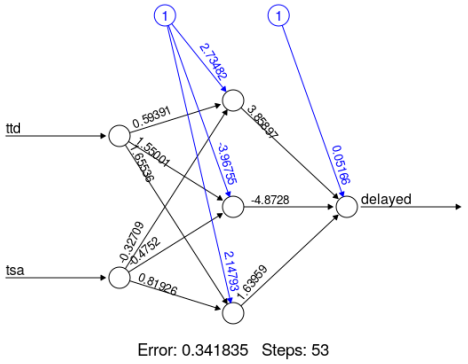


Figura II.1: Dados para treinamento da Rede Neural (a); Exemplo básico usando dados de voo (b)

A Figura II.1.b mostra a rede neural após o treinamento, onde podemos ver os pesos que serão usados durante o processamento. Neste caso já aplicado, a rede neural usou como entrada apenas os dados de TTD e TSA. Na última coluna da tabela está o resultado obtido através do processamento, indicando quais voos terão ou não atrasos, diante do modelo previamente treinado.

Outro método amplamente utilizado que se baseia no treinamento de dados para a formatação de um modelo é chamado Random Forest [Breiman, 2001], e é um algoritmo de aprendizado de máquina que funciona criando várias árvores de decisão e combinando seus resultados. Em vez de aplicar a árvore de decisão como elemento básico, a Random Forest combina a saída de várias árvores de decisão e, finalmente, apresenta sua própria saída. Ele remove o viés que um único modelo de árvore de decisão pode introduzir no sistema. Além disso, melhora significativamente o poder preditivo. Disso segue um ganho na capacidade de predição em relação às árvores de decisão ao trazer a possibilidade de análises combinadas e aleatórias entre as análises de cada árvore.

Segue exemplo, a partir de alguns cenários observados envolvendo as condições climáticas na origem de uma rota de voo, as condições meteorológicas no destino e a indicação se as condições representavam adversas ou aceitáveis para realizar o voo sem atrasos, podemos ver parte deste conjunto de dados na Figura II.2.a. Separamos o conjunto de dados entre dados de treinamento e dados que serão testados no modelo treinado. Por fim, apresentamos na Figura II.2.b um gráfico da árvore com maior número de nós, que representa uma das árvores da floresta.

Podemos ver nas Figuras II.3 e II.4 uma das possíveis representações de um modelo de aprendizado de máquina e sua aplicação. Respectivamente na Figura II.3 a estruturação do modelo de aprendizado de máquina, e na Figura II.4 o uso deste modelo para classificação de dados. Colocamos aqui como uma das representações, pois a ordem ou existência de passos pode variar em cada proposta, como a existência ou não de fase de pré-processamento e separação de dados de treino e testes, por exemplo, embora sejam largamente utilizados, e componentes de boas práticas. Na primeira figura existe a representação de coleta de dados brutos, e em seguida uma fase de

OrigWeather	LocalWeather	condition
goog	goog	acceptable
vgood	vgood	acceptable
bad	acceptable	acceptable
bad	bad	adverse
vbad	vbad	adverse
goog	goog	acceptable
bad	vgood	acceptable
acceptable	acceptable	acceptable
bad	bad	adverse
vbad	vbad	adverse
goog	goog	acceptable

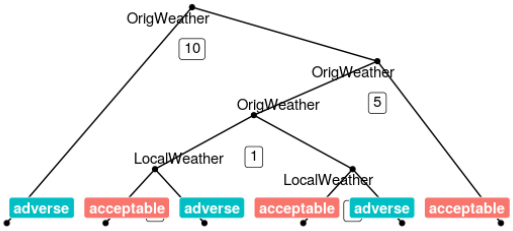


Figura II.2: Dados para treinamento da Random Forest (a); Exemplo básico usando Random Forest (b)

pré-processamento, necessária na maioria dos modelos.

O passo 1.1 de pré-processamento de dados é importante para garantir fatores de qualidade dos dados, como precisão, integridade, consistência e credibilidade. O pré-processamento de dados pode ser dividido em algumas atividades [García et al., 2016; Ramírez-Gallego et al., 2017], como redução de dados, limpeza de dados, transformação de dados e integração de dados. A redução de dados efetua a seleção de recursos, redução da dimensionalidade do espaço ou amostragem. Na seleção de atributos, o objetivo é reduzir o número de atributos, ou seja, aqueles irrelevantes ou redundantes para o problema abordado. A etapa de limpeza remove ruído, dados ausentes que não podem ser corrigidos e valores discrepantes. Além disso, esta etapa também oferece a possibilidade de imputar dados ausentes [Ramírez-Gallego et al., 2017] . A transformação de dados é responsável por transformar e consolidar os dados em um formato que torne o processo de mineração de dados mais eficiente, facilitando o entendimento de padrões nos dados. As técnicas mais comuns incluem normalização, discretização e mapeamento categórico. A integração de dados é empregada quando uma visão única e integrada de dados de mais de um banco de dados (geralmente de diferentes provedores de dados) é necessária.

Após o pré-processamento existe a fase de separação de dados de treino e testes, passo 1.2. Esta separação é vista como boa prática e largamente adotada para melhorar os resultados gerais do modelo. Quando utilizados os mesmos dados para efetuar o treinamento do modelo selecionado, e posteriormente efetuar o teste com os mesmos dados, não se está colocando os métodos de aprendizado à prova já que a afinação dos algorítimos foi feita com esses mesmos dados. A tendência é de que existe uma acurácia, resultado, iguais aos do treinamento. Quando separados os dados em partes de treinamento e testes, aumentam-se as chances de elevar o nível de adequação ao efetuar os testes e refinamento com outro conjunto de dados, que podem ser separados por volume, por período, dentre outras maneiras [Alloghani et al., 2020].

Na etapa de aprendizagem, passos 1.3 e 1.4, existe efetivamente e construção e treinamento do modelo. Neste etapa são montados os algoritmos necessários para efetuar a aprendizagem através dos dados preparados para entrada e gerar os resultados de predição. Este processo deve se repetir diversas vezes efetuando-se validação entre os resultados obtidos e resultados esperados para dados de treinamento, avaliação e melhora da desempenho e sua otimização. Finalmente, segue-se para o teste do modelo com os dados separados para teste. Ainda existem novas fases de validação e otimização [Sathya and Abraham, 2013].

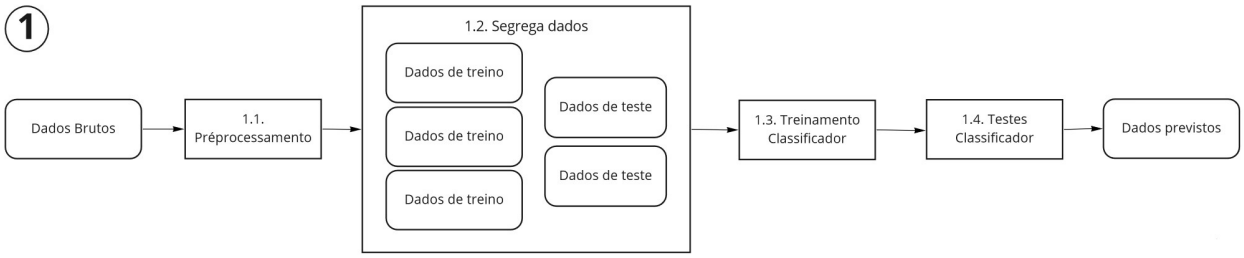


Figura II.3: Visão de processo de treinamento e teste de aprendizado de máquina.

A partir deste momento, temos um modelo pronto para processar dados de um sistema para o qual foi projetado e que podemos observar na Figura II.4. O passo 2.1 é de pré-processamento de fluxos de dados entrantes e o passo 2.2 o do processamento do modelo, classificação. Podemos fazer uma analogia com parte da visão de construção do modelo, porém aqui com processamento de dados fora dos conjuntos iniciais de treinamento e testes, em uma visão de processamento e classificação de dados reais [Sathya and Abraham, 2013].

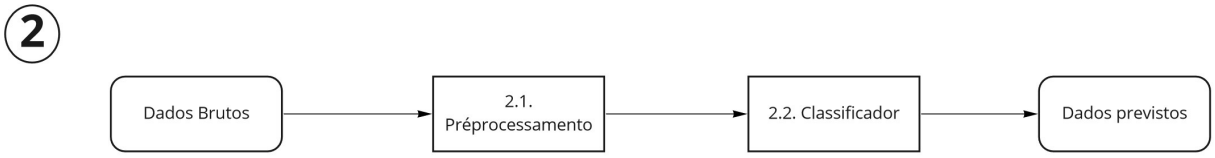


Figura II.4: Visão de aplicação de um modelo de Aprendizado de Máquina.

Capítulo III Desvio de Conceito

Os dados podem mudar com o tempo. Clima, cenário econômico, cenário político, mudanças tecnológicas, sazonalidade regional, todos esses fatores são causas de mudanças de dados que podem alterar os cenários em que ocorrem as operações de voo. Normalmente, os modelos de aprendizado de máquina pressupõem um relacionamento estático entre as variáveis de entrada e saída. Um cenário, geralmente de dados históricos, é analisado para estruturar um modelo de previsão do que ocorrerá nos próximos ciclos do mesmo evento. Este modelo representa a relação entre as variáveis de entrada, que traduzem o cenário analisado, e as variáveis de saída, que identificam o resultado dado pelo modelo uma vez treinado com os dados históricos. No entanto, como esses cenários e dados de entrada mudam ao longo do tempo, os modelos preditivos que assumem que as instâncias são distribuídas sem mudanças relevantes podem resultar em desempenho ruim ou degradar ao longo do tempo de maneiras diferentes [Lu et al., 2019]. Não apenas na área de transporte aéreo, mas o mesmo ocorre na maioria das áreas, incluindo disciplinas totalmente digitais, como comércio eletrônico, e-mails de spam, detecção de fraude e detecção de intrusão.

Ao contexto de dados e sua relação com o modelo uma vez treinado chamamos de Conceito. Uma perda de acurácia momentânea, incremental ou total pode ser motivada pela mudança deste Conceito, demandando uma complementação ao modelo de aprendizado de máquina para esta detecção. O termo Desvio de Conceito é usado em aprendizado de máquina para se referir a mudanças de conceito, ou seja, mudanças nos relacionamentos subjacentes entre dados e funções do modelo treinado. Esta complementação, que envolve a monitoração e adaptação do modelo, é algo apontado como necessário nas metodologias de exploração de dados.

Um dos principais desafios ao criar modelos de aprendizado de máquina é lidar com o desvio de conceito. Refere-se a uma mudança na distribuição dos dados que interfere na relação entre a classe de saída Y e as variáveis de entrada X com significância estatística. Formalmente, um Conceito para uma amostra i é definido como a probabilidade da distribuição conjunta χ de X e Y. Isso é descrito na Equação III.1. Um desvio de conceito entre as amostras i e j é definida como uma diferença entre as probabilidades $p_i(\chi)$ e $p_j(\chi)$ com significância estatística. Ele é descrito na Equação III.2 [Webb et al., 2016].

$$concept_i = p_i(\chi) = p_i(X, Y) \tag{III.1}$$

$$p_i(\chi) \neq p_j(\chi) \tag{III.2}$$

Toda análise e estruturação de um método começam com a decisão de quais partes do cenário de informações são relevantes, por exemplo, quais variáveis serão observadas. Porém, além das variáveis observadas pelo modelo, existem variáveis que não foram analisadas por não parecerem estar relacionadas ao problema ou afetar o resultado. Essas variáveis não fizeram parte do modelo em uma primeira seleção. Porém, com o tempo, ou com alguma mudança de contexto, essas chamadas variáveis ocultas podem fazer a diferença [Webb et al., 2016].

Por exemplo, antes das medidas de restrição social ocasionadas pela pandemia da COVID-19, variáveis como o tempo de quarentena para tripulantes e pilotos não existiam ou não eram consideradas. No entanto, havia uma variável de afastamento em caso de doença, que, sendo desprezível, em geral, estava fora do modelo porque não trouxe nenhuma mudança. Com o Desvio de Conceito, a mudança total de cenário tornou-se uma parte essencial do modelo e afeta diretamente qualquer análise.

III.1 Tipos

Conforme Gama et al. [2014], podemos tipificar os desvios, como descrito abaixo e mostra a Figura III.1:

- Abrupto - identificado logo após sua ocorrência, onde um conceito A é instantaneamente substituído por um conceito B.
- Gradual - que mistura os conceitos A e B, tornando-os ativos por um determinado período, mas, ao longo do tempo, o número de ocorrências do conceito A diminui à medida que o conceito B aumenta.
- Incremental - onde as mudanças são identificadas lentamente. Nesse tipo de desvio, as diferenças acontecem gradualmente, sendo percebidas apenas em longos períodos.
- Recorrente - a recorrência pode ser caracterizada como variações de estados com certo grau de regularidade, como os ciclos das estações, ou datas específicas como o Dia dos Pais, Dia das Mães e Natal. Os modelos aprendidos nestes períodos podem ser armazenados e reutilizados em situações futuras [Lu et al., 2019].

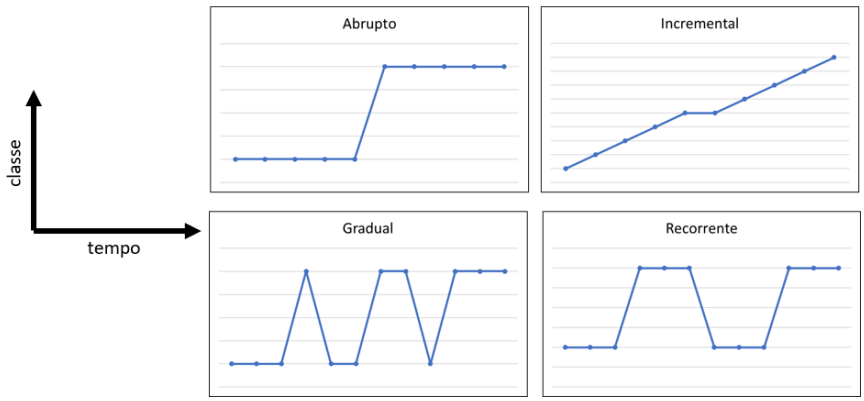


Figura III.1: Tipos de Desvio. Adaptado de Gama et al. [2014].

III.2 Drift real e virtual

Os desvios também podem ser classificados como desvios reais ou virtuais [Iwashita and Papa, 2019]. Especificamente, os reais são definidos por mudanças nos limites de classe, ou probabilidades posteriores (*i.e.*, $P(Y|X)$) independentemente das mudanças nos limites de classe. Pode ser devido a mudanças na variável de saída ($p(Y)$) ou com sua relação com as variáveis de entrada $p(X)$. Por outro lado, desvios de conceito virtuais acontecem sempre que a probabilidade condicional ($p(X|Y)$) muda, o que pode ser devido a mudanças nas variáveis de entrada ($p(X)$) [Iwashita and Papa, 2019; Lu et al., 2019; Hoens et al., 2012]. A mudança de classe pode ocorrer sem uma mudança no conceito. Um exemplo visual é mostrado na Figura III.2.

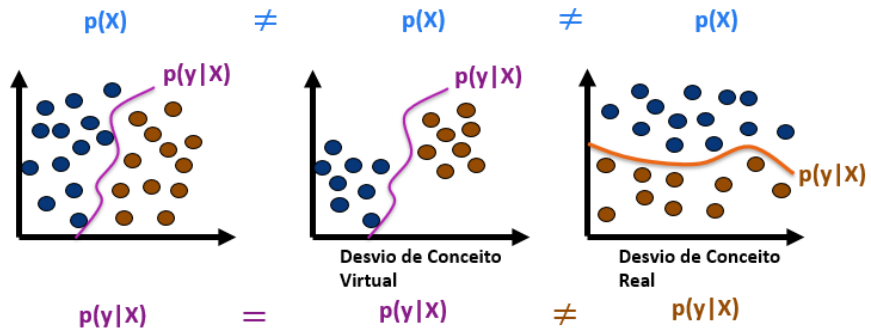


Figura III.2: Desvio de Conceito real e virtual. Adaptado de Gama et al. [2014].

III.3 Detecção de desvio de conceito

Em relação aos atrasos de voos, pode haver alterações relevantes na proporção de voos atrasados. Alguns períodos são mais críticos do que outros [Gui et al., 2020; Sternberg et al., 2016]. Por esse motivo, detectar e lidar com desvios é um assunto relevante. Ele permite uma ação específica para

evitar o aumento de erros em sistemas de aprendizagem online após o desvio ser observado [Iwashita and Papa, 2019; Lu et al., 2019; Webb et al., 2016].

Existem duas categorias principais de detecção de desvio de conceito [Lu et al., 2019]: detecção por distribuição de dados e detecção por taxa de erro. A detecção por distribuição de dados monitora os dados de entrada verificando a distância entre as distribuições de teste e novos contextos. Se a distância dos dados de entrada e os dados usados para os testes do modelo for relevante, isso pode indicar que ocorreu um desvio. A detecção por taxa de erro verifica a camada de saída analisando a precisão dos resultados, onde compara os resultados de previsão com os resultados reais para verificar a precisão do modelo. Grandes diferenças nas taxas de erros também podem indicar que houve desvio.

A Figura III.3 mostra a visão de um modelo de predição de dados usando aprendizado de máquina, porém, aqui complementado com a parte de detecção de desvio de conceito e eventual retreinamento do modelo em sua ocorrência. Existem também indicações dos pontos em que poderíamos efetuar a detecção ativa de desvio.

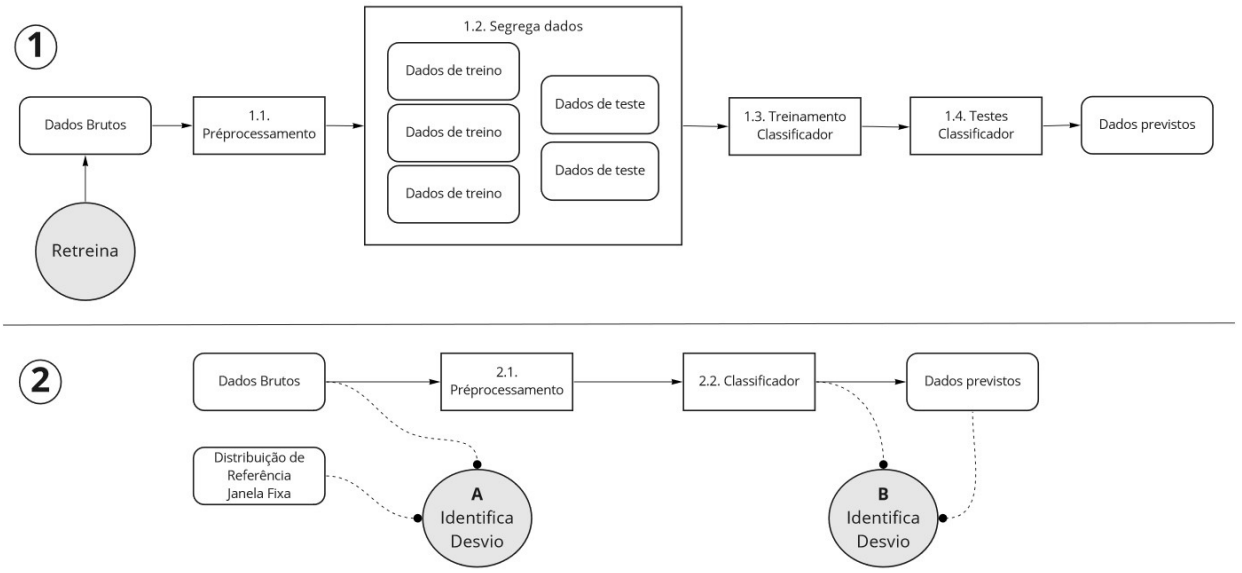


Figura III.3: Visão de um modelo de Aprendizado de Máquina e opções de detecção ativa de desvio de conceito. (A) Detecção pela análise das distribuições, (B) Detecção pela análise de indicadores de desempenho.

Na Figura III.3 existem duas marcas identificadas, A e B, feitas a partir de uma classificação adaptada das taxonomias de referência de Gama et al. [2014] e Pesaranghader and Viktor [2016]. Na marca *A* identificada na figura apresenta o ponto em que uma detecção de desvio de conceito é feita através da identificação da diferença entre as distribuições. Esta forma de detecção baseia-se na suposição de que se as distribuições apresentam diferenças além de um limite, existe desvio de conceito. Embora na figura ocorra antes da fase de pré-processamento, entendemos aqui que esta

fase pode ou não ter ocorrido. A marca B, existe a identificação de necessidade de mudança e retreinamento do modelo a partir da monitoração da evolução de algum indicador de desempenho, como a taxa de erro, para aplicar heurísticas e encontrar pontos de mudança.

Grande parte das avaliações são feitas com janelas deslizantes, conforme mostrado na Figura III.4, em que diferentes alocações de tempo t são consideradas para as avaliações e comparações. Podemos entender essas janelas como conjuntos de dados ou um conjunto de dados de fluxos contínuos particionado em lotes (intervalos de tempo do mesmo tamanho).

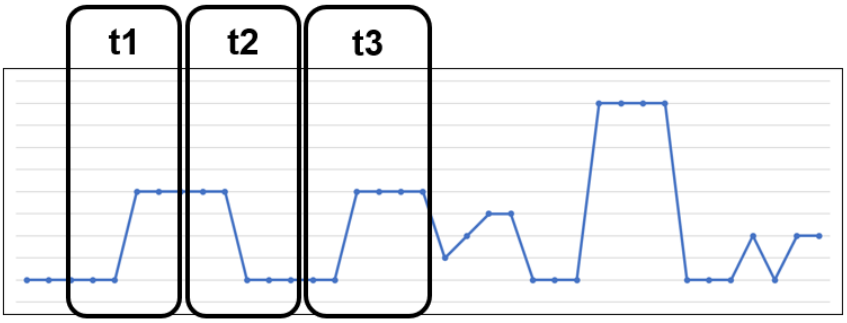


Figura III.4: Janelas deslizantes.

Para exemplificar a forma de detecção por análise de distribuição, segue modelo que identifica desvios de conceito analisando a distância entre janelas de períodos diferentes. Ele utiliza informações históricas do aeroporto de Nova York, de voos ocorridos em 2013. O objetivo da técnica é verificar a distância entre as distribuições de variáveis observadas em dois períodos utilizando a covariância. Atribui-se um maior valor de distância conforme a correlação apurada entre as variáveis de cada período.

No caso abaixo, consideram-se janelas de três meses. Os resultados abaixo foram obtidos mediante códigos escritos em R e executados no Google Colab [Bisong, 2019]. Fora instalado pacote que fornece informações sobre os voos da cidade de Nova York no ano de 2013. Esses dados foram usados para demonstrar a detecção de desvios. A figura III.5 mostra as primeiras linhas da base de origem com dados de voo.

```
> head(flights)
# A tibble: 6 x 19
   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
  <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
1  2013     1     1      517            515         2      830            819
2  2013     1     1      533            529         4      850            830
3  2013     1     1      542            540         2      923            850
4  2013     1     1      544            545        -1     1004           1022
5  2013     1     1      554            600        -6      812            837
6  2013     1     1      554            558        -4      740            728
```

Figura III.5: Primeiras linhas da base de origem com dados de voo.

Neste exemplo, o desvio de conceito é identificado através da covariância, definida como a distância de não interseção entre duas distribuições. Quanto maior a distância, mais diferentes

são as duas distribuições. Dois períodos de três meses foram carregados separadamente para comparação. Todas as variáveis são comparadas neste exemplo.

A Figura III.6 lista cada uma das variáveis nas bases de dados para os períodos comparados. Para cada uma das variáveis, existe um valor de distância associado. Quanto maior o valor, mais significativa é a diferença entre os períodos comparados. Vemos que existe uma diferença absoluta entre as variáveis do mês e do tempo quase total. Isso faz todo o sentido, pois os períodos são diferentes. Pelas informações relacionadas aos voos, notamos que existem variáveis com distâncias muito maiores que as demais, notadamente a identificação da aeronave *tailnum* e o destino *dest*, as identificações e destinos das aeronaves. Ou seja, entre os períodos analisados, existe uma diferença considerável nas aeronaves utilizadas e nos destinos cadastrados. Isso corrobora a sugestão de mudança de contexto, embora não haja registro de mudanças significativas nas taxas de atraso, seja na chegada ou na saída. Nesse caso, por exemplo, pode ser efeito de alguma sazonalidade em que foram incluídas rotas e novos destinos, com a utilização de aeronaves maiores. Isso poderia ter alterado o comportamento operacional tanto neste aeroporto quanto em outros que fazem parte da rede.

Variable	Shift	
year	0.0	
month	100.0	***
day	2.8	
dep_time	2.9	
sched_dep_time	2.8	
dep_delay	3.7	
arr_time	3.3	
sched_arr_time	3.8	
arr_delay	5.1	
carrier	1.8	
flight	4.9	
tailnum	**44.0**	***
origin	0.9	
dest	**32.1**	**
air_time	10.4	.
distance	4.8	
hour	2.2	
minute	5.6	
time_hour	97.3	***

Figura III.6: Lista cada uma das variáveis e valores de desvio.

III.4 Modos de tratamento

Podemos dividir os modos de tratamento primariamente em dois, passivos e ativos. No modo passivo, os métodos são adaptados em intervalos regulares, independentemente de as mudanças terem ocorrido. Os exemplos incluem métodos que ponderam os exemplos de acordo com sua idade

e métodos que usam janelas de tempo de tamanho fixo.

Outro exemplo de uma forma passiva é o de modelos combinados, consistindo na existência prévia de mais de um modelo, gerando um modelo combinado. Desta forma, o resultado dos modelos são combinamos de forma que a possível perda de desempenho de um dos modelos seja mitigada pela combinação.

Em uma estratégia passiva, presume-se que o desvio de conceito ocorre o tempo todo. Os modelos são constantemente atualizados. A desvantagem dessa abordagem é que ela pode retreinar os modelos, mesmo se nenhum desvio ocorreu no conjunto de dados [Iwashita and Papa, 2019; Gama et al., 2014].

No modo ativo, métodos apenas alteram o modelo de decisão após a detecção de um desvio. Eles são usados em conjunto com um modelo de detecção. Sempre que um novo lote é introduzido, a detecção de desvio é aplicada. Se nenhum desvio for detectado, o modelo treinado anteriormente ainda será usado. No entanto, se o desvio for detectado, um novo modelo é construído usando novas distribuições de dados. Por outro lado, o retreinamento contínuo pode ocorrer entre cada par de lotes, assemelhando-se à estratégia passiva. Novamente, tal decisão é baseada em se o desvio é observado sempre que um novo lote é introduzido [Iwashita and Papa, 2019; Gama et al., 2014].

A Figura III.7 parte da representação de diferentes amostras de dados em janelas de tempo e como uma forma de tratamento poderia ser aplicada, por exemplo, executando diferentes modelos de predição em cada um em diferentes janelas de tempo.

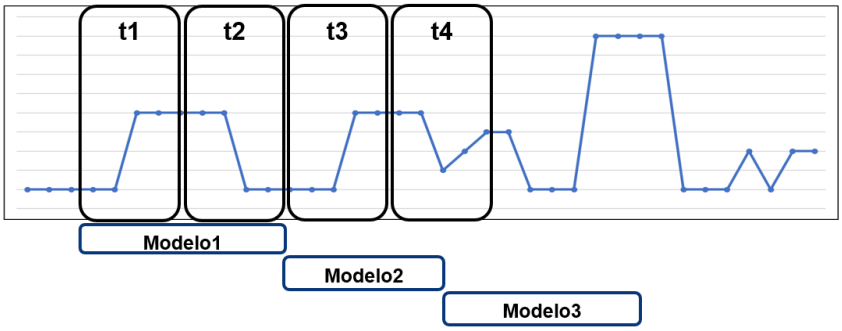


Figura III.7: Representação de diferentes modelos sendo adotados para diferentes janelas de tempo.

III.5 Classificação dos métodos

Os métodos elencados neste seção foram resultados de extensa pesquisa sobre o tema Desvio de Conceito, onde verificamos que grande parte dos trabalhos científicos referir-se a certo grupos de métodos de detecção, seja para comparações, extensão ou detalhamento, assim como referenciados em parte ou em todo como estado da arte para detecção de desvio, o que pode ser verificado no Capítulo IV. Neste trabalho consideraremos como estado da arte os métodos nominalmente citados

a seguir, que serão discutidos nesta seção, e que aparecem na Figura III.8.

A taxonomia utilizada foi adaptada de Gama et al. [2014] e Pesaranghader and Viktor [2016], onde foram categorizados detectores de desvio em três grupos gerais, quais sejam: (i) Análise sequencial; (ii) Abordagens baseadas em controle estatístico de processos; e (iii) Métodos baseados em janelas. Neste trabalho, adotaremos uma classificação adaptada, dividindo primeiramente os métodos entre aqueles que usam Análise de Distribuições e métodos que usam Análise de Desempenho.

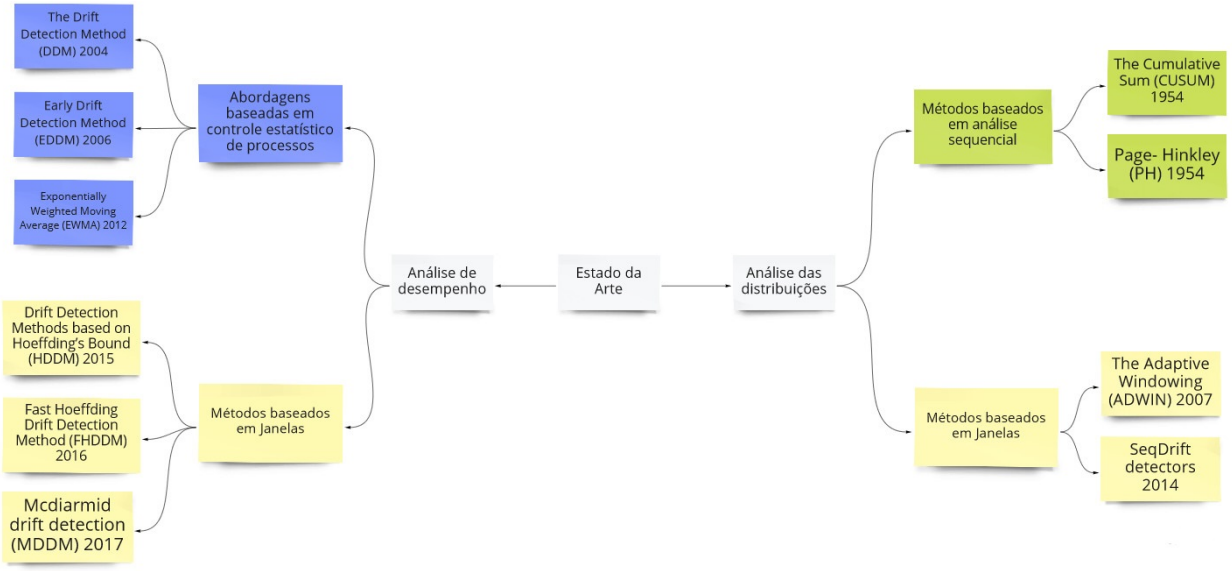


Figura III.8: Estado da Arte em Desvio de Conceito.

III.5.1 Análise de distribuições

Métodos que utilizam análise de distribuições neste trabalho serão os métodos que detectam o desvio de conceito a partir da comparação de duas instâncias de dados. Identificam o desvio por meio de uma medida de limite calculada ou indicada como um dos parâmetros. Nesta seção, seguem modelos com esta classificação.

A maioria das abordagens de detecção de mudanças em distribuições faz isso via média ou mediana, utilizando limites de que consideram a variância e desvio padrão, conforme fora mostrado pelos métodos disseminados descritos anteriormente. Dessa forma, o problema de detecção de desvio é reduzido a estimar mudanças significativas na estatística calculada a partir da sequência de valores que medem uma característica de desempenho.

Análise sequencial

Os métodos de detecção de desvio identificados por distribuições classificados como análise sequencial avaliam variar características dos dados. Armazena-se a maior média alcançada, e

quando o valor acumulado da diferença do valor corrente e a média é maior que determinado limite, ocorreu um desvio. O Cumulative Sum (CUSUM) e sua variante Page-Hinkley (PH) [Page, 1954] são membros deste grupo.

Tanto no CUSUM como em sua variante PH, existe um número de qualidade ω, que representa um parâmetro da distribuição de probabilidade; por exemplo, a média. O CUSUM foi concebido como um método para determinar mudanças nele e propôs um critério para decidir quando tomar uma ação corretiva. Quando o método CUSUM é aplicado às mudanças na média, ele pode ser usado para a detecção de etapas de uma série temporal.

Como o nome indica, CUSUM envolve o cálculo de uma soma cumulativa (sendo o que o torna sequencial). As amostras de um processo S_n são atribuídos pesos ω_n, e somados conforme descrito na Equação III.3, onde x_n corresponde ao valor atual observado, e ω_n corresponde à magnitude das mudanças que são autorizadas.

$$\begin{aligned} S_0 &= 0 \\ S_{n+1} &= \max\left(0, S_n + x_n - \omega_n\right) \end{aligned} \tag{III.3}$$

Quando o valor S ultrapassa um certo valor definido (threshold), identifica-se que houve uma mudança no valor, um desvio de conceito. A fórmula acima detecta mudanças apenas no valor positivo. Pode ocorrer um delimitador de valor negativo, quando é necessário definir um limite mínimo. Neste caso, quando o valor ultrapassa negativamente este valor, assumi-se que o threshold também foi ultrapassado.

O teste Page-Hinkley (PH) é uma variante do CUSUM que foi desenvolvido para verificar mudanças na média de um sinal gaussiano, que seria uma média zerada em uma distribuição normal [Diebold, 2006]. Ele considera uma variável cumulativa U_T, que recebe a diferença acumulada entre os valores X_i e sua média X_T. O cálculo do teste Page-Hinkley (PH) segue na Equação III.4, onde $\bar{X}_T = 1/t \sum_{i=1}^{t} X_i$ e δ corresponde à magnitude das alterações que são permitidas.

Quando essa diferença é maior que o limite δ, sinaliza-se um desvio. O limite δ depende da taxa admissível de falsos alarmes. Aumentar δ implicará em menos alarmes falsos, mas pode perder ou atrasar a detecção de alterações.

$$U_T = \sum_{i=1}^{T} \left(X_i - \bar{X}_T - \delta\right) \tag{III.4}$$

Métodos baseados em janelas

Métodos baseados em janelas usam automaticamente uma janela de referência fixa resumindo a informação passada e uma janela deslizante resumindo as informações mais recentes. Uma diferença significativa entre as distribuições dessas duas janelas sugere a ocorrência de um desvio. Para esta

verificação, testes estatísticos ou desigualdades matemáticas são aplicadas, considerando a hipótese nula de que as distribuições são iguais. O Adaptive Sliding Window (ADWIN) [Bifet and Gavaldà, 2007] e detectores Sequential Drift (SeqDrift) [Pears et al., 2014] são os modelos classificados como baseados em janelas, na classificação de análise de distribuições.

O ADWIN utiliza janelas deslizantes de tamanho variável, recalculadas online conforme a taxa de mudança observada a partir do dados nessas janelas. A janela (W) é ampliada dinamicamente quando não existe mudança clara no contexto e diminui quando uma mudança é detectada. Caso uma mudança seja identificada, elimina uma parte obsoleta de um janela.

O ADWIN se baseia no uso do limite de Hoeffding [Bentkus, 2004] para detectar desvios. A versão inicial do ADWIN mantinha uma janela (W) de instâncias em um determinado momento e compara a diferença média de quaisquer duas subjanelas ($W0$ de instâncias mais antigas e $W1$ de instâncias recentes) de W. Se a diferença média for estatisticamente significativa, então ADWIN0 remove todas as instâncias de $W0$ consideradas para representar o conceito antigo e apenas transporta $W1$ para o próximo teste. No entanto, como mencionado em [Bifet and Gavaldà, 2009], ADWIN0 sofre do uso de limite de Hoeffding, que superestima a probabilidade de grandes desvios para distribuições de pequena variação. Como tal, limite de Bernstein, muito mais restrito, foi usado em um método de acompanhamento, intitulado ADWIN [Bifet and Gavaldà, 2007]. O ADWIN utiliza taxas de falso positivos e falsos negativos para controlar desempenho. Uma janela deve ser mantida para cada dimensão de dado, ao conseguir manipular apenas dados unidimensionais.

Abaixo segue na Equação III.5 para a desigualdade de Hoeffding [Bentkus, 2004], onde $X_i, ..., X_n$ são variáveis aleatórias independentes, $E[X]$ é o valor esperado da média da população, n é o número de amostras que temos, também definido como tamanho da amostra. O valor μ é o valor esperado de nossas variáveis aleatórias, v é o valor observado de nossas variáveis aleatórias e ϵ é épsilon, que define o desvio de v e μ.

Ela fornece um limite superior para a probabilidade de que a soma das variáveis desvie de seu valor além de um certo limite pré-definido.

$$\Pr\left(\left|\frac{1}{n}\sum_{i=1}^{n} X_i - E[X]\right| > \epsilon\right) \leq 2\exp\left(-2n\epsilon^2\right). \tag{III.5}$$

A desigualdade de Bernstein, expressa na Equação III.6, é muito similar, mas considera a variância, onde $X_i \in [a, c] \forall i$, e σ^2 é a variância populacional.

$$\Pr\left(\left|\frac{1}{n}\sum_{i=1}^{n} X_i - E[X]\right| > \epsilon\right) \leq 2\exp\left(\frac{-n\epsilon^2}{2\sigma^2 + \frac{2}{3}\epsilon(c-a)}\right) \tag{III.6}$$

O SeqDrift2 usa o mesmo mecanismo de janelas do ADWIN, utilizando em sua forma de identificação de mudanças entre distribuições o limite de Bernstein [Pears et al., 2014], que apresenta

limites para a probabilidade de que a soma das variáveis aleatórias se desvie de sua média. Seq-Drift2 armazena entradas em dois repositórios chamados de esquerda e direita. À medida que as entradas são processadas, o repositório esquerdo contém uma combinação de entradas mais antigas e novas, aplicando a estratégia de amostragem. O repositório direito coleta as novas entradas que chegam. Posteriormente, encontra-se um limite superior para a diferença entre as médias de falsos negativos e positivos dos dois repositórios, ou seja, para o repositório esquerdo e para o repositório direito, usando a desigualdade de Bernstein. Finalmente, uma diferença significativa entre as duas médias sugere um desvio. Basicamente o Seqdrift2 é uma melhoria do ADWIN, com menor taxa de falsos positivos e aumento na velocidade de detecção de desvios.

III.5.2 Análise de Desempenho

Os métodos que seguem abaixo efetuam a identificação de desvio a partir de comparação de resultados do classificador com a classificação de dados reais. Uma mudança significativa nessa medida de desempenho identifica um desvio. O foco passa dos dados da distribuição para as distribuições de resultados de desempenhos aferidos ao longo das análises.

A abordagem de avaliação para detecção de desvio em relação à sua correção funciona a partir da identificação das taxas de verdadeiro positivo (VP), verdadeiro negativo (VN), falso positivo (FP) e falso negativo (FN), que indicam como uma detecção de desvio opera contra um fluxo de dados em evolução. Onde VP e VN representam as instâncias positivas classificadas corretamente e as instâncias negativas classificadas corretamente, respectivamente. FP e FN representam as instâncias negativas classificadas incorretamente como positivas e as instâncias positivas classificadas incorretamente como negativas, respectivamente. Finalmente, P e N representam todas as instâncias positivas e todas as instâncias negativas. Diante dessa nomenclatura, a *precisao* seria a fração entre todas as instâncias classificadas corretamente sobre o total de instâncias conforme expressa a Equação III.7.

$$\text{precisao } = \frac{VP + VN}{P + N} \tag{III.7}$$

A *taxa de erro* pode ser considerada uma medida de classificação alternativa que é a $1 - precisao$, conforme expressa Equação III.8.

$$taxa\ de\ erro = \frac{FP + FN}{P + N} \tag{III.8}$$

As medidas de verdadeiro positivo (VP), verdadeiro negativo (VN), falso positivo (FP) e falso negativo (FN) podem ser usadas para avaliar o desempenho de detectores de desvio. Podemos considerar de forma geral que um detector de desvio com as maiores taxas de verdadeiro positivo e

verdadeiros negativos, assim como menores taxas de falsos positivos e negativos seja preferido para comparações.

Abordagens baseados em controles estatísticos de processos

Abordagens baseadas em controles estatísticos de processos utilizam parâmetros estatísticos, como média e desvio padrão dos resultados de previsão para detectar desvios de conceito em um fluxo. Estão inclusos nesta classificação controles estatísticos que usam medida de desempenho para detecção de desvios. O Drift Detection Method (DDM) [Gama et al., 2004], o Early Drift Detection Method (EDDM) [Baena-Garcia et al., 2006], o Exponentially Weighted Moving Average (EWMA) [Ross et al., 2012] e Reactive Drift Detection Method (RDDM) [Barros et al., 2017], são membros deste grupo.

O DDM usa uma distribuição binomial para descrever o comportamento de uma variável aleatória que fornece o número de erros de classificação em uma amostra de tamanho n. O DDM calcula para cada instância i no fluxo, a probabilidade de classificação incorreta pi e seu desvio padrão si. Se a distribuição das amostras for estacionária, pi diminuirá conforme o tamanho da amostra aumenta. Um processo estacionário é aquele cujas propriedades estatísticas não mudam com o tempo [Mitchell, 1997]. Se a taxa de erro do algoritmo de aprendizagem aumentar significativamente, ele sugere mudanças na distribuição de classes, fazendo com que o modelo atual construído seja inconsistente com os dados atuais e, assim, fornecendo o sinal para atualizar o modelo. O DDM calcula os valores de pi para cada instância, e quando $pi + si$ atinge seu valor mínimo, ele armazena $pmin$ e $smin$. Em seguida, o DDM verifica duas condições para detectar se o sistema está no nível de alerta ou desvio de conceito: o nível de alerta é ativado quando $pi + si \geq pmin + 2smin$. Além deste nível, ele armazena as instâncias antecipando uma possível mudança de contexto. O nível de desvio de conceito é acionado quando $pi + si \geq pmin + 3smin$. Nesse nível, o DDM zera as variáveis ($pmin$ e $smin$) e o modelo induzido. Posteriormente, um novo modelo é aprendido usando as instâncias armazenadas desde que o nível de aviso foi acionado. O DDM mostra bom desempenho para detectar mudanças graduais e mudanças abruptas, mas tem dificuldade em detectar desvios quando a mudança é lentamente gradual. Em uma mudança lentamente gradual é possível existir acúmulo de amostras armazenadas ao longo do tempo, antes que o nível de desvio seja ativado, e haja o risco de transbordar o espaço de armazenamento da amostra [Baena-Garcia et al., 2006].

O EDDM é uma proposição de modificação do DDM feita por Baena-Garcia et al. [2006]. O mesmo mecanismo de alerta foi usado, mas em vez de usar a taxa de erro do classificador, foi proposta a distância entre erros na classificação. Eles denotam p'_i como a distância média entre dois erros consecutivos e s'_i como seu desvio padrão. Usando esses valores, as novas condições

de aviso e alarme são fornecidas comp $(p'_i + 2 \cdot s'_i) / (p'_{\max} + 2 \cdot s'_{\max}) < \alpha$ para o nível de aviso, e $(p'_i + 2 \cdot s'_i) / (p'_{\max} + 2 \cdot s'_{\max}) < \beta$ para o nível de alarme de desvio. Onde α e β são valores de níveis escolhidos para alarmes de aviso e desvio, sendo que quanto mais perto de 1, mais perto do limite o alarme será acionado. Em seu trabalho [Baena-Garcia et al., 2006] utilizam $\alpha = 0.95$ e $\beta = 0.90$. Os valores de p'_i e s'_i são armazenados quando atinge seu valor máximo $(p'_i + 2 \cdot s'_i)$, que representa 95% da distribuição, obtendo $p'_{\max}$ e $s'_{\max}$. EDDM funciona melhor do que DDM para desvio gradual lento, mas é mais sensível ao ruído. Outra desvantagem é que ele considera os limites e procura por desvios de conceito quando um mínimo de 30 erros ocorrem.

O RDDM é um derivação do DDM chamada de reativa, que desenvolve o anterior eliminando distribuições muito longas. Ou seja, independente da detecção de um desvio, após tempo determinado, descartam-se instâncias mais antigas de distribuições muito longas, visando detectar desvios mais cedo. A ideia principal por trás do RDDM é encurtar o número de instâncias de distribuições longas e estáveis, a fim de lidar com um problema conhecido de perda de desempenho do DDM. Isso não causa nenhuma modificação na forma de cálculo, mas aciona o recálculo das estatísticas usadas para detectar os níveis de aviso e desvio usando um número menor de instâncias.

O EWMA em tradução do próprio nome é a média móvel exponencialmente ponderada. É uma estatística para monitorar o processo que calcula a média dos dados de uma forma que atribui cada vez menos peso aos dados, sendo posteriormente removidos no tempo. Existe um parâmetro λ que determina a taxa na qual dados mais antigo entram na estatística. Por exemplo, um valor de $\lambda = 1$ indica que apenas os dados mais recentes entram no cálculo. Quanto mais o valor de λ estiver perto de 1, mais peso aos valores mais recentes, e quanto mais próximo a 0, maior peso aos valores mais antigos. A fórmula de cálculo do EWMA segue na Equação III.9, onde $EWMA_0$ é a média de dados históricos, Y_t é a observação no momento t, n é número de observações a ser considerado, e $0 < \lambda <= 1$ é a constante que determina a quantidade de memória do EWMA. A variância pode ser calculada através da Equação III.10, onde s é o desvio padrão calculado com dados históricos.

$$\text{EWMA}_t = \lambda Y_t + (1 - \lambda)\text{EWMA}_{t-1} \text{ for } t = 1, 2, \ldots, n \tag{III.9}$$

$$s^2_{\text{ewma}} = \frac{\lambda}{2 - \lambda} s^2 \tag{III.10}$$

Dessa forma, a linha central de controle do EWMA seria traçada como $EWMA_0$ e limites controles superiores e inferiores como $UCL = \text{EWMA}_0 + ks_{\text{ewma}}$ e $LCL = \text{EWMA}_0 + ks_{\text{ewma}}$, onde k pode variar entre ser um valor fixo 3, ou usando as tabelas encontradas em [Ross et al., 2012]. Os limites UCL e LCL são usados para controlar valores considerados em uma normalidade na distribuição. Os que ficam fora desta distribuição representam uma anormalidade e, levando isso

para dados resultantes de modelos de predição, esses controles podem ser usados para detecção de desvio quando uma existe um determinado número de ocorrências que ultrapassam esses limites.

Métodos baseados em janelas

Nesta seção seguem os métodos baseados em janelas, como na Subseção III.5.1, mas que aqui utilizam medidas de desempenho para detectar o desvio em lugar da diferença entre distribuições. São membros desse grupo os métodos de detecção de desvio baseados na desigualdade de Hoeffding, Drift Detection Methods based on Hoeffding's Bound (HDDM) [Frías-Blanco et al., 2015], Fast Hoeffding Drift Detection Method (FHDDM) [Pesaranghader and Viktor, 2016] e McDiarmid Drift Detection Method (MDDM) [Pesaranghader et al., 2018].

A família HDDM é formada por métodos de detecção de desvio de conceito baseados em limites de Hoeffding [Bentkus, 2004]. Sua mecânica funciona através da monitoração de uma média estimada de desempenho, a partir de erros e acertos de um classificador, e então detectar mudanças relevantes. Mantêm-se a mecânica de janelas e a cada período existe a estimativa de erros aferida pelo modelo de predição. O HDDM utiliza limites de Hoeffding para verificar a distância ou desigualdade das distribuições de informações sobre erros nas predições. Ou seja, diferença entre número falsos positivos ou falos negativos em tempo t_0 em relação ao número de falsos a positivos e falsos negativos em um resultado aferidos em t_1 indicam o desvio. Enquanto isso, compara a média dos elementos na janela com a média máxima observada até o momento; subsequentemente, uma diferença significativa entre as duas médias, limitada pela desigualdade de Hoeffding, indica a ocorrência de desvio.

O MDDM mantêm conceito muito parecido com o HDDM, porém, utiliza a desigualdade de McDiarmid [McDiarmid, 1989], a fim de detectar o desvio. O MDDM prossegue deslizando uma janela sobre os resultados da previsão, e associa entradas de janela a pesos. O MDDM ao ponderar os resultados de previsão associados a uma janela deslizante, coloca mais ênfase nos elementos mais recentes. Ele insere 1 na janela se o resultado da previsão estiver correto; e 0 de outra forma. Cada elemento na janela está associado a um peso. Uma diferença significativa entre as duas médias ponderadas, limitada pela desigualdade de McDiarmid, implica um desvio. Em outras palavras, o MDDM limita a diferença entre a média ponderada dos elementos na janela deslizante e a média ponderada máxima até o momento, usando a desigualdade de McDiarmid. Eventualmente, ele alarma para desvio do conceito, uma vez que uma diferença significativa é encontrada. Em um contexto de fluxo contínuo de dados, presume-se que μ_w^t é a média ponderada de uma sequência de entradas não aleatórias, no período, e μ_w^m é a média ponderada máxima observada até o momento. Cada entrada p_i está em $0, 1$ e tem um peso de w_i. Seja $\Delta\mu_w = \mu_w^m - \mu_w^t \geq 0$ a diferença entre essas duas médias ponderadas. Dado o nível de confiança δ_w, a desigualdade McDiarmid detecta

um desvio se $\Delta\mu_w \geq \varepsilon_w$. Segue Equação III.11 de definição de ε_w, Equação III.12 que define v_i.

$$\varepsilon_w = \sqrt{\frac{\sum_{i=1}^{n} v_i^2}{2} \ln \frac{1}{\delta_w}} \tag{III.11}$$

$$v_i = \frac{w_i}{\sum_{i=1}^{n} w_i} \tag{III.12}$$

O FHDDM também utiliza janelas deslizantes e limite de Hoeffding, neste caso representado por ε_d, só que no FHDDM a janela deslizante ocorre em comparação aos resultados de predição para detectar desvios. O FHDDM armazena os erros e acertos do classificador e irá calcular a probabilidade de um acerto ser observado em uma janela, armazenando e comparando com a maior probabilidade de acerto até o momento. Considerando a probabilidade p de acertos em uma janela de tamanho n em um tempo m, $\left(p_m^1\right)$, sendo a probabilidade máxima a $\left(p_{\max}^1\right)$, onde um desvio seria identificado quando $\Delta p \geq \varepsilon_d$. Abaixo, seguem Equação III.13 de definição de Δp, e a Equação III.14 de definição de ε_d.

$$\Delta p = p_{\max}^1 - p_m^1 \tag{III.13}$$

$$\varepsilon_d = \sqrt{\frac{1}{2n} \ln \frac{1}{\delta}} \tag{III.14}$$

Em [Pesaranghader and Viktor, 2016], no trabalho da definição do método Fast Hoeffding Drift Detection Method (FHDDM), foi introduzida abordagem para contar as medidas mencionadas, definindo a noção de comprimento de atraso aceitável. O comprimento de atraso aceitável pode ser definido como um limite para determinar a que distância um desvio detectado pode estar da verdadeira localização do desvio para ser considerado verdadeiro positivo. Considerando o comprimento de atraso aceitável Δ, explicamos as medidas de verdadeiro positivo, falso positivo e falso negativo da seguinte forma:

Verdadeiro positivo: um desvio que ocorre no momento t é contado como um verdadeiro positivo se foi relatado pelo método no intervalo de detecção aceitável definido como $[t; t + \Delta]$. O número de verdadeiros positivos será contado como o número de desvios detectados nesta faixa.

Falso positivo: a taxa de falsos positivos é calculada como o número geral de desvios relatados fora dos intervalos de tempo aceitáveis.

Falso negativo: Se um desvio ocorreu em um momento t, um falso negativo será contado se o método não relatou um desvio no intervalo de detecção aceitável $[t; t + \Delta]$.

Verdadeiro negativo: são os demais espaços amostrais onde poderiam ter ocorrido desvios ou aferidos desvios pelo método de detecção, mas não ocorreram nenhum dos dois, isso significando

um acerto na não identificação de desvio em momento onde não tenha ocorrido.

O comprimento de atraso aceitável, ou seja, Δ, pode ser ajustado empiricamente ou com base no conhecimento prévio sobre o domínio.

A Figura III.9 demonstra como o verdadeiro positivo, o falso positivo e o falso negativo são contados. A primeira linha exibe os locais dos desvios em um fluxo de dados e a segunda linha exibe os resultados do método de detecção. A última linha mostra como o método de detecção deve ser avaliado em cada ponto do tempo. Os quadrados verdes representam pontos de desvio detectados com sucesso (verdadeiros positivos), os quadrados vermelhos representam os pontos de não deriva que foram confundidos com pontos de deriva pelo método (falsos positivos), enquanto os quadrados amarelos representam os pontos de deriva que não foram detectados pelo método de detecção (falsos negativos).

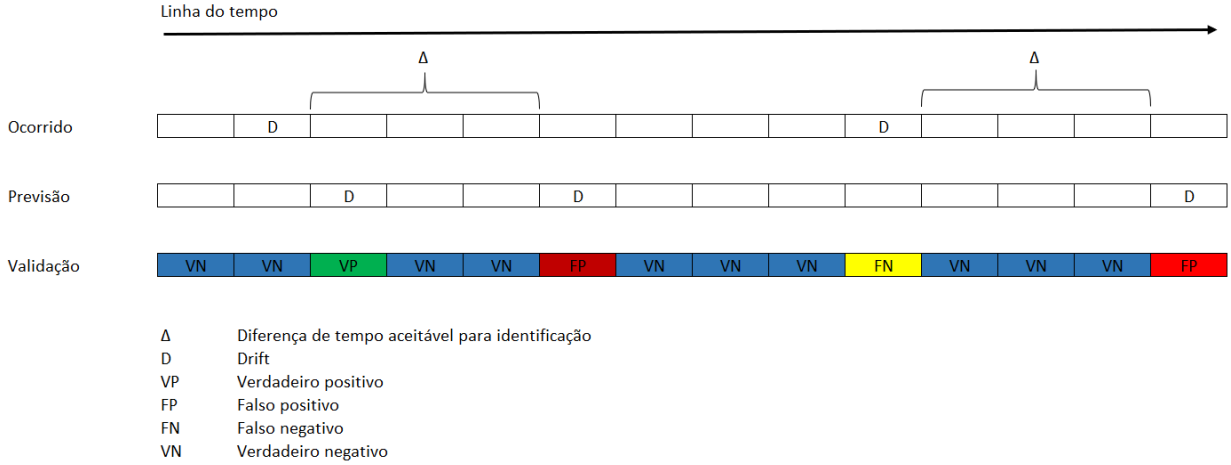


Figura III.9: Modelo de avaliação do FHDDM.

Este tipo de avaliação é uma medida de desempenho chave para métodos de detecção de desvio, para podermos avaliar a responsividade do método na detecção e em que tempo isso ocorre. Quanto mais próximo ao tempo em que o desvio ocorre de fato, tanto melhor para todo o sistema, inclusive para se poder estudar ou tratar as causas. Por exemplo, em um sistema de previsão de atrasos em voos, o quanto mais assertivo em termos de momento em que ocorre essa detecção, mais rápido poderão ser detectadas as causas e mudanças que se façam necessárias. Principalmente se isso ocorre em fluxos de dados contínuos e sistemas de detecção em tempo real. Nestes sistemas, caso a detecção se dê em tempo muito posterior, as causas podem ter cessado, assim como a própria mudança, fazendo com que possíveis mudanças sejam mais difíceis de serem programadas ou nem façam mais sentido. Um atraso de detecção mais curto permite que os algoritmos de aprendizagem usem mais instâncias da nova distribuição para treinar os modelos de classificação. Considerando o modelo de aprendizagem provavelmente aproximadamente correto [Mitchell, 1997], uma maior precisão de classificação pode ser alcançada usando mais instâncias de uma distribuição.

III.5.3 Discussão

CUSUM e Page-Hinkley (PH) são alguns dos métodos pioneiros e detectam desvio calculando a diferença de valores observados da média e alarme quando esse valor é maior do que um limiar definido pelo usuário. Esses algoritmos são sensíveis aos valores dos parâmetros, que no caso de estarem muito baixos resultam em alarmes falsos e se estiverem muito altos podem não detectar mudanças. Os métodos DDM, EDDM e RDDM mantém conjuntos de variáveis por amostragem a fim de monitorar o desvio, o que faz com que o seu uso de memória seja geralmente baixo, assim como o tempo de processamento. Esses métodos controlam a taxa de erros aferida entre a comparação da previsão e dados reais de ocorrência. Enquanto o DDM controla a taxa de erros, o EDDM verifica a distância entre erros na classificação e o RDDM resolve um problema de conceitos muitos extensos do DDM. Diferente da família DDM, os primeiros métodos que aplicam janelas deslizantes, ADWIN e SeqDrift2, mantêm uso de janelas de tempo, que sofrem acúmulo caso não exista identificação de mudanças significativas na distribuição. Isso pode causar maior necessidade de uso de memória e tempo de processamento. ADWIN e SeqDrift2 geralmente requerem mais memória para armazenar resultados de previsão, conforme mantidos em janelas deslizantes ou repositórios. Eles também são computacionalmente mais caros, devido aos procedimentos de compressão de sub-janela ou de amplificação de reservatórios. O SeqDrift2 emprega a desigualdade de Bernstein para detectar desvio. A desigualdade de Bernstein é conservadora e requer um parâmetro de variância, em contraste com, por exemplo, a desigualdade de Hoeffding, utilizada no ADWIN. Essas deficiências podem levar a atrasos de detecção mais longos e uma perda potencial de precisão. O HDDM é o primeiro método que emprega janelas deslizantes com controles de erro de classificação. Ele emprega a desigualdade de Hoeffding para detectar desvio. O FHDDM utiliza janelas deslizantes sobre o fluxo, a fim de detectar desvio. Mantemos duas variáveis, ou seja, a média de elementos na janela no tempo atual e a média máxima observada até agora. O FHDDM posteriormente emprega a desigualdade de Hoeffding para detectar desvio. Esta abordagem difere, portanto, do HDDM, que usa uma janela deslizante e mantém apenas duas variáveis.

A Tabela III.1 mostra algumas características de cada um dos métodos do estado da arte apresentados. Essa lista permite um melhor entendimento da classificação e funcionamento, assim como a comparação entre eles. Primeiro, a coluna de Informação base para detecção apresenta qual informação é utilizada para a detecção do desvio de conceito. Conforme taxonomia apresentada, uma separação básica está entre os métodos que utilizam diferença entre informações que estão sendo utilizadas para classificação ou resultados da classificação. E os métodos que usam medidas de desempenho.

Na coluna Mecanismo, apresentamos a forma de aferição de diferenças, seja entre as distribuições ou entre medidas de desempenho. O mecanismo apresenta o como o método se propõe a quantificar

a medida que será verificada. Essa é uma característica fundamental para a detecção do desvio de conceito que é abordada como divergência ou distância. Através da diferença entre distribuições ou desempenho apura-se , associado ao limite, uma tendência de desvio ou a sua ocorrência.

A coluna Limite apresenta o que o método entende como limite para a identificação ou não de um desvio. Por exemplo, o mecanismo poderia ser a diferença entre a média da distribuição corrente e a média máxima das últimas distribuições. Mas apenas essa informação não identifica se houve ou não o desvio. Essa é a forma de apresentar em números as diferenças. O que determinará se existe um alarme ou desvio de conceito é o limite que cada método propõe. Apuradas as diferenças, caso ultrapasse o limite, existe um desvio.

Tabela III.1: Tabela comparativa dos métodos

Método	**Informação base para detecção**	**Mecanismo**	**Limite**
CUSUM	Distribuição	Diferença entre valor e média	Deve ser definido
Page-Hinkley (PH)	Distribuição	Diferença entre valor e média	Deve ser definido
ADWIN	Distribuição	Diferença entre as médias das distribuições	Limite de Hoeffding e Bernstein
SEQDRIFT	Distribuição	Diferença entre as médias das distribuições	Limite de Bernstein
DDM	Taxa de erros dos resultados	Calcula a probabilidade de classificação incorreta	Limite da probabilidade mínima de erro e três vezes o desvio padrão
EDDM	Taxa de erros dos resultados	Calcula a probabilidade de classificação incorreta	Distância média entre erros. Proporção entre média e desvio padrão corrente e a média máxima e duas vezes o desvio padrão máximo
RDDM	Taxa de erros dos resultados	Calcula a probabilidade de classificação incorreta	Limite da probabilidade mínima de erro e três vezes o desvio padrão
EWMA	Taxa de erros dos resultados	Média móvel da taxa de erros (classificação incorreta)	De uma a três vezes a variância (k)
HDDM	Desempenho (a partir dos acertos e erros do classificador)	Diferença entre média na taxa de desempenho	Limite de Hoeffding
MDDM	Desempenho (a partir dos acertos e erros do classificador)	Diferença entre média na taxa de desempenho	Limite de McDiarmid
FHDDM	Desempenho (apenas predições corretas)	Diferença entre média na taxa de acertos	Limite de Hoeffding

Capítulo IV Trabalhos Relacionados

Este capítulo encontra-se dividido em três partes, quais sejam: (i) Trabalhos relacionados ao desvio de conceito; (ii) Desvio de Conceito na predição de atrasos aéreos; e (iii) Análise. No primeiro, repassamos publicações mais relevantes sobre desvio de conceito. Na segunda seção, existe um corte sobre estudos sobre atrasos em voos envolvendo desvio de conceito, ou seja, a aplicação do desvio de conceito em modelos de predição de atrasos em voos. A terceira seção apresenta uma análise sobre como está a aplicação das formas de detecção e tratamento de desvio de conceito, delimitação do estado da arte e tendências.

IV.1 Trabalhos relacionados ao desvio de conceito

Para esta seção, foi realizada pesquisa para identificar estudos relevantes sobre o desvio de conceito com string de pesquisa usada: ("concept drift") AND ("predict" OR "data science") AND pubyear $>$ 2018. Foram considerados apenas artigos (Revistas e Conferências) inteiramente escritos em inglês. A consulta foi submetida à base de dados Scopus (considerando títulos, palavras-chave e resumos). Esta pesquisa foi realizada em três momentos específicos, quais sejam: em 2019, durante os experimentos prévios que seguem no Apêndice A, publicados em Giusti et al. [2022]; em setembro de 2021, logo antes da qualificação; e, em fevereiro de 2023, para complementar a pesquisa antes da defesa. Importante salientar esses momentos diversos que seguem com a maturidade do trabalho. Em um resultado total, considerando os três momentos e artigos com mais de cinco citações, selecionamos quatorze artigos principais. Além desta seleção, outros treze foram selecionados através de *snowballing*.

Focamos também a pesquisa em desvio de conceito para mineração de dados. Embora a ideia de muitos dos métodos para detecção de desvio de conceito possam ser utilizados em diversas áreas, como alguns que este capítulo abordará, existem áreas de aplicações com definições e uso de métodos específicos, como em Mineração de Processos. Conforme Sato et al. [2022]; Ceravolo et al. [2022], não é possível possível utilizar os mesmos métodos desenvolvidos para mineração de dados em Mineração de Processos. Nessa área, por exemplo, os trabalhos existentes mostram que a Mineração de Processos ainda se concentra principalmente na análise off-line, e a avaliação de técnicas de desvio de conceito em processos é complicada devido à falta de protocolo de avaliação, conjuntos de dados e métricas comuns.

Segundo Lu et al. [2019], a detecção de desvio de conceito baseada em desempenho e na distribuição de dados mantém um papel dominante na pesquisa do assunto. Essa forma de classificação foi exatamente o primeiro nível da taxonomia proposta no Capítulo III, e a forma que separamos as próximas subseções.

IV.1.1 Análise das distribuições

Nesta subseção estão relacionados trabalhos que tratam do assunto desvio de conceito através da análise de diferenças entre distribuições de dados. Os estudos aqui apresentados detectam e ou tratam o desvio de conceito usando análise direta em distribuições. Essa análise pode ser feita usando dados reais, de previsão, ou uma combinação deles.

Em seu trabalho sobre o desempenho dos multissensores de qualidade do ar, De Vito et al. [2020] correlacionaram a dissimilaridade entre distribuições com a degradação de desempenho de monitores inteligentes da qualidade do ar. As medidas de dissimilaridade de Kullback-Leibler, Euclidiana e Hellinger foram correlacionadas com a degradação de desempenho registrada. Os resultados mostram que quantificar as mudanças na distribuição de probabilidade dos fatores relevantes permite explicar e prever desempenhos de modelos de calibração orientados por dados de campo. Diante dos resultados, a detecção de desvio de conceito foi incorporada ao modelo. E os monitores inteligentes da qualidade do ar passaram a funcionar de forma autônoma sem a necessidade de recalibração.

Em modelo de predição de demanda para redes de táxis, Saadallah et al. [2020] já embutiram detecção e tratamento de desvio de conceito. O modelo BRIGHT, uma estrutura de aprendizado supervisionado ciente do desvio de conceito para prever as quantidades de demanda. Para detecção de desvio de conceito o modelo usa o teste Page-Hinkley (PH).

O contexto recorrente é um caso particular de desvio, onde as distribuições de características já vistas no passado reaparecem à medida que o fluxo evolui. Este problema foi estudado por Namitha and Santhosh Kumar [2020] na utilização de agrupamento de fluxo, onde utilizou-se K-means como método de agrupamento. Em seu trabalho propuseram um novo algoritmo para identificar os contextos recorrentes em *clustering* de fluxo de dados. Durante a recorrência do contexto, o modelo mais adequado é recuperado do repositório e reutilizado. O algoritmo tem requisitos mínimos de memória e funciona online com o fluxo. Para detecção de mudança de contexto foi utilizado o detector baseado em análise sequencial Page-Hinkley (PH). O teste de Page-Hinkley monitora um parâmetro que representa mudança. No agrupamento, a distância entre os pontos de dados e seus centros de *cluster* mais próximos é o parâmetro escolhido. A detecção de desvio de conceito explícita ajuda a que o tamanho da janela seja variável para se ajustar ao comprimento do contexto.

Raab et al. [2020] apresentaram novo modelo de detecção de desvio de conceito, o Kswin. O modelo usa o teste de Kolmogorov-Smirnov como um detector de desvio de conceito, que espera

duas janelas de dados. O teste de Kolmogorov-Smirnov é um teste não paramétrico que aceita dados unidimensionais sem quaisquer suposições de distribuições subjacentes. Ele compara a distância absoluta entre duas distribuições empíricas cumulativas. Neste trabalho, o detector Kswin foi comparado aos modelos ADWIN, DDM e EDDM.

Em trabalho com objetivo de melhorar o custo de reprocessamento de modelos com a atualização contínua de máquinas de aprendizagem extrema sequencial online, Yang et al. [2020] propuseram um novo método de detecção de desvio de conceito para aprendizagem incremental em ambientes não estacionários. Objetivo deste trabalho foi propor mudanças que diminuíssem o número de atualizações necessárias no modelo ao longo o tempo. Estas modificações e modelos foram testados em dados e planta eólica de geração de energia. A forma específica de detecção de desvio de conceito foi feita através da medida de dissimilaridade. Esta medida ocorre entre a linha de base e o modelo treinado, usando os dados coletados, e o modelo atualizado, considerando os dados novos coletados. A distância euclidiana foi utilizada para determinar essa dissimilaridade. Essa medida de dissimilaridade foi contraposta com possíveis opções: distância de Mahalanobis, distância Chebychev e distância cosseno. Quanto maior a distância entre as distribuições, maior sua dissimilaridade, o que pode indicar um desvio e consequentemente a necessidade de mudanças no modelo de predição. Dentre outros métodos mencionados como base de comparação, estavam o DDM, ADWIN, HDDM e EWMA.

Modelos de Redes Neurais Convolucionais (CNNs) foram aplicados por DIsabato and Roveri [2019] como sistemas de aprendizado de máquina adaptáveis capazes de operar em condições não estacionárias. Redes Neurais Convolucionais normalmente são utilizadas para aprendizado utilizando imagens. Este trabalho apresenta um mecanismo adaptativo para aprender CNNs capazes de operar na presença de desvios. Este é um trabalho menos relacionado ao nosso tema, mas que mostra a necessidade de tratamento de desvio nas mais diversas aplicações. Aqui, em modelos de predição utilizando CNNs, que pelo alto poder de processamento e memória necessários teriam vantagem em contar com um método de detecção de desvio de conceito ativo, pois um método passivo seria muito custoso. Optaram pela utilização do teste não paramétrico CUSUM adequadamente modificado para operar em distribuições Bernoulli.

Um outro trabalho com foco na melhoria de métodos para aprender redes neurais profundas utilizando Redes Neurais Convolucionais (CNNs) foi trazido por Duda et al. [2020]. Neste, aplicaram-se artifícios normalmente utilizados em fluxos de dados como resumo das amostragens, que são reduções de conjuntos de dados que os representam. Substituindo a aprendizagem baseada na épocas criando aleatoriamente mini-lotes subsequentes permitidos o treinamento da rede neural para ser mais eficiente. A ideia tirada do algoritmo de *boosting* resultou em uma diminuição mais rápida da função de perda. No entanto, descobriu-se que a proposta técnica poderia resultar na

incidência de mesmos mini-lotes para distribuições diferentes. A aplicação de um detector de desvio ajudou a eliminar este problema efetivamente. O CUSUM foi utilizado como detector de desvio. O trabalho conclui que a aplicação de um detector de desvio dedicado pode permitir que o sistema reaja melhor às mudanças no modelo treinado.

Em seu trabalho sobre detecção de desvio de conceito Duda et al. [2020], trouxeram um método ativo que implementa soluções inspirado no algoritmo Random Forest para os cenários de fluxo de dados. O proposto algoritmo combina a capacidade de métodos ensemble para lidar com mudanças lentas em fluxos de dados contínuos com um novo método para detectar a ocorrência de desvio. O método aqui foi chamado de Random Forest with Features Importance (RFFI) e, diferente dos demais estudados neste trabalho, é a relevância dada à seleção de variáveis observadas. Para realizar a árvore de decisão, ele considera diferentes pesos, o que chamou de importância, para fazer a seleção de features. Embora patente que a seleção de variáveis observadas seja de suma importância para o modelo de predição e para modelos de detecção de desvio, a maioria dos trabalhos não faz menção a parte de seleção de *features* em seus modelos. Como descrito em seções anteriores, a mudança de contexto pode ocorrer pela mudança de pesos entre variáveis observadas e entrada de variáveis não observadas até então. Nesta publicação, a alteração na árvore de decisão e tratamento de desvio ocorre na detecção de mudanças em pesos de variáveis, que funciona como seleção de contexto. Dentre os métodos utilizados para detecção de desvio na Adaptive Random Forest (ARF) estavam o ADWIN, CUSUM, DDM e EDDM.

Um novo método chamado de Multiscale Drift Detection Test (MDDT), ou detecção de desvio em multiescala foi proposto por Wang et al. [2018]. Este é um modelo de detecção de desvio de conceito ativo que ocorre com dados após o pré-processamento, antes de ser aplicado o modelo de aprendizado de máquina. Ele se apoia na suposição de que existe um ou um conjunto de características, ou variáveis, que podem servir para monitorar a estabilidade da distribuição. Com base nas diferenças entre essas variáveis entre janelas de tempo, utiliza-se o teste t de student, localizam-se desvios abruptos durante o fluxo de dados. Chamou-se de multiescala pelo fato do processamento das janelas apresentar seleção de dados para visões resumidas das distribuições. Primeiro existe uma seleção das janelas, uma estacionária e outra nova, com todos os dados, onde é processado o MDDT. Caso exista diferença entre as distribuições, aferida através do teste t de student, o MDDT é processado novamente em uma escala menor, reduzida, das distribuições para confirmar ou não o desvio. Embora este modelo busque a simplicidade e resolver problemas de custo de processamento, tempo e memória, muito sensíveis para fluxos de dados, acaba apresentando análise limitada para desvios abruptos. Dentre os métodos citados como comparação estava o HDDM.

Encontramos modelo de regressão baseado em agrupamento difuso (Fuzzy c-means) em Song

et al. [2019], porém, neste caso, no lugar de sintetizar padrões como se fosse um resumo de contextos, os autores incluem continuamente as instâncias de dados mais relevantes no conjunto de treinamento. Eles se propõem a fazer isso com um processo de identificação contínua e retirada de ruídos. Esta identificação e retirada sistemática de ruídos faria com que a identificação do desvio ficasse mais acurada já que não contaria com dados que poderiam afetar as análises, tratando-se de *outliers*. O mesmo modelo proposto efetua a identificação do desvio, sendo um modelo ativo. Dentre os métodos citados como comparação estava o ADWIN.

Em um camada acima do problema observado, Song et al. [2020] efetuaram avaliação de desvio entre fuxos de dados correlacionados. Em seu trabalho, apresentaram um método denominado de Fuzzy Drift Variance (FDV) para medir correlação de desvio entre diferentes fluxos de dados. Este trabalho também utiliza Fuzzy c-means e efetuou testes em dados sintéticos com a hipótese de um sistema com vários fluxos de dados relacionados cuja identificação de desvio deveria ser tratada de forma conjunta. Este tipo de tratamento é análogo ao citado em Song et al. [2019], mas considera um cenário sistêmico onde vários fluxos de dados podem estar relacionados.

Um método ativo de adaptação de modelo utilizando Séries Temporais Fuzzy fora trazido por de Lima e Silva et al. [2020]. Normalmente Séries Temporais Fuzzy não são utilizadas para conjuntos de dados não-estacionários, o que torna a aplicação não aderente as mudanças de contextos e à sua adaptação ao desvio. Este trabalho trouxe uma Série Temporal Fuzzy Não-estacionária, no qual funções de perturbação são usadas para adaptar a associação de parâmetros de função na base de conhecimento em resposta a mudanças estatísticas nas séries temporais. As funções de perturbação tentam definir as mudanças ocorridas entre distribuições e adaptar parâmetros. O método proposto se mostrou resiliente ao desvio, adaptando parâmetros do modelo, preservando a estrutura simbólica da base de conhecimento.

Uma nova metodologia para reduzir o erros de generalização que ocorrem devido à mudança de domínio na classificação dos dados foi apresentada por Krishnan et al. [2019]. Esta redução é alcançada por introdução de uma mudança de domínio adequadamente selecionada neste modelo para os dados de treinamento através do que é referido como "modelo de distorção". Essas distorções são introduzidas por meio de uma transformação dos dados originais afim de aumentar o domínio de dados observados em treinamento e antecipar mudanças de contexto. Em seguida, uma rede neural profunda é usada para classificar o domínio aumentado. Aprendendo com o original e amostras de dados adicionais, o classificador compensa a mudança de domínio, mantendo seu desempenho nos dados originais. O objetivo é compensar a mudança ideal que pode ser introduzida pelo modelo de distorção sem significativamente degradar o desempenho do modelo. Finalmente, para efetuar essa equalização do que seria o limite de crescimento entre dados originais e distorções, existe um controle de taxa de erros feito a posteriori que compara os resultados reais aos resultados

do classificador. Este trabalho não é exatamente um conceito de detecção de desvio, mas sim uma forma de detecção entre distribuições e que tem seu modelo de classificação embutido. Mas incluímos nesta seção de trabalhos relacionados por apresentar uma ideia válida de tratamento ativo que seria a introdução de ruídos intencionais nos dados originais antes do treinamento, aumentando o domínio e tornando o modelo mais resiliente à ocorrência de desvios.

Outro trabalho que foca no problema de desvio de conceito propõe uma classificação e um novo algoritmo de detecção de classe com base no índice de coesão e separação de Mahalanobis Li et al. [2020a]. Em seu método, primeiro é usado k-means para o agrupamento de dados por sua proximidade e em seguida utilizada a distância de Mahalanobis para teste de limite entre domínios. Caso o número de ocorrências fora do limite seja maior que um limite predeterminado, existe a sugestão de desvio e o modelo é retreinado. Proposta semelhante foi apresentada em Garcia et al. [2019], onde o algoritmo Higia foi introduzido. O Higia encontra microclusters próximos para uma nova instância e classifica a instância com base na distância para o microcluster mais próximo. O método usa o raio e um limite para lidar com mudanças graduais no fluxo de dados e um buffer para lidar com novidades. O algoritmo Higia usa a distância euclidiana para calcular a distância entre uma nova instância e os microclusters mais próximos.

Em Mahdi et al. [2020], foi proposto um novo modelo chamado Diversity Measure Drift Detection Method (DMDDM) que combina a aplicação do teste Page-Hinkley [Page, 1954] com a medida de desacordo (disagreement measure), que estima o número de observações na qual o classificador está incorreto. O método foi comparado com dez métodos de detecção de desvio em diferentes cenários usando vários conjuntos de dados, onde apresentou bom resultado. Inclusive referentes ao consumo de memória e necessidade de poder de processamento. Este é um modelo ativo que utiliza um dos testes considerados como pioneiros, adicionando cálculo da medida de desacordo para identificar mudanças relevantes nas distribuições. Em seus testes o método utilizou bases de dados reais históricas de cinco áreas diferente e comparou seus resultados nessas mesmas bases com DDM, ADWIN, HDDM, FHDDM e PH Test.

Algoritmos genéticos foram utilizados em Jedrzejowicz and Jedrzejowicz [2020] para estruturar um método de classificação e identificação de desvio, validado no experimento computacional com vários conjuntos de dados reais e sintéticos. Para detecção de desvio foi proposto teste de soma de postos de Wilcoxon, que é um teste não paramétrico que pode ser usado para determinar se duas amostras independentes foram selecionadas a partir de populações que têm a mesma distribuição. Este trabalho demonstra eficácia e performance similares a demais modelos existentes, mas apresenta uma diferença fundamental. Grande parte dos modelos que estudamos até então são paramétricos e tomam como base uma distribuição normal. Isso faz com que seja necessário uma adaptação dos dados de saída dos modelos de predição, o que pode causar maior gasto ou até perda

de informações. Este trabalho traz um método ativo não paramétrico de detecção de desvio onde não é mandatório a utilização de distribuições normais. Dentre outros métodos mencionados como base de comparação, estavam o DDM e RDDM.

Mais recentemente Liu et al. [2023] levanta o problema atual e motivação dos estudos em desvio de conceitos na classificação para fluxo de dados, onde os problemas mais comuns são: identificação de desvios de conceito, ruídos e os custos exorbitantes associados à rotulagem de instâncias não rotuladas em fluxos de dados. Este trabalho propõe uma estrutura de aprendizagem ativa baseada em uma estratégia de dupla consulta e a lei da memória humana de Ebbinghaus, denominada de CogDQS. Não se trata apenas de um método para detecção de desvio de conceito, mas todo um modelo de classificação que embute a identificação e tratamento do desvio. A política para discernir o desvio do ruído e a substituição de instâncias desatualizadas por novos conceitos é baseada em os três critérios da curva de esquecimento de Ebbinghaus: recordação, o período de desbotamento e a força da memória. O CogDQS conseguiu mostrar que essa abordagem produz modelos precisos e estáveis com capacidade de generalização baixo custo de armazenamento e processamento computacional, em comparação com os modelos AOBagging, AOBoosting, DACC e ADOB. Todos os testes e comparações foram feitas através do Massive Online Analysis (MOA) [Bifet et al., 2010], que é uma plataforma aberta para análise de fluxo de dados, mesma origem dos métodos HDDM, FHDDM e RDDM. Embora seja um modelo de classificação com a identificação e tratamento do desvio de conceito, ele o faz através da análise das distribuições de dados, no caso fluxo de dados.

Para tratar o problema de análise de fluxo de dados não categorizados com a preocupação sobre capacidade computacional e memória, Hoeltgebaum et al. [2022] criaram o Real-time Adaptative Component (RAC), um modelo de regressão ponderada, que combina o classificador com um método de detecção de anomalias, desvios de conceito. Como componente principal o modelo RAC utiliza LASSO como modelo de regressão. Como complemento ao modelo, introduz a utilização do método Hall–Buckley–Eagleson (HBE). Onde existe função de distribuição cumulativa (cdf) de uma soma ponderada positivamente de variáveis aleatórias qui-quadradas usando o método HBE. Então considera-se um nível de significância α que indica o desvio. O RAC pode ser considerado um método ativo e que efetua a detecção por características nas distribuições dos dados. Os experimentos foram separados em duas linhas, na classificação e na identificação de desvios de conceito. O modelo foi comparado aos métodos de desvio de conceito CUSUM e EWMA dentre outros.

Vemos em trabalhos como Sebastião and Gama [2007] na aplicação para detecção de mudança nos exemplos de geração de distribuição de fluxos contínuos de dados. Neste trabalho o cálculo da divergência foi utilizado para resolver o problema de construção de histogramas a partir de fluxos de dados de alta velocidade que mudam com o tempo. Nele, foram comparadas as mesmas distribuições

sinalizando a mudança após atingimento de um determinado limite usando três métodos diferentes: a diferença absoluta de entropia, a divergência de Kullback-Leibler e a distância cosseno. Os resultados experimentais indicam que a divergência de Kullback-Leibler exibe alta probabilidade na detecção de alterações, taxas de detecção mais rápidas, com poucos alarmes falsos positivos.

IV.1.2 Análise de desempenho

Nesta subseção estão relacionados trabalhos que tratam do assunto desvio de conceito através da análise de desempenho. Os estudos aqui apresentados detectam e ou tratam o desvio de conceito usando análise direta nas taxas de erros e acertos. Essa análise sempre será feita contando com o fator de erro do classificador.

Um novo modelo de detecção de desvio de conceito foi proposto por Lin et al. [2019], o Linear Four Rates (LFR). O método foi proposto para análise de detecção de desvio de conceito para avaliar desempenho de monitores IoT. Dentre outros dois modelos de comparação, o método foi comparado ao DDM, citado como método mais popular. O método LFR tem mecanismo análogo ao DDM, mas busca aprimorar a identificação de desvio que ocorre através da medida de desempenho do classificador. O DDM se concentra na taxa de erro geral e, portanto, falha em detectar um desvio, a menos que a soma dos falsos positivos e falsos negativos mude. O LFR atribui pesos diferentes para as classificações para resolver este ponto.

Dornadula and Geetha [2019] desenvolveram um novo método de detecção de fraude em cartões de crédito para dados em tempo real. O modelo classifica padrões de comportamento, onde os titulares do cartão são agrupados em grupos diferentes com base no valor da transação. O método usa estratégia de janela deslizante para agregar as transações feita pelos portadores de cartões de diferentes grupos de forma que o padrão de comportamento dos grupos possa ser extraído respectivamente. Ele usa um mecanismo de feedback para detecção de desvio de conceito calculando acurácia, precisão e o Coeficiente de Correlação de Matthews (MCC).

Incorporado a modelo de classificação de preço para tarifa elétrica, Ng et al. [2019] implementaram uma detecção de desvio de conceito. A adaptação ao desvio de conceito feita pela aferição de desempenho, onde a classificação é ponderada pelo custo da classificação. Ou seja, o valor que pode ser incorporado ao impacto de uma classificação errada. Este novo modelo não apresenta a complexidade, ou alguma base de comparação com os modelos do estado da arte apontados nesta tese. Sua novidade está na ponderação pelo custo do desempenho, pelo viés de perda ou ganho financeiro e pelo direcionamento voltado para aplicação direta em área de negócio.

Song et al. [2020] apresentaram modelo de detecção de desvios em fluxos contínuos de dados via regressão utilizando cluster baseados em lógica fuzzy. O agrupamento difuso é uma forma de clustering onde cada um dos elementos pode pertencer a mais de um cluster, e por isso é dito

como soft. A nova abordagem de regressão adaptativa foi chamada de FUZZ-CARE, utilizada para reconhecer, treinar e armazenar padrões dinamicamente e atribuir o grau de adesão dos próximos exemplos pertencentes a estes padrões. Em vez de projetar um procedimento complicado para continuamente buscar o padrão mais recente, sendo uma abordagem comum na literatura, FUZZ-CARE resume informações anteriores úteis para ajudar prever exemplos recém-chegados. Neste caso, a diferença deste modelo para outras propostas similares é tentar sumarizar características, ou contextos, previamente para facilitar a classificação de novas informações, e com isso também a identificação de mudanças na distribuição, tendo como base os contextos definidos na estruturação do modelo. Podemos considerá-lo também um modelo ativo por se tratar de uma identificação feita durante a classificação, onde as distribuições de dados de entrada são estudadas em relação à classificação existente no sentido de identificação de novo contexto. Em termos de detecção de desvio de conceito é um modelo mais simples, que utiliza o erro médio absoluto, dados de predição em determinado período e os dados reais de ocorrência.

Em outro trabalho integrado onde é apresentado o modelo já com uma proposta de detecção de desvio de conceito, Le et al. [2019] avaliaram lista de vulnerabilidades em abordagens usando Processamento de Linguagem Natural (PNL). Neste modelo, existe o treinamento em seis métodos diferentes: Naıve Bayes, Regressão, Support Vector Machine (SVM), Random Forest (RF), XG-Boost - Extreme Gradient Boosting (XGB) e Light Gradient Boosting Machine (LGBM). A forma de detecção de desvio de conceito é feita por observação direta de medidas de desempenho, sem efetuar retreinamento. O modelo é selecionado para a classificação em função da última medida de desempenho aferida.

Em seu trabalho Tian et al. [2019] propõem novo método que utiliza One-Class Support Vector Machine (OCSVM) para detecção de desvio. Máquinas de vetores de suporte (SVMs) são modelos de aprendizado supervisionado que analisam dados e reconhecem padrões e podem ser usados para tarefas de classificação e regressão. Normalmente, o algoritmo SVM recebe um conjunto de exemplos de treinamento rotulados como pertencentes a uma de duas classes. O algoritmo SVM representa os exemplos como pontos no espaço, mapeados de forma que os exemplos das categorias separadas sejam divididos por uma lacuna clara que seja a mais ampla possível. Novos exemplos são mapeados nesse mesmo espaço e considerados pertencentes a uma categoria ou outra, com base em qual lado da lacuna eles se enquadram. Portanto, em SVM de uma classe, o modelo de vetor de suporte é treinado em dados que possuem apenas uma classe, sendo a classe “normal”. Ele infere as propriedades de casos normais e, a partir dessas propriedades, pode prever quais exemplos diferem dos exemplos normais. Porém, apesar de amplamente utilizado para detecção de anomalias, OCSVM ainda tem o desafio de lidar com sistemas não estacionários de fluxos de dados, onde as distribuições subjacentes de dados variam com o tempo. Pra resolver esse problema,

é proposto aprendizagem incremental baseada em OCSVM. Para uma amostra de entrada, existe uma comparação relativa com três conjuntos de vetores em OCSVM: vetores de suporte de margem, vetores de suporte de erro e vetores de reserva. Este método é outro exemplo de método ativo que utiliza a medição de desempenho, através do vetor de erros, para identificar desvio de conceito.

O Dynamic Updated Ensemble (DUE) foi proposto por Li et al. [2020b] para resolver o problema de aprendizagem de fluxos de dados desequilibrados com desvio. O DUE é um método baseado em *bagging* que seria a aplicação da combinação do resultado de mais de um modelo de aprendizado de máquina para fazer previsões mais precisas do que qualquer modelo individual. O framework é usado para gerar classificadores candidatos para cada conjunto de treinamento. Ele cria vários classificadores candidatos para cada bloco e os pondera usando um método de ponderação por partes. O peso dos membros do conjunto são baseados em seu desempenho nos exemplos mais recentes e no nível de estabilidade. Esses contextos testados são periodicamente atualizados para fazer o conjunto reagir a diferentes tipos de desvio. Em seus experimentos, o método apresentado concentrou-se principalmente em duas classes, mas entende-se que poderia ser estendido para questões multi-classe. Este método de detecção de desvio parte da análise de cenários e usa uma árvore de decisão para escolha do melhor método de aprendizagem de máquina dados resultados de eficiência aferidos em testes anteriores. Diante da similaridade da distribuição de dados, o melhor método é escolhido. Os contextos e eficiência dos modelos são constantemente recalculada.

Jiang et al. [2022] montou método de classificação ensemble proposto para melhorar a precisão da previsão para IoT verde. Especificamente, um classificador fuzzy baseado em regras é combinado com um algoritmo de ponderação dinâmico para melhorar a precisão da classificação. Em solução que trabalha com fluxos de dados no ambiente de Internet das Coisas (IoT) verde, no controle inteligente do interruptor de luz, na previsão de lâmpadas economizadoras de energia em edifícios inteligentes. Embora não seja em si um método ativo de detecção de desvio de conceito, mas sim um método passivo, que tenta mitigar o desvio com um modelo mais dinâmico, selecionamos este trabalho pelo seu fato apresentar também um modelo atualizado por aprendizado incremental onde mudanças nos fluxos de dados são associados a diferentes modelos e existe a proposta de reaproveitamento. Esta é uma característica que não identificamos em nenhum dos métodos de desvio de conceito até então estudados, mas que existe em nossa proposta de forma mais estruturada. Basicamente, aqui, além de um modelo que tenta atenuar o desvio de conceito, existe a tentativa de reaproveitamento de modelos para conceitos conhecidos, evitando o retreinamento e tratando desvios recorrentes.

Mais um exemplo de modelo de aprendizado de máquina adaptativo é introduzido por Wang et al. [2022]. Eles apresentam novo modelo de adaptação ao desvio de conceito, chamado iterações adaptativas (AdIter), que escolhe automaticamente o número de iterações para diferentes gravidades

de desvios para melhorar a precisão da previsão para fluxos de dados. O modelo de aprendizado base é o Gradient Boosting Decision Tree (GBDT), o qual foi adaptado. Para reduzir o impacto do desvio de conceito, o modelo propõe dois níveis de tratamento, o de corte de classificadores fracos e o de retreinamento, neste último quando se identifica que o desvio é muito grande para ser tratado sem retreino. Basicamente existe a identificação de número de classificadores fracos no início do modelo e caso esse número aumente localmente, existe o corte de classificadores que não estão apresentando com desempenho. No caso desse número de classificadores ultrapassar um limite, existe o retreino. O AdIter não é isoladamente um método de detecção e tratamento de desvio de conceito, mas podemos classificá-lo por analogia como um método ativo e por desempenho, já que efetua a detecção e tratamento considerando o desempenho dos classificadores. E embora a comparação seja feita com demais modelos adaptativos como ARF Hoeffding Tree Classifier, OnlineBoosting KNN ADWIN Classifier, OnlineRUSBoost KNN ADWIN Classifier, OnlineBagging KNN ADWIN Classifier e SRP Hoeffding Tree Classifier, vemos que a base para comparação foi muito semelhante ao método ADWIN.

IV.2 Desvio de conceito na predição de atrasos aéreos

A literatura anterior sobre previsão de atraso é extensa [Carvalho et al., 2021]. Foi realizado um estudo de mapeamento sistemático para identificar estudos de ciência de dados relevantes sobre a previsão de atraso de voo com string de pesquisa usada: ("flight delay") AND ("classification" OU "regression" OU "prediction")). Foram considerados apenas artigos (Revistas e Conferências) inteiramente escritos em inglês. A consulta foi submetida à base de dados Scopus (considerando título, palavras-chave e resumos) em dezembro de 2020 e retornou 141 artigos. Além disso, dois estudos foram adicionados por meio de *snowballing*.

Em nossa pesquisa, a maioria absoluta dos estudos utiliza o cenário de dados estáticos para treinamento e teste, um cenário que desconsidera o desvio de conceito [Chen et al., 2008; Khan-mohammadi et al., 2014; Alonso and Loureiro, 2015]. Porém, encontramos seis estudos que já consideram as informações como fluxo de dados, ou seja, como janelas de tempo, e abordam o desvio de conceito [Kim et al., 2016; Khamassi and Sayed-Mouchaweh, 2014; Pesaranghader and Viktor, 2016; Munoz Hernandez et al., 2019; Wang et al., 2019; Ai et al., 2019]. Embora, de fato, apenas os três primeiros investiguem o problema do classificador binário.

Kim et al. [2016] usaram deep neural networks para prever atrasos em voos nos EUA usando janelas de sete e nove dias antes da previsão. Uma rede neural profunda adicional foi treinada para classificar o status crítico de atraso diário. Uma vez que a escala utilizada era de rotas de voo, essas previsões foram adicionadas aos dados individuais de rotas de voo e usadas como entrada em uma

Rede Neural Profunda. Embora a precisão tenha sido superior a 80% na maioria dos experimentos, métricas-chave como f1, precisão e recall não foram relatados. Como nenhuma estratégia específica foi usada para identificar ou lidar com desvio de conceito ativamente, consideramos essa abordagem como uma estratégia passiva, um modelo adaptativo.

Khamassi and Sayed-Mouchaweh [2014] propuseram uma nova abordagem baseada em distância ativa de erro para detecção de desvio e monitoramento, chamado EDIST. Especificamente, o EDIST compara novos dados com os dados existentes e retreina o modelo se mudanças significativas são encontradas na distribuição. O classificador usado neste caso foi o Hoeffding Trees. Estratégias ativas do Drift Detection Method (DDM) e do Early Drift Detection Method (EDDM) foram implementadas por motivo de comparação. Eles aplicaram essas técnicas a muitos conjuntos de dados sintéticos e do mundo real (incluindo o conjunto de dados de voo dos EUA). No entanto, apenas a precisão foi usada para relatar o desempenho da previsão.

Pesaranghader and Viktor [2016] em seu trabalho de introdução do Fast Hoeffding Drift Detection Method (FHDDM), testaram muitas técnicas para detecção de desvio de conceito. Especificamente, seu trabalho propõe usar o Teorema da Desigualdade Hoeffding para testar a diferença na probabilidade de uma determinada classe entre duas distribuições de 25 casos, caracterizando uma estratégia ativa. Os métodos DDM, EDDM, ADWIN e HDDM foram implementados por razões de comparação. A precisão relatada estava em torno de 65% para todos os experimentos com dados de aviação. O método proposto apresentou melhores resultados para os conjuntos de dados de voo dos EUA, em comparação com os demais modelos.

IV.3 Análise

Com base na extensa pesquisa que realizamos, podemos afirmar que a maioria dos trabalhos sobre desvio de conceito promovem um estudo agnóstico. Esses estudos têm como objetivo estruturar uma forma flexível de detecção de desvio de conceito que se adapte aos diversos modelos de aprendizado de máquina. E também que sejam independentes da aplicação, por exemplo, que tenham bom desempenho com detecção de desvios tanto em modelos de predição de atrasos aéreos, como quaisquer outras aplicações de modelos de classificação.

Os trabalhos com aplicação direta em determinada área de atuação, como os citados na subseção acima, como de Dornadula and Geetha [2019] e Ng et al. [2019], são mais raros. Assim como são mais escassos os trabalhos com uso de bases de dados reais [Lu et al., 2019]. A maioria dos trabalhos pesquisados nesta tese, não apenas os citados neste capítulo, utilizam bases de dados comuns, públicas e históricas, e na sua maioria sintéticas.

Verificamos que os métodos de detecção de desvio de conceito não oferecem alguma relação clara

com métodos de aprendizagem de máquina utilizadas em modelos de predição. Diversos métodos foram utilizados nesse modelo, e em análises correlatas, efetuando complementação com um ou mais métodos de detecção de desvio de conceito. A Figura IV.1 mostra os métodos de aprendizado de máquina utilizados nas pesquisas de trabalhos relacionados, que mostra uma prevalência em métodos de agrupamento, onde incluímos Fuzzy c-means e K-means, seguidos por métodos de redes neurais, contemplando aqui também redes neurais convolucionais. Outros métodos utilizados foram Random Forest, SVM, Hoeffding Trees e Naive Bayes.

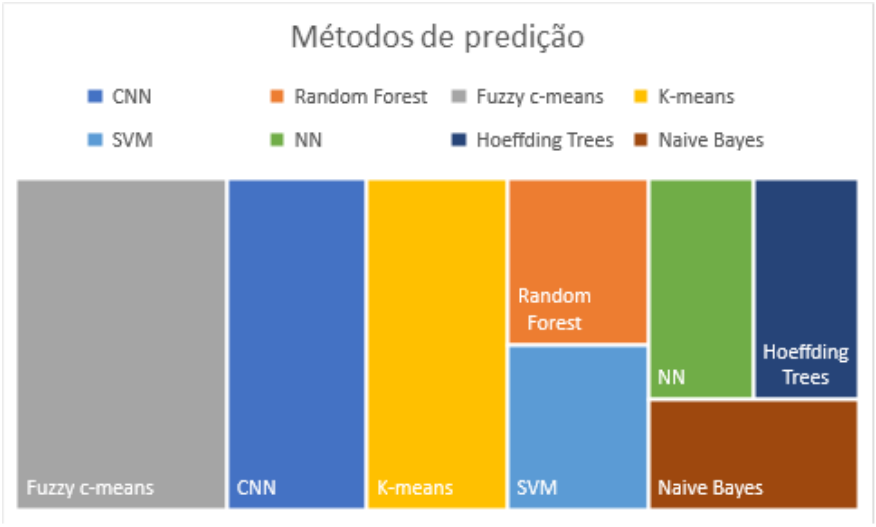


Figura IV.1: Principais modelos de predição citados nas publicações de trabalhos relacionados.

Com relação aos métodos de detecção de desvios utilizados ou mencionados como base de comparação, apresentados abaixo na Figura IV.2 , verificamos que os métodos prevalentes são aqueles que utilizam janelas deslizantes, com Adwin, HDDM, EDDM, HDDM e FHDDM; seguidos por abordagens baseadas em controle estatístico de processos, como DDM, EDDM, RDDM e EWMA; e, na ordem de mais utilizados, abordagens de análise sequencial como CUSUM e Page-Hinkley.

Modelos adaptativos e combinados, modelos com detecção explícita do desvio, têm papel crescente nas recentes propostas sobre desvio de conceito. Em contraste, a pesquisa de reciclagem de modelos com detecção explícita de desvio de conceito diminuíram. Sendo que são cada vez mais desejados modelos com detecção explícita, capacidade de tratamento de fluxos de dados reais e integração com as metodologias de aprendizado de máquina [Lu et al., 2019].

Conforme exposto anteriormente, a forma de classificação mais básica de desvio é a feita por através da identificação da diferença entre as distribuições ou desempenho. A primeira baseia-se na hipótese de que se as distribuições apresentarem diferença além de um limite, existe desvio de conceito. Neste ponto estão métodos do estado da arte ADWIN, SeqDrift, CUSUM e Page–Hinkley. Outros exemplos de modelos que encontramos em nossa pesquisa foram Sebastião and Gama [2007] que usaram a dissimilaridade Kullback–Leibler; t2 de Hotelling [Hoens et al., 2012]; verossimi-

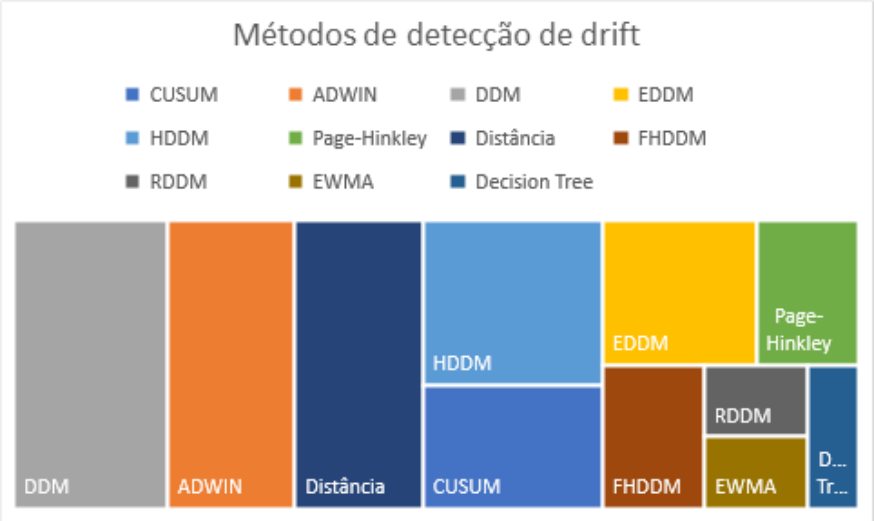


Figura IV.2: Principais métodos de detecção de desvio de conceito citados nas publicações de trabalhos relacionados.

lhança logarítmica semi-paramétrica (SPLL) [Kuncheva, 2013]; Dos Reis et al. [2016] que usou o teste Kolmogorov–Smirnov; Li et al. [2020a] e Yang et al. [2020] que utilizaram a distância de Mahalanobis; Jedrzejowicz and Jedrzejowicz [2020] com o teste Wilcoxon; e, distância Chebychev e também distância Cosseno em Yang et al. [2020].

Na forma de identificação por desempenho, existe a identificação de necessidade de mudança e retreinamento do modelo a partir da monitoração da evolução de algum indicador de desempenho, como a taxa de erro, para aplicar heurísticas e encontrar pontos de mudança. Por exemplo, o método DDM considera três estados diferentes para a classificação de evolução do erro: (i) sob controle, quando o erro é estável; (ii) fora de controle, quando o erro aumenta significativamente em comparação com o passado recente; e (iii) advertência, quando o erro está aumentando, mas não atingiu o estado fora de controle. EDDM também considera a distância entre erros no lugar de considerar apenas a magnitude do erro. Outros métodos que podemos usar como controle de desempenho do processo são: EWMA, RDDM, HDDM, FHDDM e MDDM. Os métodos empregam um conjunto de regras considerando distribuição dos erros do classificador para definir o limiar dos estados.

Conforme colocado no Capítulo III, podemos dividir Desvio de Conceito em tipos Abrupto, Gradual, Incremental e Recorrente. Verificamos também no Capitulo IV que o estado da arte normalmente trabalha bem com o tipo abrupto e está avançando na detecção da forma gradual, sendo mais complexo a identificação de desvios incrementais e recorrentes. Lembrando aqui que os tipos Abrupto são identificados logo após sua ocorrência, onde um conceito A é instantaneamente substituído por um conceito B. Os Graduais onde existe a mistura os conceitos A e B, tornando-os ativos por um determinado período, mas, ao longo do tempo, o número de ocorrências do conceito A diminui à medida que o conceito B aumenta. Incremental onde as mudanças são identificadas

lentamente. Nesse tipo de desvio, as diferenças acontecem gradualmente, sendo percebidas apenas em longos períodos. E a recorrência pode ser caracterizada como variações de estados entre válido e inválido com certo grau de regularidade.

Entendemos que em diversas áreas, como a área de aviação, e variação de atrasos em voos, podem ocorrer todos os tipos de desvios e, por isso, quanto mais abrangente a possibilidade de identificação de desvios, independente do tipo, melhor. Neste sentido, nossa proposta visa aumentar o espectro de atendimento de modelos de identificação de desvio de conceito do estado da arte, com o desenvolvimento de técnicas de armazenamento e seleção de modelos já treinados visando melhorar a atuação e desempenho particularmente em desvios recorrentes. Neste tipo de identificação, existe uma seleção ativa de modelo de aprendizado de máquina de acordo com pesos e formas de distribuições de variáveis de entrada.

Capítulo V Kairos - Método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos

Neste capítulo é apresentado o método Kairos, um método ativo para detecção e tratamento de desvio de conceito que complementa os modelos de predição de atrasos aéreos. Conforme colocado no Capítulo II, o campo de atrasos aéreos é amplamente estudado, mas apenas mais recentemente apresentou interesse na aplicação de métodos de aprendizagem de máquina na construção de sistemas de predição. O número de trabalhos vem crescendo e encontramos diversos tipos de modelos sendo desenvolvidos. No entanto, identificamos que existe carência de aplicações práticas ou indicação de melhor forma de uso para esta área. Grande parte das publicações e aplicações sobre aprendizagem de máquina foca nos métodos e em comparações com demais formas de tratamento para casos gerais, sem aplicação específica para determinada área, como a de voos comerciais.

O termo "Kairos" na mitologia grega relaciona-se com o aspecto qualitativo do tempo, significando momento oportuno, ocasião certa, oportunidade [Miller, 1994]. A inspiração para o nome atribuído ao método representa a busca pelo melhor momento para identificação e tratamento do desvio.

V.1 Visão geral do Kairos

A Figura V.1 apresenta uma visão geral do Kairos, a qual detalharemos durante este capítulo. Esta visão geral começa com a etapa inicial de treinamento e testes, que inclui as atividades de (1) pré-processamento inicial dos dados e (2) treinamento de um modelo inicial de aprendizado de máquina. A atividade de pré-processamento é alimentada por dados históricos; após a atividade de treinamento, temos um modelo de classificação treinado. A seta (a) representa o final da etapa de treinamento e testes, que ocorre quando o modelo treinado alcançou as medidas de desempenho desejadas pelo processo específico de treinamento empregado.

O método Kairos mantém uma tabela de registros em que cada entrada associa um modelo treinado a uma caracterização da distribuição dos dados usados para o treinamento desse modelo (janela de dados). Essa tabela é usada para calcular a divergência (ou distância) entre distribuições de diferentes janelas de dados e identificar que modelo treinado deverá ser usado para futuras predições. Na etapa inicial de treinamento e testes essa tabela ainda se encontra vazia; o cálculo da divergência nesse caso é feito entre os dados de treinamento e de teste usado na construção do

modelo inicial de aprendizado de máquina. Essa distância de referência será usada como limite de comparação posteriormente. Ao final dessa etapa, são armazenados na tabela de registros o modelo treinado, a distribuição dos dados de teste e a distância calculada entre os dados de teste e os de treinamento, representados pela aresta (b). Esta é a primeira entrada na tabela de registros.

O retângulo com borda mais fina assinalado com o título "Ciclo do Kairos" delimita uma etapa contínua com atividades cíclicas de tratamento e classificação do fluxo de dados de entrada. A atividade (3) é a de pré-processamento de novas janelas de dados obtidos em um determinado período, o que pode variar de processo para processo. Ao final do pré-processamento de uma nova janela de dados, ocorre o cálculo da distância entre essa nova janela de dados e as janelas de referência armazenadas em cada entrada da tabela de registros (4). A aresta (c) representa a leitura de informações de distribuições e distâncias de referência armazenadas na tabela de registros.

A distância calculada entre a nova janela de dados e a janela de referência de uma entrada na tabela de registros é comparada com a distância de referência associada à entrada. A distância de referência determina um limite: se a distância calculada for inferior a esse limite, considera-se que o modelo armazenado na entrada correspondente da tabela de registros é adequado para efetuar classificações no fluxo de dados de entrada atual. Nesse ponto, diversas situações podem ocorrer, desde não se identificar modelo adequado algum na tabela de registros, até se identificar múltiplos modelos adequados. Nas seções seguintes são apresentados os critérios utilizados para tratar essas diferentes situações. O passo (5) é o responsável pela aplicação desses critérios, que podem levar à manutenção do modelo corrente, ou à seleção de um modelo presente na tabela de registros – aresta (d) – ou ainda ao treinamento de um novo modelo – passo (7) – a ser inserido na tabela – aresta (e). O passo (6) representa a classificação dos dados do fluxo de entrada utilizando o modelo selecionado.

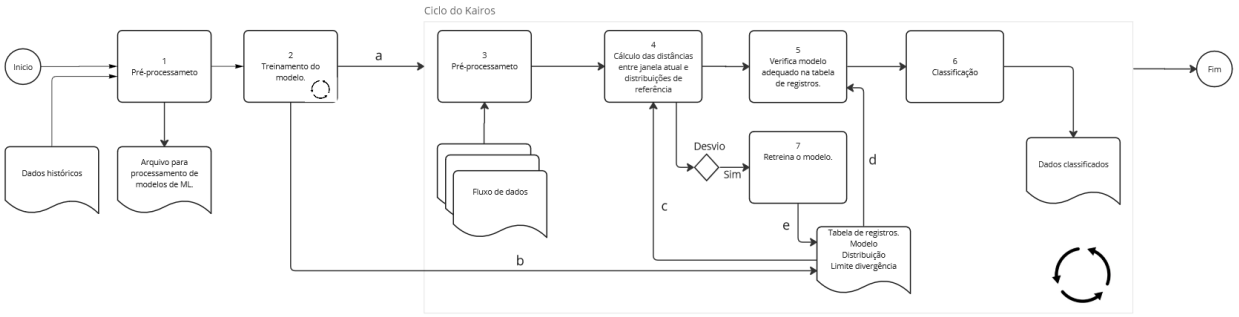


Figura V.1: Visão geral do método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos - Kairos.

V.2 Etapas detalhadas

A Figura V.2 mostra o esquema de aprendizado de máquina com as atividades de detecção e tratamento de desvio de conceito propostas pelo Kairos. Abaixo, segue descrição dos passos:

1. Etapa de definição do modelo de aprendizado de máquina.

    1.1 Etapa de pré-processamento, com as modificações de dados brutos necessárias ao processo.

    1.2 Na etapa de segregação de dados é onde existe a divisão dos dados brutos em dados de treino e testes.

    1.3 Na etapa de treinamento, o modelo é treinado com os conjuntos de dados segregados para treinamento.

    1.4 Na etapa de testes, o modelo é testado com os conjuntos de dados segregados para testes.

    1.5 Nesta etapa, existe a redução de dimensionalidade e cálculo de distância entre as distribuições de treinamento e teste. Ao final desta etapa, quando o ciclo for finalizado com o bom desempenho do modelo, armazenaremos a última distribuição de referência.

2. Ciclos de processamento do modelo, com a classificação de dados entrantes em fluxo de dados contínuo. Nesta fase, já temos um registro na tabela de registros de modelos e divergências. O primeiro registro com o modelo treinado, distribuição de referência e o limite de distância.

    2.1 Etapa de pré-processamento, com as modificações de dados brutos necessárias ao processo.

    2.2 Na etapa de cálculo de distância, existe a redução de dimensionalidade e, posteriormente, o cálculo da distância com relação à distribuição de referência. Isso é feito para cada uma das distribuições de referência na tabela de registros.

    2.3 Na etapa de verificação do desvio, existe a comparação direta de cada distância calculada anteriormente com o limite de referência para a distribuição selecionada. Caso o valor calculado para cada distribuição seja maior que todos os limites registrados, existe a identificação do desvio e necessidade de retreinamento do modelo. Caso o valor calculado para cada distribuição seja menor que todos os limites registrados aplicado o parâmetro percentual $\alpha = 0..1$, existe a identificação do desvio e necessidade de substituição do modelo. O limite de referência de cada registro funciona como o limite superior, e o parâmetro α funciona como o limite inferior para definição de limites do desvio.

2.4 Na etapa de seleção do modelo verifica-se dentre os registros qual apresenta menor diferença em relação ao limite, prosseguindo com a sua seleção. Independente da detecção do desvio no passo anterior, existe, dentro do ciclo, a seleção do modelo mais adequado. O desvio só poderá ser tratado no próximo ciclo, após as novas ocorrências esteram disponíveis para efetuarmos novo treinamento.

2.5 Na última etapa, existe o processamento do modelo selecionado como mais adequado.

3. Caso ocorra a indicação de necessidade de retreino, assim que houverem novos dados de ocorrência, antes de começar um próximo ciclo de processamento, os modelos são retreinados.

3.1 Na etapa de retreinamento, o modelo é treinado com os conjuntos de dados do último período pré-processados. Onde também existe o cálculo de distância entre a distribuição atual de dados e a distribuição referente ao último modelo usado. Caso não exista slot vazio na tabela de registros, o mais antigo é substituído.

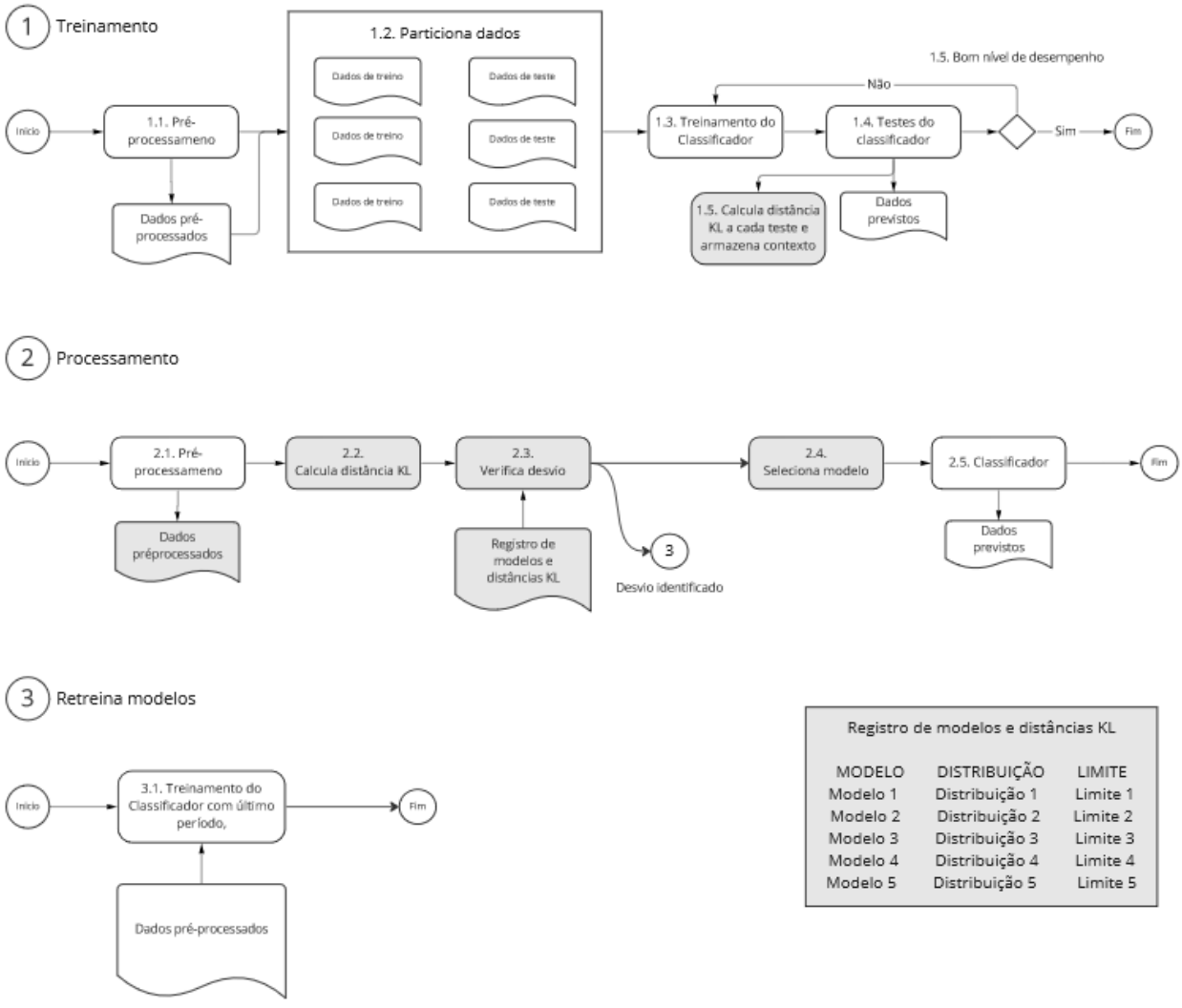


Figura V.2: Esquema de geral do método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos - Kairos.

V.3 Formalização dos principais conceitos do método

Nesta seção o método é formalizado, com a apresentação dos principais conceitos que o formam. Acompanham esta formalização o diagrama geral da sua estrutura e exemplificação do funcionamento com dados sintéticos.

Seja $J = \{i_1, i_2, \ldots, i_n\}$ o conjunto de tuplas na janela de dados, onde n é o número total de pontos. Cada ponto i_k $(1 \leq k \leq n)$ pode ser representado como um vetor de atributos:

$$i_k = (a_1^k, a_2^k, \ldots, a_m^k),$$

onde a_j^k $(1 \leq j \leq m)$ são os atributos associados ao ponto i_k e m é o número total de atributos.

Dessa forma, a janela de dados J pode ser representada como um conjunto de vetores de atributos:

$$J = \{(a_1^1, a_2^1, \ldots, a_m^1), (a_1^2, a_2^2, \ldots, a_m^2), \ldots, (a_1^n, a_2^n, \ldots, a_m^n)\},$$

onde cada vetor corresponde a um ponto i_k na janela, e a_j^k representa o j-ésimo atributo do k-ésimo ponto.

Cada janela de dados representa um período de informações, seja nas fases de treinamento e testes, como no ciclo de processamento. Distâncias são medidas entre distribuições construídas a partir de objetos em duas janelas, uma de referência J_n, e uma deslizante J_t. A janela de referência representa uma distribuição de referência, e a janela deslizante representa a delimitação de objetos dos fluxos contínuos de dados. As duas janelas têm o mesmo tamanho.

Quando em tempo de treinamento e testes, a janela deslizante é a distribuição de testes e a de referência a distribuição de treinamento. Já no ciclo de processamento de fluxo de dados, a janela deslizante é a distribuição de dados atual, e a de referência sendo cada uma das distribuições armazenadas na tabela de registros. A distância é a medida de divergência entre as duas distribuições.

Como há diversas dimensões (diversos atributos) nos dados tratados, aplica-se a redução de dimensionalidade através de Linear Discriminant Analysis (LDA) [Hastie et al., 2009]. Comparando a distância no subespaço dado pelas L médias das classes, nota-se que distâncias ortogonais a esse subespaço não adicionam novas informações, pois contribuem de igual forma para cada classe. Assim, podemos transformar um problema m-dimensional para um problema com $L-1$ dimensões ao projetar os dados ortogonalmente nesse subespaço. No nosso caso, transformamos em um subespaço unidimensional H_1 (onde $L = 2$).

Seja X_{J_n} e X_{J_t} as matrizes de dados das janelas referência e deslizante, respectivamente. Ambas

as matrizes têm dimensão $n \times m$, onde n é o número de objetos (ou pontos) nas janelas e m é o número de atributos (dimensões) originais.

$$X_{J_n} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix}, \quad X_{J_t} = \begin{bmatrix} x'_{11} & x'_{12} & \cdots & x'_{1m} \\ x'_{21} & x'_{22} & \cdots & x'_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x'_{n1} & x'_{n2} & \cdots & x'_{nm} \end{bmatrix}$$

Distribuições de saída:

Após aplicar o LDA, obtemos um subespaço unidimensional H_1. As matrizes de dados transformadas são projetadas nesse subespaço e têm dimensão $n \times 1$:

$$Y_{J_n} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad Y_{J_t} = \begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_n \end{bmatrix}$$

onde $y_i = W^T x_i$ e $y'_i = W^T x'_i$ são as projeções dos pontos de entrada no subespaço unidimensional H_1. Aqui, W é o vetor de pesos obtido a partir do LDA.

Essa notação representa as distribuições antes e depois da aplicação do LDA. Note que, ao comparar as distâncias no subespaço H_1, estamos comparando as distribuições Y_{J_n} e Y_{J_t}, sendo as versões transformadas das distribuições originais.

Como próximos passos, ambas as janelas são normalizadas, Equações V.1 e V.2, e transformadas em distribuições de probabilidade, nas Equações V.3 e V.4. Esses passos são necessários como preparação para a forma de cálculo de distância e comparação entre as distribuições. Aplicamos a função *minmax_scale* [Python Software Foundation, 2023] em cada uma das distribuições Y_{J_n} e Y_{J_t}, para redimensionar os valores de cada distribuição para o intervalo [0, 1]. As novas distribuições escaladas são denotadas por $Y_{J_n,norm}$ e $Y_{J_t,norm}$, respectivamente.

Cada elemento da distribuição é dividido pelo somatório de todos os elementos, para garantir que a soma de todos os valores seja igual a 1, o que é uma característica de distribuições de probabilidade. As novas distribuições de probabilidade serão denotadas por $Y_{J_n,prob}$ e $Y_{J_t,prob}$, respectivamente.

Portanto, temos:

$$Y_{J_n,\,\text{norm}} = \frac{Y_{J_n} - \min\left(Y_{J_n}\right)}{\max\left(Y_{J_n}\right) - \min\left(Y_{J_n}\right)} \tag{V.1}$$

$$Y_{J_t,\,\text{norm}} = \frac{Y_{J_t} - \min\left(Y_{J_t}\right)}{\max\left(Y_{J_t}\right) - \min\left(Y_{J_t}\right)} \tag{V.2}$$

$$Y_{J_n,\,\text{prob}} = \frac{Y_{J_n,\,\text{norm}}}{\sum_{i=1}^{n} Y_{J_n,\,\text{norm}\ i}} \tag{V.3}$$

$$Y_{J_t,\,\text{prob}} = \frac{Y_{J_t,\,\text{norm}}}{\sum_{i=1}^{n} Y_{J_t,\,\text{norm}\ i}} \tag{V.4}$$

Aqui, aplicamos entropia relativa ou divergência de Kullback-Leibler (KL) entre duas massas de probabilidade $Y_{J_n,prob}$ e $Y_{J_t,prob}$, para apurar sua divergência. Entendemos que a forma de análise de distribuição será a mais adequada para uma análise mais rápida, por não depender de análise de indicadores de desempenho. Em nosso modelo utilizaremos a distância ou divergência de Kullback-Leibler (KL) [Kullback, 2012], também conhecida como entropia relativa, que mede a distância entre duas distribuições de probabilidade e, portanto, pode ser usada para testar a mudança.

Uma das maneiras mais gerais de representar divergência entre duas distribuições é a entropia relativa da teoria da informação, também conhecida como divergência de Kullback-Leibler (ou KL) [Cover and Thomas, 2012]. Essencialmente, a divergência KL é a expectativa da diferença logarítmica entre a probabilidade de duas distribuições. A KL mede a quantidade de informação perdida ao se aproximar uma distribuição de probabilidade de referência (a "verdadeira" distribuição) por meio de outra distribuição de probabilidade (a distribuição aproximada). É uma medida da diferença entre as duas distribuições sendo expressa em termos de entropias, sendo medidas da incerteza associada a cada distribuição. Portanto, embora a KL possa ser interpretada como uma medida de distância em alguns contextos, é importante entender que ela é uma medida de diferença entre distribuições e não é uma medida de distância com todas as propriedades matemáticas que esperamos de uma distância. Em termos informação, neste trabalho, vamos nos referir a divergência Kullback-Leibler como distância, assim como foi discutida em diversos trabalhos como em [Amari, 2016; Lee and Seung, 1999; Cover and Thomas, 2012].

Muitas variantes foram propostas procurando resolver essa falta de simetria e natureza não métrica entre duas distribuições, preservando sua relação com a taxa de erro do classificador. Exemplos incluem a distância de Chernoff, a distância de Bhattacharyya e a divergência de Jensen-Shannon. No entanto, todas as medidas mencionadas têm outras desvantagens e mostraram nenhuma superioridade conclusiva em ambientes experimentais. A divergência KL também tem várias vantagens que torna a expressão adequada para ser usada como medida de similaridade entre duas distribuições. Como a Relação com a Razão de Verossimilhança (RV), onde a divergência de KL é frequentemente apresentada como o valor esperado da razão de verossimilhança logarítmica. Essa definição pode ser especialmente útil quando conhecemos a distribuição a partir dos dados e é fácil

calcular o log da razão entre as duas probabilidades. A Estimativa de Máxima Verossimilhança (MLE) pode ser vista como uma forma minimizada da divergência KL entre os dados da distribuição empírica e a distribuição do modelo proposto. O estimador de máxima verossimilhança é aquele que minimiza a distância KL até a distribuição verdadeira. O primeiro termo da expansão de Taylor das divergências KL é equivalente à estatística do teste do qui-quadrado. Além disso, o teste espacial de Kulldorff utilizado para agrupamento de dados poder ser estatisticamente comparada a distância KL entre a distribuição dos dados empíricos e a distribuição de Poisson. A desigualdade de Pinsker, distância de variação total entre duas densidades, chamada distância estatística, é calculada em função da divergência de KL dessas duas densidades [Basterrech and Woźniak, 2022].

Atividades referentes aos passos 1.5 e 2.2 da Figura V.2. Com isso, teremos a divergência de Kullback-Leibler entre as distribuições de probabilidade $Y_{J_n,prob}$ e $Y_{J_t,prob}$, denotado pela Equação V.5. Esse valor pode ser interpretado como a distância (em termos de informação) entre as distribuições, sendo que valores maiores indicam uma maior diferença entre elas.

$$d = D_{KL}(Y_{J_n,prob}, Y_{J_t,prob}) \tag{V.5}$$

Esta divergência será nossa medida da diferença entre as duas distribuições. A próxima etapa é determinar se essa medição é estatisticamente significativa. Formalmente, afirmamos a hipótese nula $H_0 : p(x) = q(x)$ e deseja-se determinar a probabilidade de observar o valor d se H_0 for falso.

Ao iniciar a fase de treinamento do modelo M, tomamos como referência temporária a distribuição $p(x)$, e para cada rodada de treinamento e testes, afere-se d entre a distribuição $p(x)$ e $q(x)$, valor que chamaremos de $\hat{d}$ como referência de limite para apuração da distância que será armazenado na tabela de registros. Quando se finaliza a fase de treinamento e testes, o modelo M é associado ao primeiro registro de armazenamento de modelos com o valor de referência de $\hat{d}$, associado à distribuição $p(x)$.

Para cada ciclo de processamento, seleciona-se o período atual, efetua-se o pré-processamento, e calcula-se a distância com referência a cada uma das distribuições armazenadas na tabela de registros. Esta é a etapa de verificação do desvio, onde existe a comparação direta de cada distância calculada anteriormente com o limite de referência para a distribuição selecionada. Respectivamente a atividade 2.3 na Figura V.2. Caso o valor calculado d para cada distribuição seja maior que todos os limites $\hat{d}$ registrados, existe a identificação do desvio e necessidade de treinamento de um novo modelo. Caso o valor calculado d para cada distribuição seja menor que todos os limites registrados aplicado o parâmetro percentual $d < \hat{d} \cdot \alpha$, existe a identificação do desvio e necessidade de retreinamento do modelo. O parâmetro α pode variar entre 0 e 1. De outra forma, existindo algum registro onde a distância esteja abaixo do limite, e dentro do limite já aplicado o parâmetro percentual $d > \hat{d} \cdot \alpha$ e $d < \hat{d}$, existe modelo já treinado que pode ser reutilizado.

Formalmente, o processo de verificação do desvio pode ser descrito da seguinte maneira:

- Seja d a medida de diferença entre as distribuições de probabilidade $Y_{J_n,prob}$ e $Y_{J_t,prob}$, dada por $d = D_{KL}(Y_{J_n,prob}||Y_{J_t,prob})$.
- Seja $\hat{d}$ o limite de referência de cada registro na tabela de registros.
- Se $d > \hat{d}$, então há um desvio e é necessário realizar um retreinamento do modelo.
- Se $d < \hat{d} \cdot \alpha$, então há um desvio e é necessário realizar um retreinamento do modelo.
- Se $\hat{d} \cdot \alpha \leq d < \hat{d}$, então o modelo já treinado pode ser reutilizado.

$$\begin{cases} \text{Retreinamento} & \text{, se para todas as linhas da tabela de registros,} \\ & \text{temos que } d < \hat{d}^i \cdot \alpha \text{ ou } d > \hat{d}^i \\ \text{Já existe um modelo treinado adequado} & \text{, caso contrário} \end{cases}$$

Mesmo que exista indicação de retreinamento, o modelo só é retreinado no próximo ciclo de dados, então, para o ciclo atual, precisamos selecionar o modelo mais próximo, mesmo que todos tenham ficado fora dos limites. Na etapa de seleção do modelo verifica-se dentre os registros qual apresenta menor diferença em relação ao limite e a distância calculada entre a janela atual e a armazenada em cada registro, prosseguindo com a sua seleção. Representa a atividade 2.4 na Figura V.2, e segue na Equação V.6.

$$M_{selecionado} = \arg\min_i D_i(J_t, J_i) \tag{V.6}$$

Onde $D_i(J_t, J_i)$ é a distância calculada entre a janela atual J_t e a janela armazenada J_i associada ao modelo M_i, e $\arg\min_i D_i(J_t, J_i)$ é o índice do modelo com a menor distância em relação ao limite e a distância calculada. O modelo selecionado $M_{selecionado}$ será utilizado para a predição no ciclo atual.

Finalmente, como último passo, se foi identificado anteriormente que existe necessidade de retreino, a partir dessa nova janela de dados, é gerado um novo modelo treinado M_{novo}. A distribuição atual de dados J_t é armazenada juntamente com a distância calculada entre o fluxo atual e a distribuição referente ao último modelo usado. Além disso, o novo modelo treinado M_{novo} é adicionado à tabela de registros. Se a tabela de registros atingir seu tamanho máximo, o registro mais antigo é descartado e o novo é adicionado em seu lugar.

Tabela V.1: Notação matemática utilizada no método.

Notação	**Descrição**
J	Conjunto de pontos na janela de dados
i_k	k-ésimo ponto na janela de dados
a_j^k	j-ésimo atributo do k-ésimo ponto na janela de dados
m	Número total de atributos
n	Número total de pontos na janela de dados
X_{J_n}	Matriz de dados da janela de referência
X_{J_t}	Matriz de dados da janela deslizante
W	Vetor de pesos obtido a partir do LDA
Y_{J_n}	Distribuição transformada da janela de referência
Y_{J_t}	Distribuição transformada da janela deslizante
H_1	Subespaço unidimensional obtido a partir do LDA
$Y_{J_n,norm}$	Distribuição de probabilidade normalizada da janela de referência
$Y_{J_t,norm}$	Distribuição de probabilidade normalizada da janela deslizante
$Y_{J_n,prob}$	Distribuição de probabilidade da janela de referência
$Y_{J_t,prob}$	Distribuição de probabilidade da janela deslizante
d	Distância de Kullback-Leibler entre as distribuições $Y_{J_n,prob}$ e $Y_{J_t,prob}$
$\hat{d}$	Valor de referência para a distância d
α	Parâmetro percentual para apuração do desvio

V.4 Algoritmos

Abaixo, seguem os algoritmos referentes aos passos acima. No Algoritmo 1, temos a primeira parte que representa o treinamento e testes do modelo de aprendizagem de máquina. Durante o processo, existe o cálculo da distância KL entre as distribuições utilizadas. Ao final, armazena-se uma distribuição de referência e a maior distância KL registrada. Neste algoritmo, existe a utilização de seis funções: `Preprocessamento`, `CalculaDivergencia`, `EfetuaTreinamento`, `EfetuaTestes`, `VerificaDesempenhoModelo` e `ArmazenaRegistroTabela`.

A função `PreProcessamento` representa aqui apenas a fase de pré-processamento que será utilizada e não faz parte do modelo. Sendo definida conforme os dados e de cada necessidade. A função `CalculaDivergencia` do Algoritmo 2, expressa o cálculo da distância KL entre duas distribuições. As funções `EfetuaTreinamento`, `EfetuaTestes` são apenas uma representação do treinamento e testes do classificador. A função `VerificaDesempenhoModelo` verifica o desempenho do modelo treinado e testado. E a função `ArmazenaRegistroTabela` armazena os resultados do modelo treinado e testado em uma tabela, com suas distribuições e limite de distância calculado.

Segue a função `CalculaDivergencia` no Algoritmo 2, que expressa o cálculo da distância KL entre duas distribuições. Algumas funções importantes chamadas neste algoritmo são: `LDA` que realiza a análise de discriminante linear (LDA) nos dados de entrada; `Normaliza` função que normaliza os dados de entrada para ter média zero e desvio padrão igual a um; e, `Transforma em probabilidade`, função que transforma os dados de entrada em uma distribuição de probabilidade.

Algoritmo 1 Treinamento e teste do modelo

```
1: enquanto Ciclo de treinamento e testes faça
2:     PreProcessamento(dadosdetreinamento)
3:     Carrega janelatreinamento com distribuição de comparação.
4:     Normaliza janelatreinamento.
5:     modelo ← EfetuaTreinamento(janelatreinamento)
6:     PreProcessamento(dadosdetestes)
7:     Carrega janelatestes com distribuição de comparação.
8:     Normaliza janelatestes.
9:     EfetuaTestes(modelo,janelatestes)
10:    desempenhoModelo ← VerificaDesempenhoModelo(modelo)
11:    distancia ← CalculaDivergencia(janelatestes,janelatreinamento)
12:    se desempenhoModelo >= limiteAceitacao então
13:        AbandonaLoop.
14:    fim se
15: fim enquanto
16: ArmazenaRegistroTabela(Modelo, distancia, janelatestes)
```

Algoritmo 2 Calcula a divergência KL

```
1: função CalculaDivergencia(P, Q)
2:     LDA(P)
3:     LDA(Q)
4:     Normaliza(P)
5:     Normaliza(Q)
6:     Transforma em distribuição de probabilidade(P)
7:     Transforma em distribuição de probabilidade(Q)
8:     devolve ∫ p(x) log ( p(x) / q(x) ) dx
9: fim função
```

O Algoritmo 3 representa o processamento do classificador após o treinamento, onde os dados entrantes são fluxos de dados constantes. Primeiro existe a seleção de nova janela de dados e o seu pré-processamento. Então, começa-se a buscar o melhor modelo na tabela pré-existente de distribuições e modelos já treinados. Para cada distribuição de referência encontrada na tabela, calcula-se a distância KL, referente à nova janela de entrada. Caso a distância calculada esteja no limite registrado na tabela, contando com o parâmetro α, seleciona-se o classificador já treinado. Caso, até o final da tabela de registro não se encontre classificador aderente, no limite de distância, efetua-se novo treinamento do classificador, com a distribuição de entrada, nova janela de dados. Com a necessidade de retreinamento, registra-se também a ocorrência de um desvio de conceito. A função `ProcessaModeloML` aqui representa o processamento do classificador selecionado. A função `VerificaDesvioModelo` é chamada para verificar se o desvio entre os novos dados e a distribuição de comparação é significativo. A função é melhor detalhada mais abaixo no Algoritmo 4. O resultado dessa função é um booleano desvio e o modelo mais adequado, utilizado na função `ProcessaModeloML`. Caso seja identificado um desvio significativo, o algoritmo registra esse desvio, efetua o treinamento do modelo com os novos dados e calcula a divergência entre a última janela de dados utilizada pelo modelo antigo e a nova janela de dados utilizada pelo modelo atualizado. Por fim, é armazenado o registro da tabela com o novo modelo, a distância calculada e a nova janela de dados.

Algoritmo 3 Processamento do classificador

```
1:  enquanto Fluxo contínuo de dados faça
2:      Recebe novosdados
3:      PreProcessamento(novosdados)
4:      Carrega novajanela com distribuição de comparação.
5:      Normaliza novajanela.
         desvio, modeloMaisAdequado = VerificaDesvioModelo(novosdados)
         ProcessaModeloML(novajanela, modeloMaisAdequado)
6:      se desvio então
7:          Registra desvio
8:          novoModelo ← EfetuaTreinamento(novosdados)
9:          distancia ← CalculaDivergencia(janela(modeloMaisAdequado), novajanela)
10:         ArmazenaRegistroTabela( novoModelo, distancia, novajanela)
11:     fim se
12: fim enquanto
```

O Algoritmo 4 detalha função `VerificaDesvioModelo` que implementa a detecção do desvio e seleção do melhor modelo, baseado na distância KL (Kullback-Leibler). Esta função recebe uma nova janela de dados *janelanova* como entrada e faz a comparação com as distribuições de referência armazenadas na tabela de registros. O algoritmo utiliza a distância KL para calcular a divergência entre a nova janela de dados e as janelas de referência na tabela. A distância KL é uma medida de divergência entre duas distribuições de probabilidade e pode ser utilizada para medir

a diferença entre a distribuição de referência e a distribuição dos novos dados. Caso a distância entre a nova janela de dados e a janela de referência seja menor do que a distância de referência multiplicada por um fator α, o algoritmo registra que não houve desvio (desvio = False). Caso contrário, o algoritmo registra que houve desvio (desvio = True). O algoritmo também calcula a diferença entre a distância de referência e a distância entre a nova janela de dados e a janela de referência $difDistancia$. Essa diferença é utilizada como um critério adicional para selecionar o melhor modelo, caso haja desvio. O modelo com a menor distância pode ser selecionado e armazenado na variável $modeloComMenorDistancia$.

Algoritmo 4 Detecção do desvio e seleção do melhor modelo

```
1:  função VerificaDesvioModelo(janelanova)
2:      enquanto Tabela com distribuições e distâncias KL de referência faça
3:          Carrega janelareferencia com distribuição de comparação.
4:          Carrega distanciareferencia de distribuição de referência.
5:          distancia ← CalculaDivergencia(janelanova,janelareferencia)
6:          se (distancia ≤ distanciareferencia) and (distancia ≥ distanciareferencia * α) então
7:              Registra desvio = False
8:          senão
9:              Registra desvio = True
10:         fim se
11:     fim enquanto
12:     Armazena modeloComMenorDistancia
13:     devolve desvio, modeloComMenorDistancia
14: fim função
```

V.5 Exemplo passo a passo

Colocamos um exemplo a seguir com dados de voos extraídos da ANAC com uma simplificação do que seria a execução do fluxo definido anteriormente. A Figura V.3 mostra o resultado ao final das atividades de treinamento e testes do primeiro modelo gerado na etapa inicial de operação do Kairos. Na parte inferior esquerda da figura há a representação da distribuição do primeiro mês de dados, após a aplicação de LDA. O primeiro mês corresponde à distribuição usada para o treinamento. Na parte inferior direita da figura há a distribuição do segundo mês, após aplicação de LDA, e corresponde aos dados de teste. A parte superior direita da figura mostra a distância KL calculada entre as duas distribuições. Com isso, o primeiro registro da tabela de registros (parte superior direita da figura) recebe o modelo M treinado, a última distribuição de referência $p(x)$, e o valor do limite KL.

Na etapa cíclica de processamento afere-se sempre a distância entre a janela referência $q(x)$ e a janela selecionada $p(x)$, apurando d. A janela referência corresponde a cada uma das distribuições armazenadas na tabela de registros. Compara-se o valor de d com $\hat{d}$ para cada registro, cada

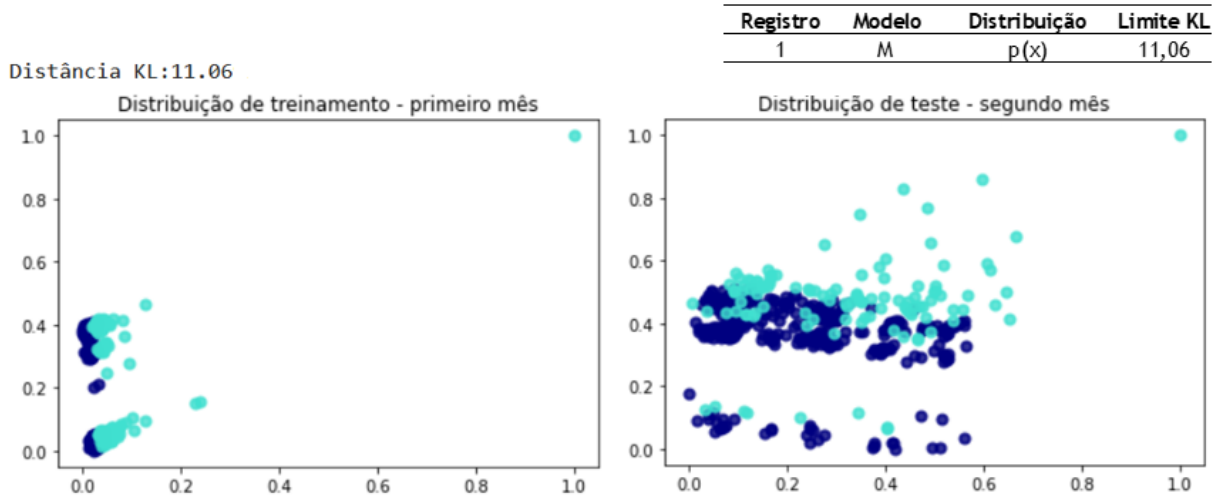


Figura V.3: Maior diferença KL apurada na fase de treinamento e testes. Primeiro registro na tabela de registros.

distribuição da tabela, e verifica-se onde $d \leq \hat{d}$ e qual a menor distância. Existe uma varredura da tabela de registros de modelos já treinados. Para o exemplo abaixo, consideramos o parâmetro $\alpha = 0$, ou seja, será considerado o valor cheio de $\hat{d}$ para identificação para seleção do modelo e identificação de desvio.

A Figura V.4 mostra um exemplo de ciclo de processamento onde é calculada a diferença entre a distribuição de referência armazenada na tabela e a nova distribuição sendo processada. No exemplo, a distância d foi de $7,88$, menor que o valor limite $\hat{d}$ do modelo M. Dessa forma, não se detecta desvio, e o modelo é mantido. A figura mostra como a tabela de registros manteve-se inalterada.

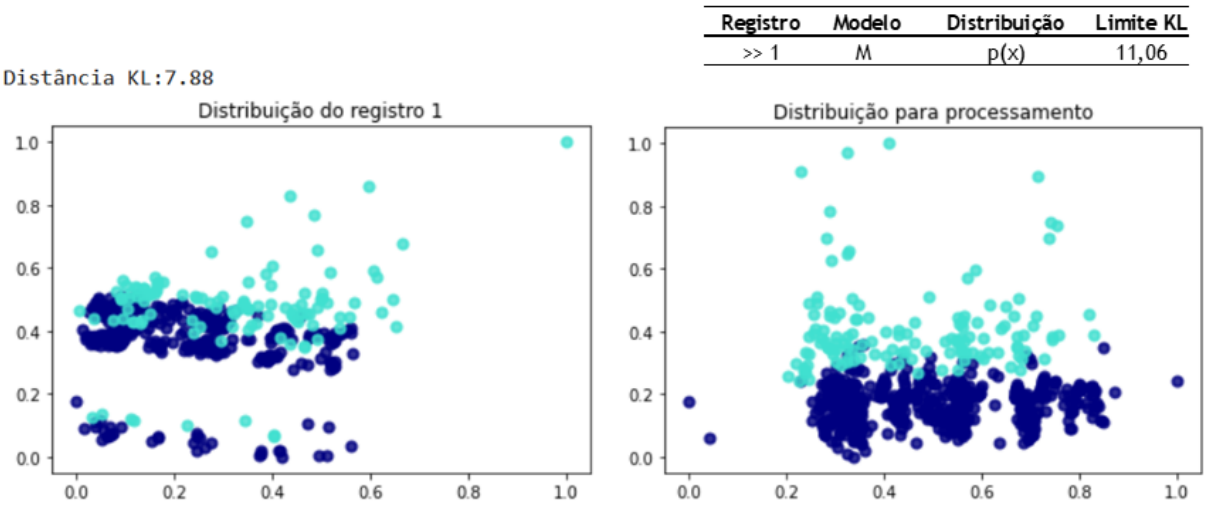


Figura V.4: Diferença KL apurada no primeiro ciclo de processamento. Menor que o limite.

A Figura V.5 mostra exemplo de novo ciclo de processamento onde se detecta um desvio de conceito. No exemplo é calculada a diferença entre a distribuição de referência armazenada na

tabela e a nova distribuição sendo processada. No exemplo, a distância d foi de $14,99$, maior que o valor limite $\hat{d}$ do modelo M, de $11,06$. Dessa forma, é detectado o desvio, e um novo modelo será treinado. O novo modelo M', função da nova distribuição $p(x)'$ e o valor $\hat{d}$ entram como novo registro na tabela de referência, conforme mostra a parte superior direita da figura.

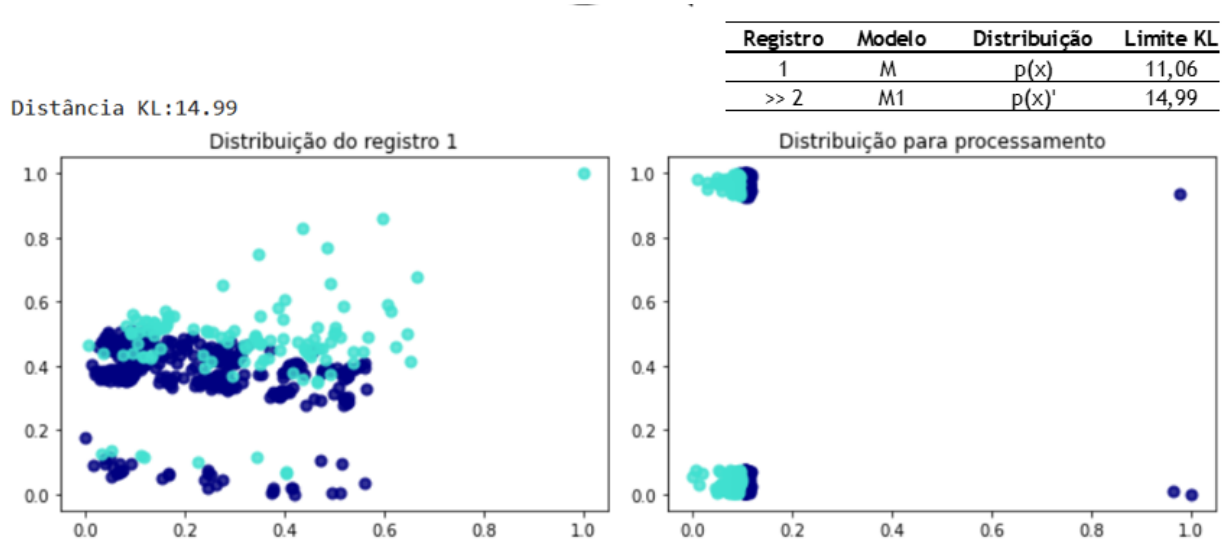


Figura V.5: Diferença KL apurada em novo ciclo de processamento. Ultrapassou o limite. Tabela mantida, sem detecção do desvio.

A Figura V.6 mostra exemplo de novo ciclo de processamento onde se detecta um desvio de conceito. Neste ciclo, já se utiliza o modelo M', a distribuição $p(x)'$ e o valor $\hat{d}$ (segunda linha da tabela, correspondendo ao modelo atualmente selecionado para uso em classificações de voos). A figura mostra um exemplo onde é calculada a diferença entre a distribuição de referência armazenada na tabela e a nova distribuição sendo processada. No exemplo, a distância d foi de $17,06$, maior que o valor limite $\hat{d}$ do modelo M', de $14,99$. Dessa forma, é detectado o desvio. Os outros registros armazenados e já treinados da tabela são testados. No presente exemplo, o primeiro registro é selecionado e a diferença KL calculada entre as distribuições.

A Figura V.7 mostra o resultado da comparação da nova distribuição com a do primeiro registro já armazenado. No exemplo, a distância d foi de $10,98$, menor que o valor limite $\hat{d}$ do modelo M, sendo de $11,06$. Dessa forma, o modelo M é selecionado para o processamento corrente. A tabela de registros foi mantida a mesma, mas o registro corrente mudou, voltou a ser o primeiro. Note que, primeiro, o desvio só é detectado se nenhum modelo já registrado for aderente e, segundo, que o teste será feito com cada registro existente da tabela, selecionando o mais aderente. Na etapa de seleção do modelo, verifica-se dentre os registros qual apresenta menor diferença em relação ao limite e a distância calculada entre a janela atual e a armazenada em cada registro, prosseguindo com a sua seleção.

A verificação é repetida a cada ciclo de processamento, com a seleção do registro mais adequado

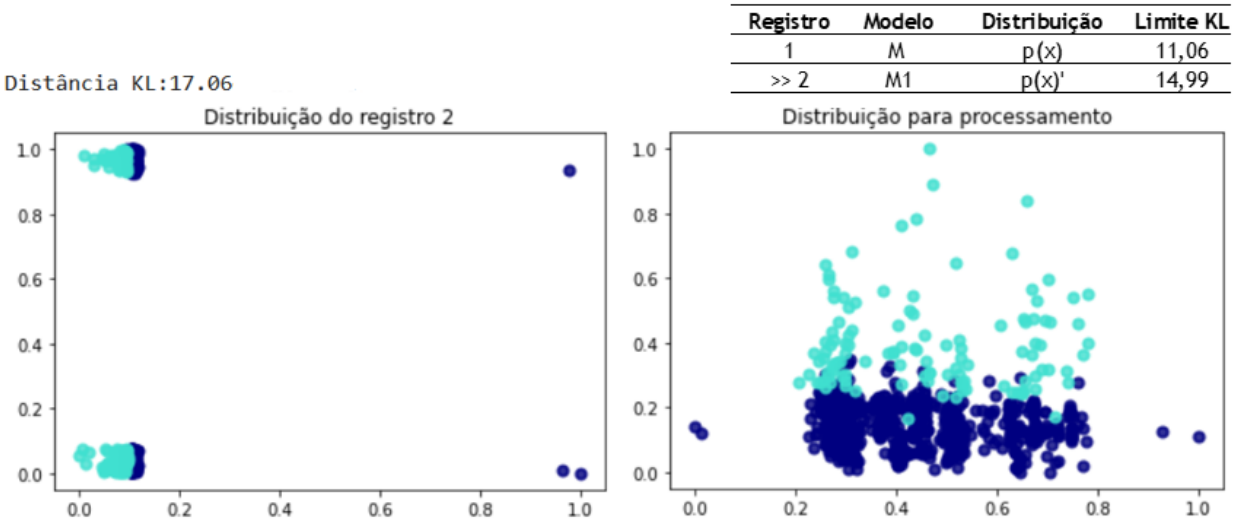


Figura V.6: Diferença KL apurada em novo ciclo de processamento. Ultrapassou o limite. Tabela de registros, após desvio detectado e modelo retreinado.

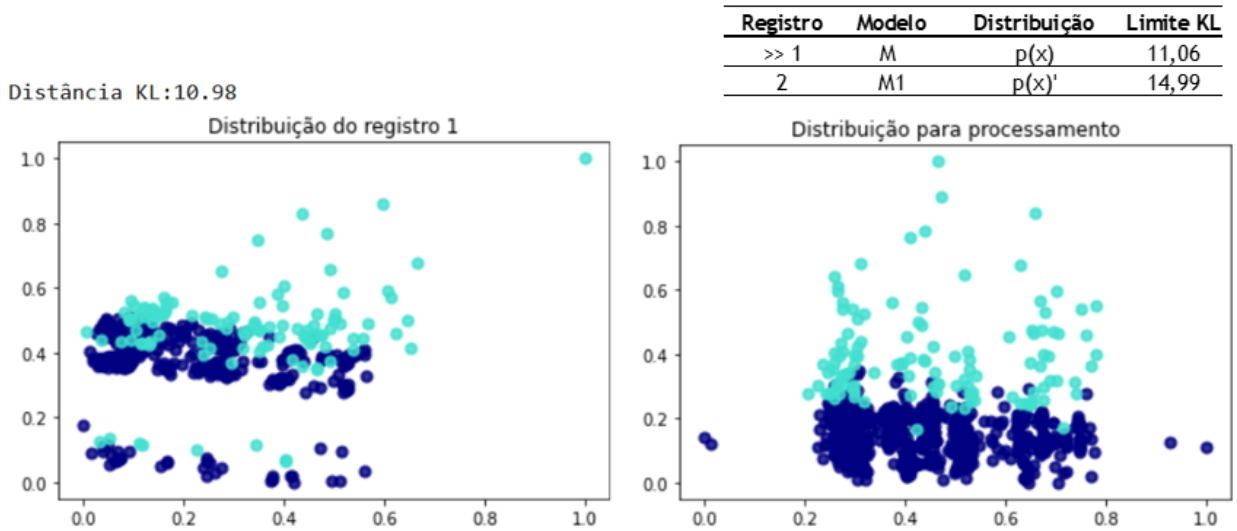


Figura V.7: Com o desvio, o primeiro modelo foi testado e selecionado novamente. Distância KL estava no limite. Tabela de registros se manteve, pois o modelo não foi retreinado.

e modelo já treinado. Caso se apure que nenhum dos modelos já verificados é adequado, um novo modelo é treinado e criado um novo registro. Isso faz com que o Kairos fique mais robusto com o passar do tempo e trate mais contextos de dados sem necessitar de retreinamento do modelo. Se a tabela de registros atingir seu tamanho máximo, o registro mais antigo é descartado e o novo é adicionado em seu lugar.

V.6 Implementação

Para o processamento e tratamento dos dados foi utilizado o Google Colab [Google, 2023], em ambiente básico de CPU e memória. A parte de pré-processamento foi desenvolvido em *R*

[R Core Team, 2023], e a parte do método e resultados em *Python* [Python Software Foundation, 2023], todos disponíveis no GitHub [Carvalho, 2023]. Para o pré-processamento, em linguagem *R*, as principais bibliotecas usadas foram: *doParallel*, *magrittr*, *dplyr*, *readxl*, *sqldf*, *psych*, *ggplot*2 e *ggpubr*. Para a parte de geração de modelos e processamento, em *Python*, as principais bibliotecas utilizadas foram: *sklearn.discriminant_analysis*, *sklearn.metrics*, *sklearn.model_selection*, *scipy.stats*, *torch*, *sklearn.naive_bayes*, *sklearn.ensemble* e $skmultiflow.drift_detection$.

Segue indicação de principais bibliotecas utilizadas por parte do processo:

- Cálculo da distância entre distribuições: *sklearn.preprocessing*, *sklearn.discriminant_analysis* e *math*.
- Estruturação e processamento dos modelos de aprendizado de máquina: $sklearn.ensemble.RandomForestClassifier$, $sklearn.naive_bayes.GaussianNB$, *sklearn.preprocessing*, e *sklearn.model_selection*.
- Métricas: *sklearn.metrics*.
- Estruturação e execução dos métodos ADWIN, DDM, HDDM e Page Hinkley: $skmultiflow.drift_detection$.

Capítulo VI Experimentos

Este capítulo contém a estruturação e o passo a passo de como foram feitos os experimentos do método proposto, apresentando quais são as fontes de dados e como foram processadas; como foi montado o método proposto; como foram montados os métodos de comparação; e, como foram feitas as comparações.

VI.1 Pré-processamento

Nesta subseção, descrevemos quais fontes de dados foram selecionadas e coletadas, assim como cada passo do pré-processamento efetuado para dar origem às bases de dados usadas pelos modelos de aprendizado de máquina. Os dados e grande parte do pré-processamento partem do trabalho prévio de Teixeira et al. [2020]. Esse trabalho fornece informações sobre o Brazilian Flight Dataset (BFD). A descrição inclui as fontes de dados de voos regulares (VRA) e dados meteorológicos. O VRA é fornecido pela ANAC (Agência Nacional de Aviação Civil) e contém todos os voos brasileiros. Os dados meteorológicos são fornecidos pela ASOS (Automated Surface Observing Systems). A ASOS é gerida pela Iowa State University (EUA). Ele contém dados meteorológicos coletados por sensores instalados em aeroportos de todo o mundo. O artigo traz informações dos anos 2000 até 2019, dois quais extraímos 3 anos, 2016, 2017 e 2018, para efetuar o processamento dos modelos e testar os métodos de identificação e tratamento de desvio de conceito.

Havia diferenças no caminho, formato e conteúdos dos arquivos da ANAC desde o trabalho prévio de Teixeira et al. [2020], além de inconsistências verificadas nos arquivos BFD gerados a partir de 2013. Informações como temperatura, humidade, pressão e velocidade do vento estavam repetidas em cada tupla. Dessa forma, novos arquivos fonte foram extraídos nas fontes e os scripts alterados para o pré-processamento. Os scripts foram mantidos em linguagem R, conforme fonte original. Todo o material base e descrições constam na página do Data Analytics Lab (DAL) [Ogasawara, 2023].

A Figura VI.1 mostra o fluxo de etapas do pré-processamento realizado para cada ano, e que será detalhado em seguida.

1. Coleta de arquivos de voos no site da ANAC.

   O primeiro passo é a coleta dos arquivos de voos que se encontra no site da ANAC. São os arquivos denominados VRA - Voo Regular Ativo, disponibilizados em pastas por ano e

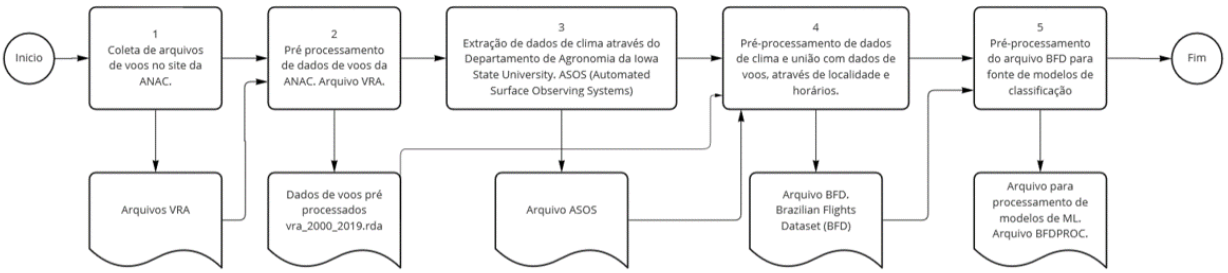


Figura VI.1: Esquema do fluxo do pré-processamento realizado.

mês. Os caminhos e formatos dos arquivos foram alterados em relação ao trabalho prévio de Teixeira et al. [2020]. Os mesmos podem ser encontrados agora em [ANAC, 2023].

2. Pré-processamento de dados de voos da ANAC.

   O passo 2 efetua o pré-processamento dos arquivos de voos da ANAC para o formato esperado pelo projeto BFD. A lógica de processamento do trabalho prévio de Teixeira et al. [2020], feita em R, foi alterada para suportar o novo formato do arquivo. Esta lógica efetua a transformação para o formato esperado para junção com dados de clima. A lógica também foi alterada para processar cada ano ao invés de todo o conjunto original.

3. Extração de dados de clima através do Departamento de Agronomia da Iowa State University ASOS (Automated Surface Observing Systems).

   O Departamento de Agronomia da Iowa State University nos Estados Unidos compila diariamente não só informações do sistema ASOS dos EUA, mas também de várias entidades ligadas à aviação civil e militar de todo o planeta. Esta universidade disponibiliza estes dados gratuitamente para download através do site da universidade em [IEM, 2023].

4. Pré-processamento de dados de clima e união com dados de voos, por localidade e horários.

   Este passo efetua a junção entre dados dos voos e de clima dos passos anteriores. Ele traz dados de clima tanto da partida como da chegada para cada voo.

5. Pré-processamento do arquivo BFD para fonte de modelos de classificação.

   O último passo já parte do que seria o arquivo final BFD, que seria o arquivo publicado no trabalho prévio de Teixeira et al. [2020], não fossem as diferenças encontradas. A partir do arquivo BFD, existe a transformação para o formato que utilizaremos para os experimentos.

   Colunas do arquivo de saída do último passo:

   - week - Número da semana em ordem, começando em semana 1.
   - expected_depart_date - data esperada da partida no formato AAAAMMDD.
   - expected_depart_hour - hora esperada da partida no formato 24h.

- expected_duration - duração esperada do voo.
- departure_delay - atraso na partida, sendo a diferença em minutos entre a partida real e a partida esperada.
- depart_temperature - temperatura na partida em graus celsius.
- depart_dew_point - ponto de condensação da água na partida.
- depart_humidity - umidade na partida.
- depart_pressure - pressão atmosférica na partida.
- depart_visibility - visibilidade na partida.
- depart_wind_speed - velocidade do vento na partida.
- XXXX_ORIG - indicação de partida ocorrida no aeroporto XXXX. Valor um marcando ocorrência e zero não ocorrência. Há um campo como este para cada um dos dez principais aeroportos do Brasil:
  - SBBR: Brasília, no Distrito Federal.
  - SBCF: Confins, em Minas Gerais.
  - SBCT: São José dos Pinhais, no Paraná.
  - SBGL: Galeão, no Rio de Janeiro.
  - SBGR: Guarulhos, em São Paulo.
  - SBKP: Viracopos, em São Paulo.
  - SBPA: Salgado Filho, no Rio Grande do Sul.
  - SBRJ: Santos Dumont, no Rio de Janeiro.
  - SBSP: Congonhas, em São Paulo.
  - SBSV: Dep. Luis Eduardo Magalhães, na Bahia.
- expected_arrival_date - data esperada de chegada no formato AAAMMDD.
- expected_arrival_hour - hora esperada de chegada.
- real_arrival_date - data real de chegada.
- real_arrival_hour - hora real de chegada.
- real_duration - duração real em minutos.
- arrival_delay - atraso na chegada, sendo a diferença em minutos entre a chegada real e a chegada esperada.
- arrival_temperature - temperatura na chegada em graus celsius.
- arrival_dew_point - ponto de condensação da água na chegada.

- arrival_humidity - umidade na chegada.
- arrival_pressure - pressão atmosférica na chegada.
- arrival_visibility - visibilidade na chegada.
- arrival_wind_speed - velocidade do vento na chegada.
- XXXX_DEST - indicação de chegada ocorrida no aeroporto XXXX. Valor um marcando ocorrência e zero não ocorrência. Há um campo como este para cada um dos dez aeroportos listados na descrição do campo XXX_ORIG.
- AZU - indicação da companhia aérea Azul. Valor um marcando ocorrência e zero não ocorrência.
- GLO - indicação da companhia aérea Gol. Valor um marcando ocorrência e zero não ocorrência.
- TAM - indicação da companhia aérea TAM. Valor um marcando ocorrência e zero não ocorrência.
- expected_year - ano no formato AAAA.
- expected_month - mês no formato MM.
- departure_delay_bin15 - indicação de atraso na partida maior do que quinze minutos. Aferido pelo campo departure_delay. Valor um marcando ocorrência e zero não ocorrência.
- arrival_delay_bin15 - indicação de atraso na chegada maior do que 15 minutos. Aferido pelo campo arrival_delay. Valor um marcando ocorrência e zero não ocorrência.

VI.2 Bases de dados para os experimentos

A partir dos arquivos gerados foram selecionadas três visões dos dados para diferentes processamentos através dos modelos, para apurar possíveis diferenças nos resultados e comportamentos tanto dos modelos de aprendizado de máquina quanto dos métodos de detecção de desvio de conceito. Na visão de malha aérea, foram selecionados todos os voos dos dez principais aeroportos no Brasil, onde eles eram origens ou destino desses voos. Na visão de aeroportos, cada base de dados era composta apenas de voos com origem ou destino em determinado aeroporto. Por fim, na visão de companhias, cada base de dados era composta apenas de voos operados por determinada companhia.

VI.3 Configuração e estrutura dos experimentos

Esta seção apresenta a estrutura utilizada para os experimentos, tanto para o método Kairos proposto, como para os demais métodos comparados. Além das visões de bases de dados elencadas na seção VI.2, os experimentos foram também feitos em duas diferentes escalas de tempo para avaliação de desvios, semanal e mensal. Além disso, foi criado um parâmetro para os experimentos com todos os métodos. Esse parâmetro permite selecionar o período para avaliação de desvios.

Foram desenvolvidos três diferentes modelos de aprendizado de máquina para classificação de atrasos em voos: Random Forest, Rede Neural e Naive Bayes. Os três modelos utilizam as mesmas bases de dados e são os mesmos utilizados em todas as formas de análise: sem retreino, método passivo e métodos ativos.

Para embasar o direcionamento dos experimentos, assim como para habilitar as comparações, foram selecionados alguns métodos de detecção de desvio. Como forma de continuidade dos experimentos prévios detalhados no Apêndice A, selecionamos o método ativo que efetua a detecção de desvio pela média de diferença entre distribuições. Mantivemos também como base o método passivo de retreinamento a cada período, assim como a linha de base, sem detecção de desvio, que seria o não retreino dos modelos de classificação em qualquer período. Como forma de apuração do funcionamento dos métodos do estado da arte e de seu comportamento nos cenários de classificação de atrasos em voos, selecionamos quatro métodos ativos de detecção de desvio, sendo um método de cada quadrante em nossa taxonomia apresentada no Capítulo III: ADWIN, DDM, HDDM e Page Hinkley. Segue o resumo dos métodos e formas para experimentos:

1. Execução dos modelos de classificação sem retreinamento. Também chamado em nosso trabalho de Linear, corresponde à execução da classificação, através dos três modelos, a cada período determinado, semanal ou mensal.

2. Execução do método passivo, que retreina os modelos de classificação a cada período, mesmo sem a identificação de desvio.

3. Execução do método ativo de detecção de desvio pela média, conforme trabalho prévio publicado e detalhado no Apêndice A.

4. Execução do método ativo ADWIN.

5. Execução do método ativo DDM.

6. Execução do método ativo HDDM.

7. Execução do método ativo Page Hinkley.

A estrutura selecionada teve como objetivos:

1. Manter a base dos experimentos prévios, com a linha de base sem retreino, método passivo de retreino a cada período, e o método por média.
2. Estender experimentos com bases de dados em diferentes níveis qualitativos, notadamente visões por malha aérea, companhias e aeroportos.
3. Estender experimentos em diferentes níveis temporais, no caso, semanal e mensal.
4. Verificar o comportamento dos métodos do estado da arte, e suas características, nos mesmos cenários.

Avaliadas características e cenários, foi desenvolvido o método Kairos, já se utilizando de obervações realizadas sobre os resultados. Sempre utilizando as mesmas bases de dados, períodos e modelos. A partir da análise do cenário anterior e do comportamento dos métodos do estado da arte, foi possível direcionar a melhor forma de proposição do novo método. A seguir, faremos uma explanação de toda a estrutura, que não segue uma ordem cronológica, mas representa todos os passos de processamento e comparações.

Apresentamos na Figura VI.2 o mesmo esquema já proposto no Capítulo V, que aqui será melhor detalhado em termos de como foram submetidos os experimentos. Foram treinados e testados os três modelos: Random Forest, Rede Neural e Naive Bayes.

1. Método Kairos - Treinamento e testes.
    1.1 Etapa de pré-processamento, com a modificação de dados brutos necessárias ao processo. O pré-processamento foi efetuado anualmente para o período de três anos, 2016, 2017 e 2018.
    1.2 Na etapa de segregação de dados é onde ocorre a divisão dos dados brutos em dados de treino e testes. Nesta etapa, foram selecionados os dois primeiros meses do ano, o primeiro para treinamento e o segundo para testes.
    1.3 Na etapa de treinamento, o modelo é treinado com os conjuntos de dados do primeiro mês.
    1.4 Na etapa de testes, o modelo é avaliado com os conjuntos de dados do segundo mês. No Kairos, nesta etapa, existe o cálculo de distância entre os dados de teste e treinamento, sendo armazenada na tabela de registro a distribuição de teste, com janela de comparação, assim como a distância KL como limite.

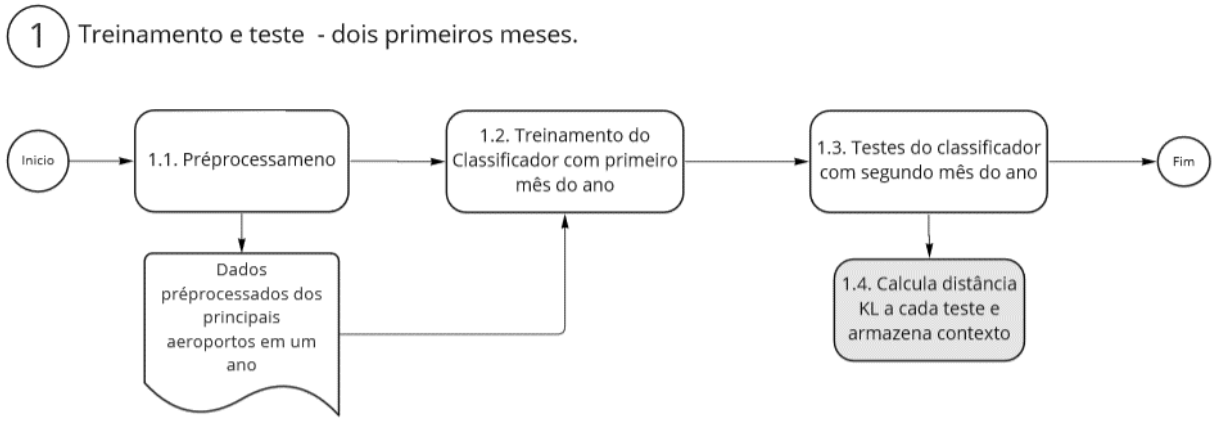


Figura VI.2: Treinamento e teste do método Kairos.

2. Ciclos de processamento do modelo, conforme Figura VI.3, com a classificação de dados. Nesta fase, já temos um registro na tabela de referência de modelos e distâncias. O primeiro registro contém os três modelos treinados, a distribuição de referência (janela fixa) e o limite de distância.

   2.1 Seleciona o período para processamento, sempre com a janela de um mês, com incrementos mensais ou semanais diante da configuração selecionada. Ou seja, a janela de comparação armazenada e a janela de processamento sempre são de um mês, mas o incremento de novos dados pode ser feito a cada semana, ou a cada mês. Os experimentos foram feitos das duas formas.

   2.2 Na etapa de cálculo de distância, existe a normalização da distribuição de entrada, com janela de mesmo tamanho da janela fixa. Posteriormente, é calculada a distância com relação à distribuição de referência selecionada. No final deste passo, teremos a distância calculada entre as distribuições. O cálculo é feito para cada registro existente na tabela. Caso seja o primeiro processamento, existe apenas um modelo na tabela.

   2.3 Na etapa de verificação do desvio, ocorre a comparação direta com o limite de referência para a distribuição selecionada. O cálculo foi feito com cada registro e verifica-se a menor diferença dentre os casos em que a distância calculada ficou no limite anteriormente registrado para aquela distribuição. Caso o valor ultrapasse o limite em todos os registros, existe a identificação de desvio.

   2.4 Na etapa de seleção do modelo, o Kairos prossegue com o modelo mais adequado selecionado na tabela de referência de modelos e distâncias, diante da verificação anterior, onde seleciona-se aquele com menor diferença entre as distribuições. A representação reduzida da tabela segue na Figura VI.5.

2.5 Nesta etapa, existe o processamento dos modelos selecionados diante da escolha na tabela de registros.

2.6 A cada passo, existe a comparação com os resultados reais, sendo registrados: tempo de processamento, indicação se houve desvio, indicação se houve retreino, acurácia, recall, precision e F1.

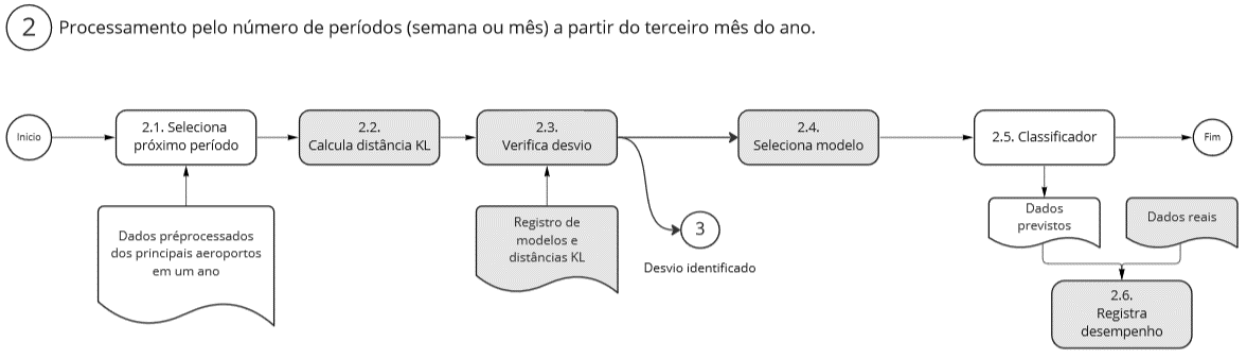


Figura VI.3: Processamento do Kairos a partir do terceiro mês.

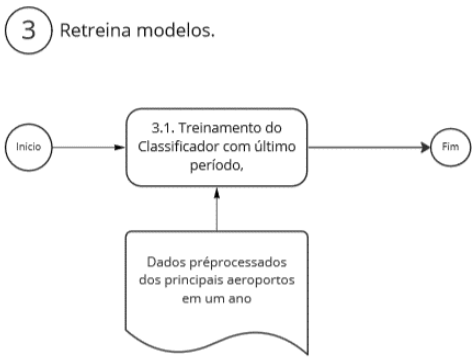


Figura VI.4: Retreina os modelos no Kairos.

Registro de modelos e distâncias KL

MODELO	DISTRIBUIÇÃO	LIMITE
Modelo 1	Distribuição 1	Limite 1
Modelo 2	Distribuição 2	Limite 2
Modelo 3	Distribuição 3	Limite 3
Modelo 4	Distribuição 4	Limite 4
Modelo 5	Distribuição 5	Limite 5

Figura VI.5: Registro de modelos treinados, janelas de comparação e distâncias limite.

3. Retreino de modelos no Kairos. Pode ser visto na Figura VI.4, que ocorre quando identificado desvio no fluxo de processamento.

3.1 Seleciona o último período utilizado para processamento, para o qual foi identificado o desvio, e retreina os modelos.

Conforme explicado no começo desta seção, mantivemos também como base o método passivo de retreinamento a cada período, assim como a linha de base, sem detecção de desvio, que

seria o não retreino dos modelos em qualquer período. Por serem mais simples, e facilmente derivados pelos fluxos apresentados, não representamos graficamente os fluxos do processamento sem identificação de desvio, ou o fluxo de identificação passiva, com retreino a cada período. Mas, para ambos, a cada período, existe a comparação com os resultados reais, sendo registrados: tempo de processamento, indicação se houve desvio, indicação se houve retreino, acurácia, recall, precision e F1.

Em seguida, detalhamos a forma dos experimentos com os métodos do estado da arte nos cenários de classificação de atrasos em voos, onde foram selecionados quatro métodos ativos de detecção de desvio: ADWIN, DDM, HDDM, Page Hinkley. O mesmo fluxo foi usado para o processamento do método ativo efetuado pela média. A Figura VI.6 mostra o fluxo de treinamento e teste dos modelos.

4. Treinamento e testes em cada um dos demais métodos.

   4.1 Etapa de pré-processamento, com a modificação de dados brutos necessárias ao processo. O pré-processamento foi efetuado anualmente para o período de três anos, 2016, 2017 e 2018.

   4.2 Na etapa de treinamento, o modelo é treinado com os conjuntos de dados do primeiro mês.

   4.3 Na etapa de testes, o modelo é testado com os conjuntos de dados do segundo mês.

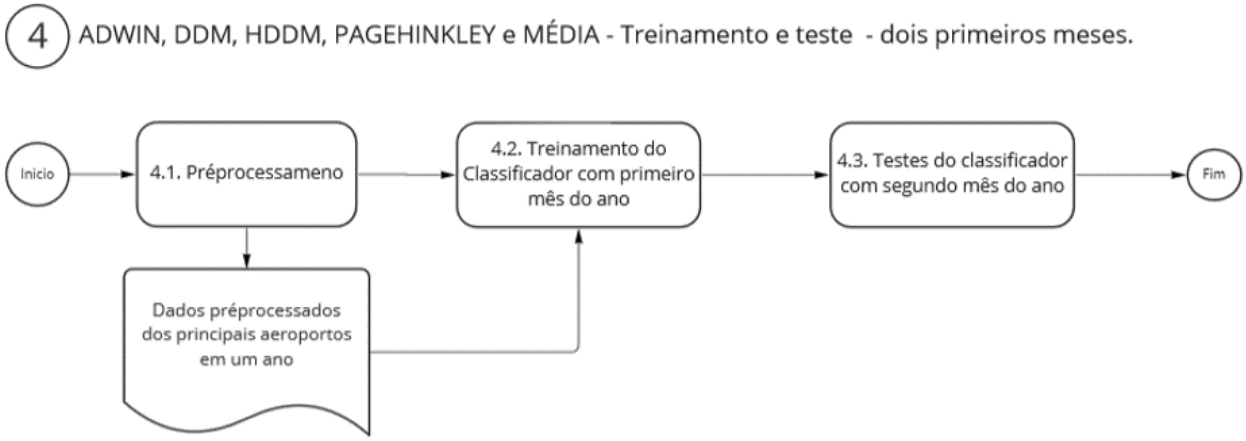


Figura VI.6: Retreina os modelos no Kairos.

Tanto o treinamento como o processamento de método ocorrem separadamente, mas sempre obedecendo uma mesma estrutura lógica, com os mesmos dados, períodos e formas de decisão sobre retreino e aferição de desempenho. A Figura VI.7 mostra o fluxo de processamento.

5. Ciclos de processamento dos modelos com a classificação de dados, para os métodos ADWIN, DDM, HDDM, Page Hinkley e Média.

   5.1 Seleciona o período para processamento, sempre com a janela de um mês, com incrementos mensais ou semanais diante da configuração selecionada. Ou seja, a janela de comparação armazenada e a janela de processamento sempre são de um mês, mas o incremento de novos dados pode ser feito a cada semana, ou a cada mês. Os experimentos foram feitos das duas formas.

   5.2 Nesta etapa, existe o processamento dos modelos.

   5.3 A forma de identificação do desvio varia para cada método, o que pode ocorrer de diversas formas, como foi melhor detalhado no Capítulo III. Caso o desvio seja identificado, os modelos são retreinados, conforme Figura VI.8.

   5.4 A cada passo, existe a comparação com os resultados reais, sendo registrados: tempo de processamento, indicação se houve desvio, indicação se houve retreino, acurácia, recall, precision e F1.

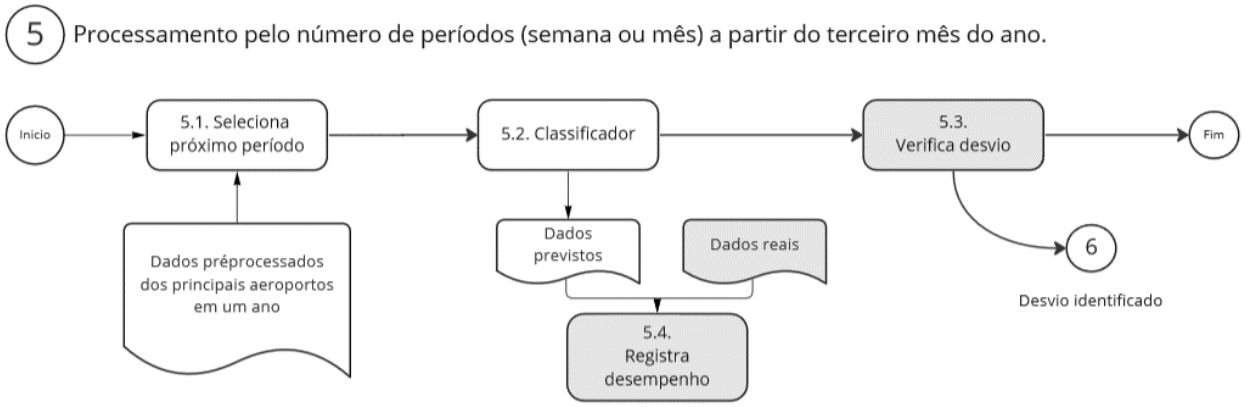


Figura VI.7: Fluxo de processamento dos modelos para os métodos ADWIN, DDM, HDDM, Page Hinkley e Média.

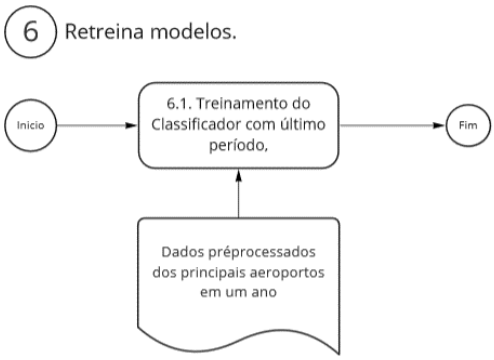


Figura VI.8: Fluxo de retreinamento dos modelos para os métodos ADWIN, DDM, HDDM, Page Hinkley e Média.

VI.4 Forma de avaliação e armazenamento

Nesta seção discorremos sobre a estrutura de comparações e avaliações desenvolvidas, ilustrada na Figura VI.9. Para cada período de processamento, feito para cada base de dados selecionada, houve a aferição e registro de: tempo de processamento em segundos, indicação se houve desvio, indicação se houve retreino, acurácia, recall, precision e F1. As métricas de desempenho comparam os dados, resultados dos modelos de classificação e a informação real de atraso de cada voo. Isso foi feito para todos os métodos avaliados: a classificação do mesmo modelo sem retreino; o retreino a cada período; os métodos ativos do estado da arte ADWIN, DDM, HDDM, Page Hinkley; o método ativo prévio deste trabalho, efetuado pela média; e, finalmente para o novo método Kairos.

Dessa forma foi possível a comparação isolada ou combinada de cada uma dessas métricas, nas diferentes escalas e tempo e por todas as bases de dados. Todos os dados gerados foram armazenados em arquivos CSV separadamente. Também foram gerados e armazenados gráficos para cada uma das métricas.

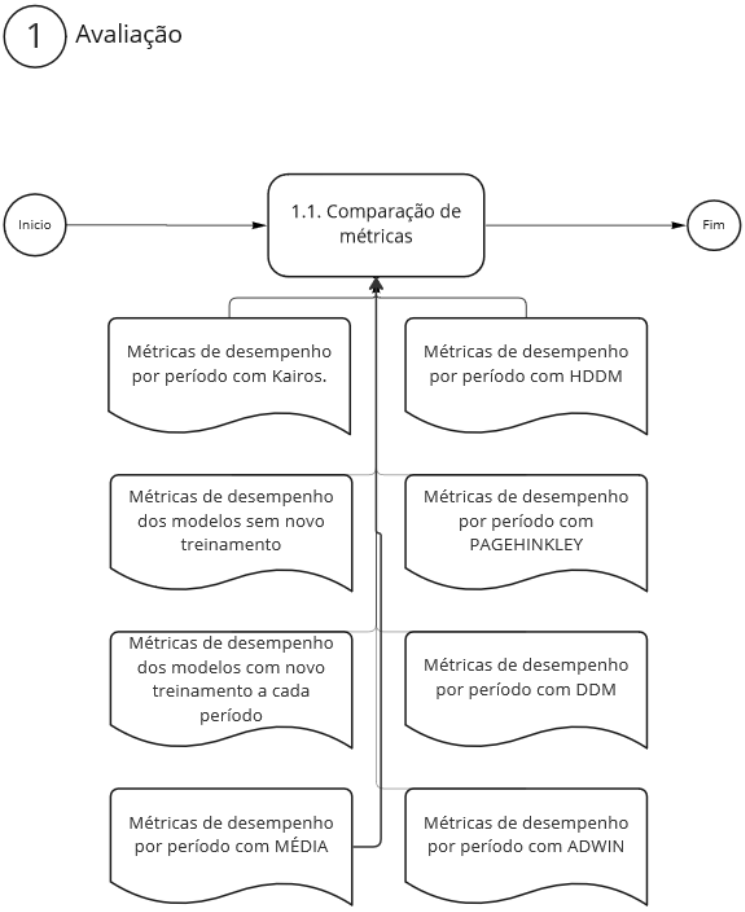


Figura VI.9: Estrutura básica de avaliação.

VI.5 Diferentes parâmetros para os experimentos

Muitas das escolhas possíveis das configurações e parâmetros foram experimentadas ao longo do trabalho, embora a cobertura desses experimentos não seja exaustiva. Esta seção descreve os demais parâmetros e configurações testados e os que foram utilizados para os experimentos. Ao final, foram utilizados para os experimentos os parâmetros com melhores resultados.

1. Janelas para comparação

   1.1 Primeira janela - opção de registrar a distribuição, janela inicial de teste, e sempre comparar com ela.

   1.2 [*Utilizado*] Armazenar e comparar com cada distribuição armazenada - armazena a distribuição no momento do teste e depois cada distribuição nos momentos de retreino, junto aos modelos treinados com esta.

2. Janela de retreinamento

   2.1 Todo o período anterior - utilização de todo o período anterior para treinamento.

   2.2 Seis meses - utilização do período de até 6 meses treinamento.

   2.3 [*Utilizado*] Um mês - utilização de um mês para treino e retreino.

3. Referência de distância

   3.1 Menor que o limite.

   3.2 Maior que o limite.

   3.3 Menor diferença.

   3.4 Limites.

   3.5 Menor diferença para cada registro.

   3.6 Menor diferença para cada registro com parâmetro α variando entre $0,30$ e $0,90$.

   3.7 [*Utilizado*] Menor diferença para cada registro com parâmetro $\alpha = 0,90$.

4. Número de registros na tabela

   4.1 Entre um e dez registros.

   4.2 [*Utilizado*] Cinco registros.

5. Parâmetros para detecção de desvio nos demais métodos

   Todos os parâmetros para detecção de desvio foram mantidos padrão conforme publicações, geralmente parâmetro $\alpha = 0,95$ ou $\alpha = 0,90$, conforme Capítulo III e Apêndice A.

Capítulo VII Resultados

Este capítulo contém os resultados obtidos e análises feitas a partir destes resultados. A primeira seção apresenta brevemente quais são as fontes de dados e como foram executados os experimentos; como foi montado o método proposto; como foram montados os métodos de comparação; e, como foram efetuadas as comparações. As demais seções apresentam as respostas para as perguntas de pesquisa listadas a seguir.

Como base para apresentação dos resultados, seguem as perguntas de pesquisa colocadas no Capítulo I: "Qual é o impacto da detecção e tratamento de desvio de conceito nos atrasos de voos na aviação e como isso se compara à abordagem tradicional sem considerar o desvio de conceito? Quais são as vantagens dos métodos ativos em relação aos passivos na detecção e tratamento do desvio de conceito nos atrasos de voos? Além disso, se existe ganho na aplicação de um novo método ativo para detecção e tratamento de desvios de conceito no contexto de atrasos aéreos, que trate suas características como comportamentos sazonais?"

Lista-se a seguir um conjunto de perguntas que irão direcionar as respostas para os perguntas gerais da pesquisa:

- Q1: A identificação e tratamento de desvios de conceito via método ativo consegue ser mais efetiva do que a manutenção do mesmo modelo ou algum método passivo?
- Q2: Os métodos de identificação e tratamento de desvios podem alterar sua eficácia ou comportamento diante de diferentes dimensões dos dados, como análises de informações de companhias aéreas, aeroportos ou toda a malha aérea?
- Q3: Os métodos de identificação de tratamento de desvios podem alterar sua eficácia ou comportamento diante da dimensão dos dados com relação ao período das janelas de testes e comparações?
- Q4: Quais são os métodos ativos atuais com melhor desempenho para cenários de aviação?
- Q5: Como o Kairos se comporta comparativamente, do ponto de vista de desempenho, ao estado da arte dos métodos de detecção de desvio para cenários de aviação?
- Q6: Qual a evolução do Kairos em relação ao método preliminar desenvolvido, método ativo proposto em Giusti et al. [2022]?

- Q7: A comparação entre o Kairos e o estado da arte dos métodos de detecção de desvio para cenários de aviação muda diante de diferentes dimensões dos dados, como análises de informações de companhias aéreas e aeroportos?
- Q8: Existe relação entre o número de desvios identificados e o desempenho dos métodos?
- Q9: Como os diferentes métodos avaliados se comportam do ponto de vista de tempo de processamento?

VII.1 Protocolos experimental e de dados

Esta seção apresenta os protocolos experimental e de dados. Descrevemos um resumo das escolhas e parâmetros utilizados para seleção e processamento fundamentais para o entendimento e reprodução dos experimentos. Subdividimos esta seção apresentando primeiro o protocolo experimental e em seguida o protocolo de dados. Existe um maior detalhamento sobre como foram estruturados os experimentos no Capítulo VI.

VII.1.1 Protocolo experimental

Foram desenvolvidos três diferentes modelos de aprendizado de máquina para classificação de atrasos em voos: Random Forest, Rede Neural e Naive Bayes. Os três modelos utilizam as mesmas bases de dados e são os mesmos utilizados em todas as formas de análise: sem retreino, método passivo e métodos ativos.

Como base de comparação, foram selecionados quatro métodos ativos de detecção de desvio, sendo um método representativo de cada quadrante em nossa taxonomia apresentada no Capítulo III: ADWIN, DDM, HDDM e Page Hinkley. Mantivemos também o método passivo de retreinamento a cada período, assim como a linha de base, sem detecção de desvio, que seria o não retreino dos modelos em qualquer período. Tanto o treinamento como o processamento de método ocorrem separadamente, mas sempre obedecendo uma mesma estrutura lógica, com os mesmos dados, períodos e formas de decisão sobre retreino e aferição de desempenho.

Para cada período de processamento, feito para cada base de dados selecionada, houve a aferição e registro de: tempo de processamento em segundos, indicação se houve desvio, indicação se houve retreino, e métricas de desempenho de classificação (acurácia, recall, precision e F1-score). As métricas de desempenho comparam os dados, resultados dos modelos de classificação e a informação real de atraso de cada voo. Isso foi feito para todos os métodos avaliados: a classificação do mesmo modelo sem retreino; o retreino a cada período; os métodos ativos do estado da arte ADWIN, DDM, HDDM, Page Hinkley; o método ativo proposto em Giusti et al. [2022], feito pela média; e,

finalmente para o novo método Kairos.

Foram utilizados os seguintes parâmetros:

- Janela de comparação: armazena a distribuição no momento do teste e depois cada distribuição nos momentos de retreino, junto aos modelos treinados com esta.
- Janela de retreino: utilização de um mês para treino e retreino.
- Tabela de registros: foi utilizado espaço de armazenamento de até cinco registros na tabela de registros.
- Referência de distância: menor diferença para cada registro com parâmetro $\alpha = 0,90$.
- Parâmetros para detecção de desvio nos demais métodos: todos os parâmetros para detecção de desvio foram mantidos padrão conforme publicações, geralmente parâmetro $\alpha = 0,95$ ou $\alpha = 0,90$, conforme Capítulo III e Apêndice A.

VII.1.2 Protocolo de dados

A nossa fase de pré-processamento parte do trabalho prévio de Teixeira et al. [2020]. Esse trabalho fornece informações sobre o Brazilian Flight Dataset (BFD), o qual transformamos em um novo formato para utilização por parte dos modelos de aprendizado de máquina. A partir dos arquivos gerados foram selecionadas três visões dos dados para diferentes processamentos através dos modelos, visando apurar possíveis diferenças nos resultados e comportamentos tanto dos modelos de aprendizado de máquina como dos métodos de detecção de desvio de conceito. Na visão de malha aérea, foram selecionados todos os voos dos dez principais aeroportos no Brasil, onde eles eram origens ou destino desses voos. Na visão de aeroportos, cada base de dados era composta apenas de voos com origem ou destino em determinado aeroporto. Por fim, na visão de companhias, cada base de dados era composta apenas de voos operados por determinada companhia.

Foram utilizados os seguintes parâmetros:

- Período: entre os anos de 2016 e 2018.
- Período de cada processamento: anual.
- Visões: malha aérea, aeroportos e companhia aérea.

Embora tenhamos destacado na introdução a relevância do período de COVID-19 (2019-2022) como um exemplo de desvio de conceito na aviação, optamos por não utilizar dados desse período nos experimentos realizados. Essa decisão foi tomada com base em alguns fatores que consideraremos, como uma análise onde os desvios não são tão explícitos como na pandemia. Primeiramente,

o período de COVID-19 foi marcado por mudanças significativas nas operações e nos padrões de voos devido a restrições de viagens, fechamento de fronteiras e outras medidas implementadas para conter a propagação do vírus. Essas alterações drásticas na indústria da aviação podem ter introduzido desvios de conceito atípicos e extremos nos dados, o que poderia comprometer a análise e a interpretação dos resultados.

Além disso, o período de COVID-19 foi caracterizado por uma alta volatilidade e incerteza, tornando desafiador estabelecer uma linha de base estável e representativa para os atrasos de voos. A presença de fatores externos e não relacionados ao desempenho operacional regular, como políticas governamentais e mudanças nas preferências dos passageiros, poderia obscurecer os efeitos do desvio de conceito que desejamos investigar.

Visando obter uma compreensão mais sólida dos atrasos de voos em condições mais estáveis e previsíveis, optamos por utilizar dados anteriores ao período de COVID-19 nos nossos experimentos. Essa escolha nos permite analisar padrões e comportamentos mais consistentes ao longo do tempo, facilitando a comparação e a interpretação dos resultados obtidos com os modelos de aprendizado de máquina e os métodos de detecção de desvio de conceito.

VII.2 Diferença entre métodos ativos, passivo e não tratamento

A pergunta `Q1` (questionamento sobre se a aplicação de métodos de detecção de desvio de conceito gerariam melhores resultados nos modelos de predição de atrasos de voo) já havia sido abordada inicialmente nos experimentos prévios, com resultados positivos. O objetivo do questionamento é verificar se a identificação e tratamento de desvios de conceito – método ativo – no caso de classificação de atrasos de voos consegue ser mais efetiva do que a manutenção do mesmo modelo ou algum método passivo, como retreino.

Nesta seção existe uma evolução da análise do cenário de detecção de desvios em atrasos de voos, onde são contemplados quatro métodos do estado da arte, ADWIN, DDM, HDDM e PAGE HINKLEY, conforme Capítulo III. Escolhemos um método de cada quadrante da taxonomia para compor os métodos ativos e ter uma visão mais abrangente dos cenários analisados anteriormente.

A resposta aqui se torna mais consistente, com o apoio dos resultados dos demais métodos ativos. Podemos ver na Figura VII.1 a comparação entre a aplicação do modelo sem retreino, linha de base, o método passivo e os métodos ativos, contemplando tanto a *media*, método ativo proposto em Giusti et al. [2022], como os demais métodos do estado da arte. Neste exemplo, com a base de dados de malha aérea, em período mensal, consolidado dos três anos, vemos claramente a prevalência dos métodos passivo e ativos sobre o não retreino. E a prevalência de dois dos métodos ativos utilizados.

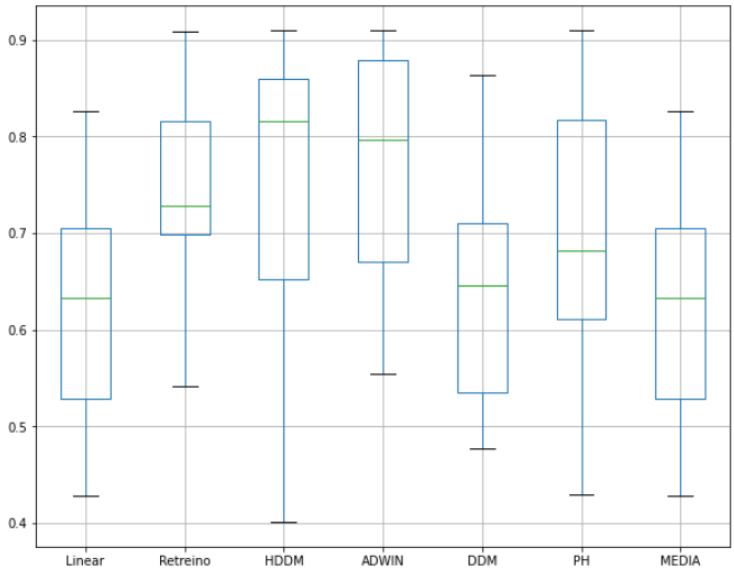


Figura VII.1: Comparação entre a linha de base, método passivo e ativos. Box plot média mensal do F1 de todos os modelos - entre 2016 a 2018 - visão de malha aérea.

VII.3 Diferença entre diferentes dimensões de dados

Em resposta ao `Q2`, nesta seção verificamos o comportamento dos métodos com a aplicação em diferentes dimensões de dados, notadamente as dimensões aeroporto, companhia e malha aérea. O objetivo foi averiguar se os métodos de identificação e tratamento de desvios podem alterar sua eficácia ou comportamento diante da dimensão dos dados.

A Figura VII.2 mostra a dimensão aeroportos, sendo a visão consolidada do processamento feito para cada aeroporto. Diferente da visão de malha aérea, todo o período foi processado separadamente apenas com voos partindo ou chegando de cada aeroporto. A figura mostra um cenário similar de prevalência dos métodos passivo e ativos sobre o não retreino.

A Figura VII.3 mostra a dimensão companhias, sendo a visão consolidada do processamento feito para cada companhia. Diferente da visão de malha aérea, todo o período foi processado separadamente apenas com voos operados por determinada companhia. Novamente, ocorre um cenário similar de prevalência dos métodos passivo e ativos sobre o não retreino.

VII.4 Diferença entre diferentes frequências de períodos de análise

Em resposta ao `Q3`, esta seção mostra os resultados dos experimentos em diferentes frequências de tempo de avaliação de desvio. Foram testados os períodos semanal e mensal em nossos experimentos. O objetivo foi apurar se os métodos de identificação de tratamento de desvios podem alterar sua eficácia ou comportamento diante da dimensão dos dados com relação ao período das

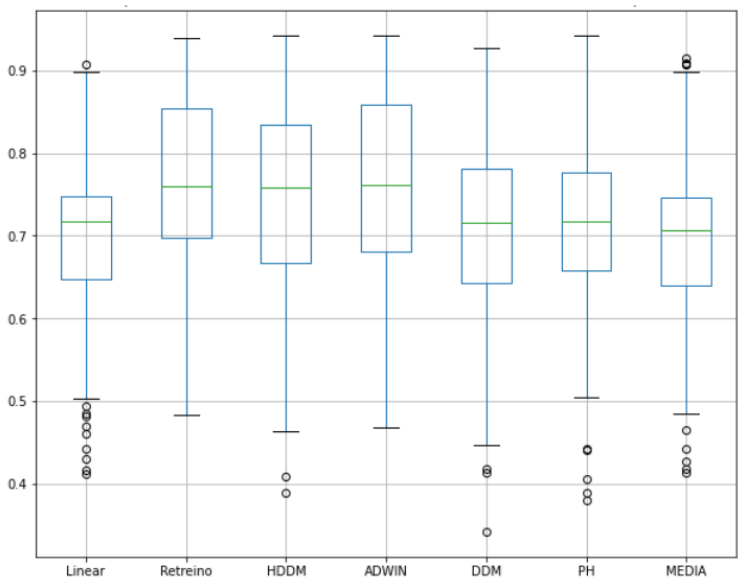


Figura VII.2: Comparação entre a linha de base, métodos passivo e ativos na visão aeroportos. Box plot média mensal do F1 de todos os modelos - entre 2016 a 2018 - visão aeroportos.

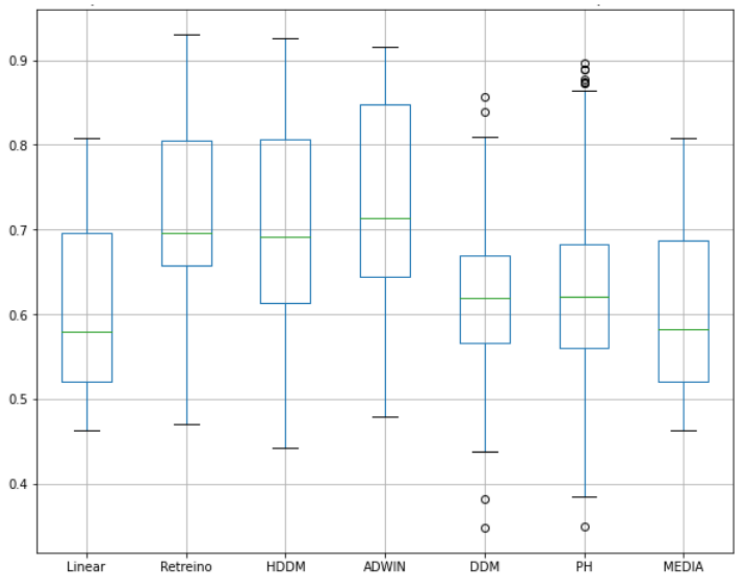


Figura VII.3: Comparação entre a linha de base, métodos passivo e ativos na visão de companhias. Box plot média mensal do F1 de todos os modelos - entre 2016 a 2018 - visão de companhias aéreas.

janelas de testes e comparações.

Apuramos que sim, existe diferença na aplicação de métodos de detecção de desvios em diferentes períodos de análise. Os experimentos mostram que o tratamento de desvio foi melhor que o não tratamento. Mas, com janelas de testes e comparações menores, como semanal, o tratamento passivo foi melhor em uma avaliação consolidada. Isso pode ser explicado pela menor taxa de identificação de desvios, pelas janelas curtas de comparação. Em contrapartida, com uma frequência muito maior de retreinamento através do método passivo.

A Figura VII.4 mostra a comparação entre a linha de base, métodos passivo e ativos na visão

malha aérea, período semanal. Mesma visão de malha aérea mostrada na Figura VII.1, com o processamento dos três anos, mas aqui com a detecção do desvio e retreino do método passivo a cada semana.

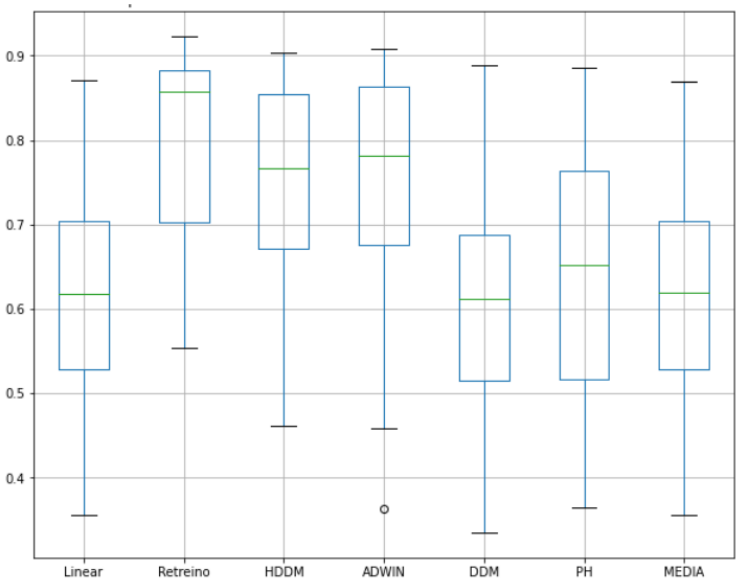


Figura VII.4: Comparação entre a linha de base, métodos passivo e ativos na visão malha aérea, período semanal. Box plot média semanal do F1 de todos os modelos - entre 2016 a 2018 - visão de malha aérea.

VII.5 Estado da Arte para cenário de aviação

Em resposta ao `Q4`, nesta seção, identificamos os métodos ativos com melhores resultados nas métricas de desempenho, dentre os métodos do estado da arte em desvio de conceito. Para verificar quais são os métodos ativos com melhor desempenho para cenários de aviação, e serão então usados como base de comparação com o novo método. Embora o foco seja a comparação com os métodos ativos, mantém-se o método passivo - retreino a cada período - no estado da arte pelo seu destaque em termos de desempenho no cenário de aviação.

Conforme podemos verificar nas Figuras VII.1, VII.2 e VII.3, os métodos HDDM e ADWIN foram os métodos com melhores métricas de desempenho para os cenários analisados. Ambos métodos de análise com janelas deslizantes. Esse é um indicativo para utilização de métodos com janelas deslizantes para tratamento de desvios de conceito em cenários de classificação de atrasos em voos.

VII.6 Comparação do Kairos ao estado da arte para cenário de aviação

Nesta seção segue a resposta ao `Q5`. Uma vez selecionada a base de comparação, formada pelos métodos ativos HHDM e ADWIN, além do método passivo, os métodos com melhor resultados, e que neste trabalho identificamos como o estado da arte para o cenário de detecção de desvios de conceito para classificação de atrasos em voos, pudemos desenvolver o Kairos. A Figura VII.5 mostra a comparação consolidada entre os métodos passivo, HDDM, ADWIN e Kairos. Vemos que o Kairos apresentou melhores resultados em relação aos demais métodos selecionados. Observamos na figura que a média do F1 de desempenho dos modelos usados com método Kairos têm maior concentração acima de 80%.

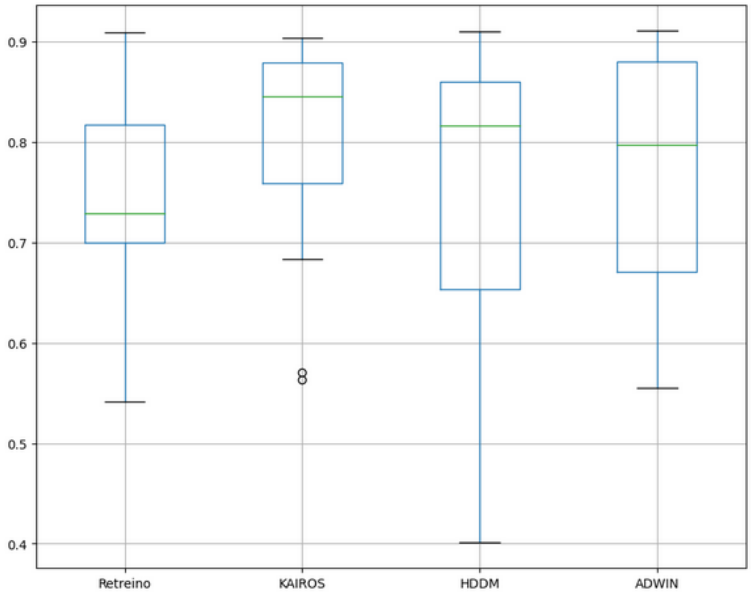


Figura VII.5: Comparação os métodos passivo, ativos ADWIN, HHDM, e Kairos. Box plot do valor médio do F1 entre Kairos e métodos do Estado da Arte.

VII.7 Evolução do método por diferença entre distribuições

Nesta seção segue a resposta ao `Q6`. O método preliminar, por análise da diferença nas médias entre as distribuições, apresentou resultados claramente inferiores nas métricas de desempenho, e por isso, nem faz parte de base de comparação com o Kairos. Porém, julgamos importante efetuar uma comparação entre os dois para mostrar a evolução no desenvolvimento do método desde a ideia inicial. A Figura VII.6 mostra a comparação consolidada dos resultados do método por *media* e o Kairos.

O desempenho do Kairos foi superior ao método prévio, sempre utilizando apenas a comparação entre as distribuições. Essa evolução foi possível diante das informações coletadas anteriormente

que possibilitaram indicações de melhoria no método, assim como observação dos demais métodos do estado da arte em detecção de desvio de conceito.

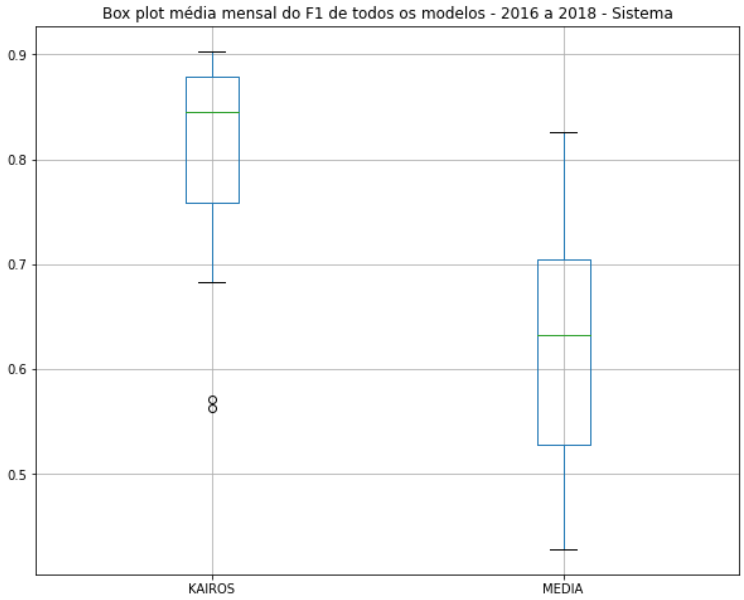


Figura VII.6: Comparação consolidada dos resultados dos métodos por *media* e o Kairos. Comparação entre média mensal do F1 dos modelos - entre 2016 a 2018 - visão de malha aérea.

VII.8 Comparação do Kairos em diferentes bases de dados

Neste seção verificamos o comportamento com a aplicação em diferentes dimensões de dados, notadamente as dimensões aeroporto, companhia e malha aérea, dentre os métodos passivo, ativos ADWIN, HHDM, e Kairos. Visando averiguar se os métodos de identificação e tratamento de desvios pode alterar sua eficácia ou comportamento diante da dimensão dos dados.

A Figura VII.7 mostra a dimensão aeroportos, sendo a visão consolidada do processamento feito para cada aeroporto. Diferente da visão de malha aérea, todo o período foi processado separadamente apenas com voos partindo ou chegando de cada aeroporto. Ao final, vemos aqui uma visão consolidada diferente da malha aérea, mas que mostra um cenário similar, de prevalência do Kairos em comparação aos demais métodos do estado da arte em aviação. Observamos na figura que a média do F1 de desempenho dos modelos usados com método Kairos têm maior concentração acima de 80%.

A Figura VII.8 mostra a dimensão companhias, sendo a visão consolidada do processamento feito para cada companhia. Diferente da visão de malha aérea, todo o período foi processado separadamente apenas com voos operados por determinada companhia. Ao final, vemos aqui uma visão consolidada diferente da malha aérea, mas que mostra um cenário similar, de prevalência do Kairos em comparação aos demais métodos do estado da arte em aviação. A média do F1 de

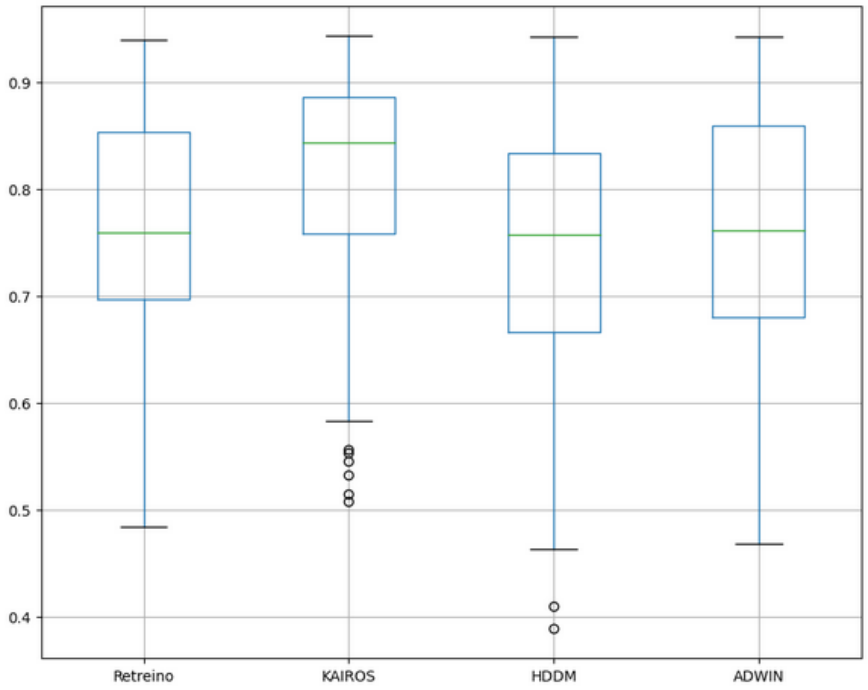


Figura VII.7: Comparação entre o método passivo, HDDM, ADWIN e Kairos na visão aeroporto. Box plot do valor médio do F1 do Kairos e Estado da Arte - entre 2016 a 2018 - visão de aeroportos.

desempenho dos modelos usados com método Kairos têm concentração acima de 80% similar ao ADWIN, mas apresenta melhores valores que os demais acima de 60%.

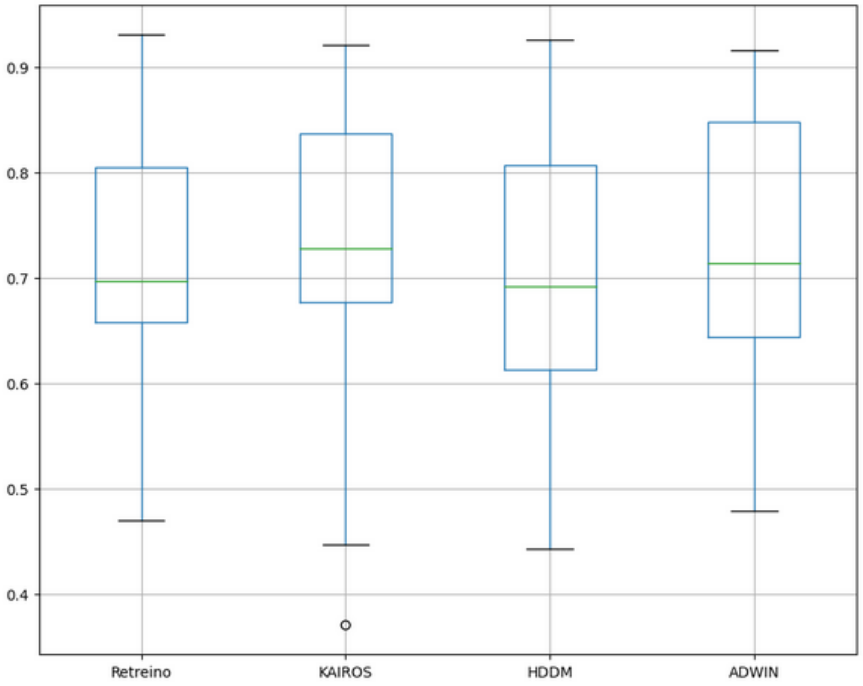


Figura VII.8: Comparação entre HDDM, ADWIN e Kairos na visão de companhias. Box plot do valor médio do F1 do Kairos e Estado da Arte - entre 2016 a 2018 - visão de companhias aéreas.

VII.9 Comparação do Kairos em diferentes frequências de tempo

Em resposta ao Q7, nesta seção verificamos o comportamento com a aplicação em diferentes frequências de tempo de avaliação e tratamento de desvios, dentre os métodos passivo, ativos ADWIN, HHDM, e Kairos. Visando averiguar se os métodos de identificação e tratamento de desvios pode alterar sua eficácia ou comportamento diante de diferentes frequências de tempo.

A Figura VII.5 mostrou a comparação consolidada entre os métodos na frequência mensal de identificação e tratamento de desvios. O Kairos apresentou melhores resultados em relação aos demais métodos selecionados. O Kairos se mostrou melhor em identificar os desvios e efetuar o tratamento nos cenários apresentados. A Figura VII.9 mostra a mesma comparação na frequência semanal. Ela mostra um cenário similar, de prevalência do Kairos em comparação aos demais métodos ativos do estado da arte em aviação. Pode-se ver também um empate no desempenho do Kairos e o método passivo neste cenário onde existe um novo treinamento do modelo a cada período mais curto, como o semanal.

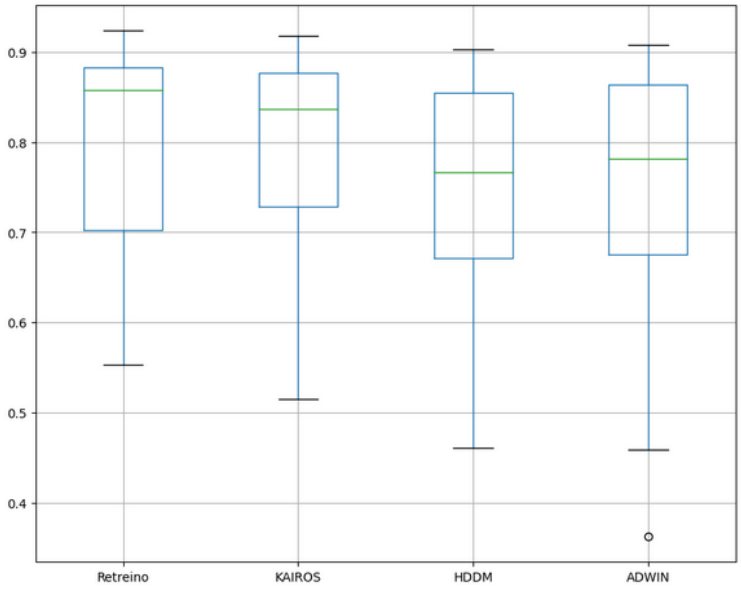


Figura VII.9: Comparação entre o método passivo, ativos HDDM e ADWIN, e Kairos na frequência de tempo semanal. Plot da média semanal do F1 de todos os modelos - entre 2016 a 2018 - visão de malha aérea.

VII.10 Análises sobre número de desvios que ocorrem nos voos

Em resposta ao Q8, nesta seção analisamos o número de desvios identificados com a utilização de cada método, dentre o Kairos e os métodos do estado da arte em aviação. A primeira diferença que conseguimos identificar, e pode ser vista na Figura VII.10, é que o número de desvios muda

conforme o modelo de aprendizado de máquina no ADWIN e HDDM, e de uma única forma no Kairos. Essa é uma característica do Kairos que o difere dos demais métodos ativos. Como o Kairos efetua a detecção do desvio apenas avaliando o histórico de dados e novas distribuições, em uma linha de tempo, a identificação do número de desvios e momentos indicados para tratamento independem do modelo de aprendizado de máquina utilizado. Como os demais métodos de detecção de desvios utilizam alguma métrica de desempenho, o número de desvios identificados e momentos indicados para tratamento podem diferir dependendo do modelo de aprendizado de máquina. Vemos claramente que a utilização de ADWIN e HHDM registram números diferentes de desvios, e em momentos diferentes, para cada um dos três modelos de aprendizado de máquina utilizados.

Outra característica específica do Kairos em relação aos demais é o fato de reaproveitar modelos já treinados, a partir da tabela de registros. Isso significa que existe um índice de reaproveitamento que faz que número de desvios e número de novos treinamentos sejam números diferentes. É possível afirmar que o índice de reaproveitamento será menor ou igual ao número de desvios detectados. Isso ocorre porque, quando um desvio é identificado, tem-se a opção de reutilizar algo já existente ou realizar um novo treinamento. Portanto, o número de desvios representa um limite superior para o número de novos treinamentos necessários.

Seja: N: o número total de desvios detectados. R: o número de desvios em que é possível reutilizar modelo já existente. T: o número de desvios que requerem um novo treinamento.

O índice de reaproveitamento (I) pode ser formalizado da seguinte forma:

$$I = \frac{R}{N}$$

O índice de reaproveitamento (I) é calculado dividindo o número de desvios em que é possível reutilizar algo já existente (R) pelo número total de desvios detectados (N). Ele fornece uma medida da proporção de desvios onde a reutilização é viável em relação ao total de desvios. É importante destacar que o índice de reaproveitamento (I) é limitado superiormente pelo número de desvios (N), uma vez que todos os desvios requerem alguma ação, seja reutilização ou novo treinamento. Portanto, $I \leq N$. O Kairos apresentou índice de pelo menos 40% nos experimentos ($I = 0,4$). Por exemplo, na Figura VII.10 existem dezessete desvios identificados pelo Kairos, o que, na verdade, se traduziu em dez novos treinamentos. Então, do ponto de vista de comparação de número de novos treinamentos, o valor de comparação do Kairos é de dez, contra, por exemplo, trinta do método passivo.

Seguindo na análise, pode-se ver ainda que na aplicação do ADWIN o número de desvios para os modelos Naive Bayes e Random Forest ficou em patamar maior que o registrado pelo Kairos. E que na aplicação do HDDM, o número de desvios para os modelos Naive Bayes e Random Forest no mesmo patamar pelo registrado no Kairos. O modelo de Redes Neurais não registrou desvios

tanto na aplicação do ADWIN quanto do HDDM, onde então podemos fazer uma comparação sobre número de desvios. Podemos observar na Figura VII.5 que o HDDM teve melhores resultados nas métricas de desempenho em relação ao ADWIN, ao mesmo tempo que teve um número menor de desvios identificados e, por conseguinte, um número menor de retreinamentos do modelo. O que nos leva a identificar que o número de desvios de retreinamentos por si só não determinam melhores resultados, mas sim os momentos mais adequados de retreinamento derivado dos melhores momentos na detecção dos desvios.

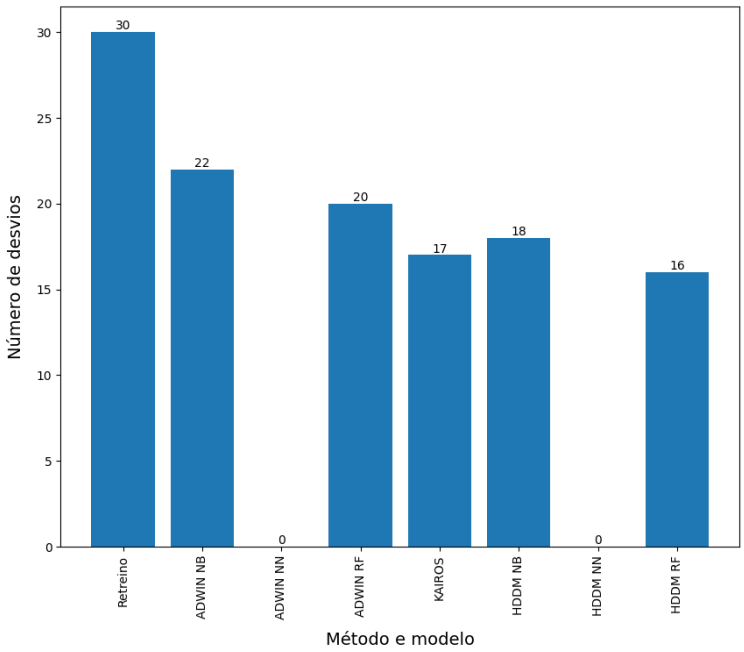


Figura VII.10: Comparação do número de desvios de conceito registrados com uso do HDDM, ADWIN e Kairos. Número de desvios por método e modelo, base mensal, na visão de malha aérea."

A inexistência de registro de desvios no modelo de Redes Neurais tanto no ADWIN quanto no HDDM são decorrentes da forma de detecção de desvio pelas métricas de desempenho, na distância entre ocorrência de erros. As formas de detecção foi detalhada no Capítulo III. Essa diferença grande no comportamento apenas no modelo de Redes Neurais nos fez analisar este resultado em específico, seguindo na Figura VII.11. Vemos que a identificação dos desvios e tratamento que ocorreu no Kairos gerou melhor resultado que os demais métodos, ADWIN e HDDM. A média do F1 dos modelos com uso do Kairos e método passivo têm maior concentração acima de 80%.

Importante salientar nesta análise que o Kairos teve desempenho similar ao método passivo. Porém, o kairos detectou desvio em cerca de 56% (17 em 30 meses) dos períodos, contra um novo treinamento em todos os 30 meses por parte do método passivo. E, considerando o índice de reaproveitamento, esse número de novos treinamentos efetivos é bem menor. O Kairos chegou a um número de novos treinamentos, que mesmo muito abaixo do número total de períodos, apresentou o mesmo desempenho do método passivo. Isso indica que a identificação e os novos treinamentos

ocorreram no momento efetivo para propiciar melhor desempenho, sem maior uso de tempo ou recursos.

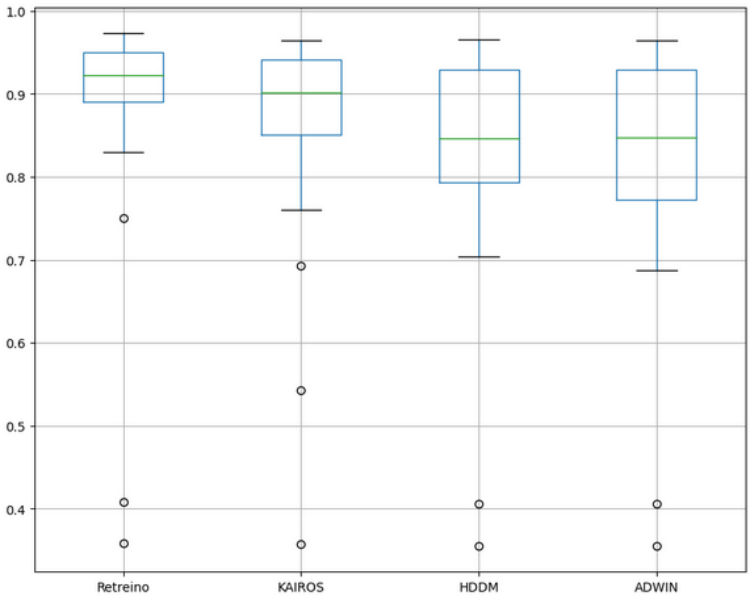


Figura VII.11: Comparação os métodos passivo, ativos ADWIN e HHDM, e Kairos, apenas com processamento do modelo de Redes Neurais. Box plot mensal do F1 de Redes Neurais - entre 2016 a 2018 - na visão de malha aérea.

VII.11 Avaliação sobre tempo decorrido

Em resposta ao `Q9`, efetuamos aqui uma análise relacionada ao tempo decorrido nos processamentos usando o Kairos e demais métodos do estado da arte em aviação, ADWIN e HDDM. Existe registro do tempo decorrido a cada processamento, o que foi salvo no formato de arquivos *CSV* e gráficos. Aqui apresentamos uma visão geral, consolidada dos três anos, que traduz um comportamento similar ao encontrado em todas as dimensões. O Kairos apresentou o maior tempo em relação aos demais, cerca de 30% a mais que o ADWIN e 60% a mais que o HDDM. Para todos os trinta e seis meses, incluindo treinamento, processamento, detecção do desvio e tratamento, o Kairos levou cerca de quatro minutos. Aproximadamente sete segundos para cada mês, seguidos de cinco segundos gastos pelo ADWIN e três segundos pelo HDDM.

A comparação considerou o processamento de todos os três modelos de aprendizado de máquina. Pela característica do Kairos, de detectar o desvio apenas pelos dados de entrada, independente do modelo de aprendizado de máquina, existe a detecção do desvio e tratamento de todos os modelos a cada desvio de conceito. Nos demais métodos, isso ocorre dependendo dos modelos e, como observamos na Seção VII.10, não houve detecção ou tratamento de desvios para o modelo de Redes Neurais, tanto no ADWIN quanto no HDDM. Essa característica fez com que o Kairos efetuasse número significativamente maior de retreinos do modelo de Redes Neurais, explicando o maior

tempo de processamento. Considerando todos os modelos de aprendizado de máquina utilizados, o ADWIN identificou quarenta e dois desvios, o HDDM vinte e quatro desvios e o Kairos cinquenta e um desvios, o que se aproxima muito aos percentuais de diferença nos tempos entre os métodos. A conclusão da análise é o tempo ser diretamente proporcional ao número de desvios identificados e não ao mecanismo do método em si. Ou seja, o método Kairos não oferece menor desempenho no tempo na sua mecânica de processamento. Os tempos entre os três métodos são similares, e podemos verificar a diferença no tempo decorrido diretamente através do número de desvios identificados. Da mesma forma, pode-se ver que não existe ganho de desempenho em tempo de processamento no aspecto do reaproveitamento de modelos e do índice de reaproveitamento discutido na seção anterior.

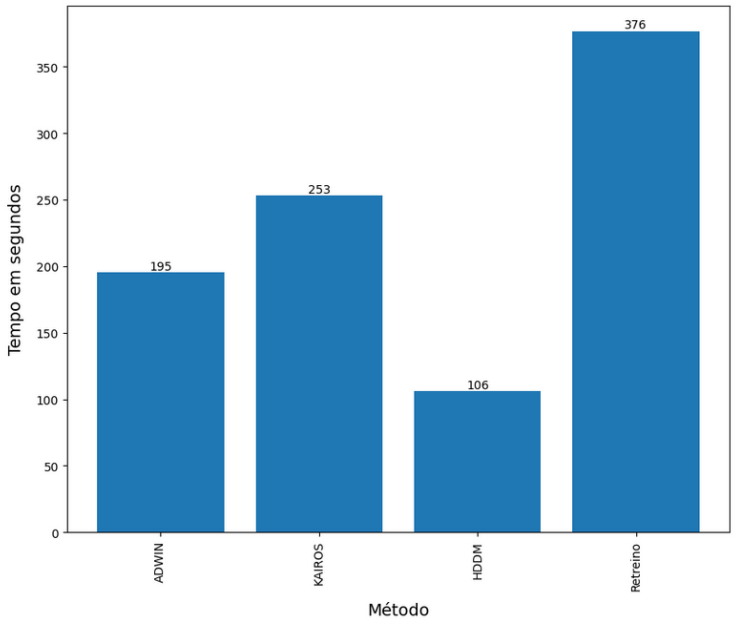


Figura VII.12: Comparação do tempo de processamento decorrido registrados com uso do HDDM, ADWIN e Kairos. Tempo decorrido (em segundos), base mensal, visão de malha aérea.

Capítulo VIII Conclusões

Este capítulo contém as considerações finais deste trabalho. Aqui, as contribuições e descobertas mais importantes são destacadas, com discussão adicional sobre a metodologia, a análise e os experimentos. O capítulo conclui com o direcionamento de futuros trabalhos, com a evolução do método com base nos experimentos e resultados.

Os modelos são geralmente estruturados com dados históricos que dão suporte a treinamento e teste. No entanto, há mudanças que ocorrem com o passar do tempo que tornam os resultados dos modelos menos precisos. O contexto de dados pode ser modificado, por exemplo, por mudanças nas leis, regulamentos, participantes, aspectos econômicos, políticos e de saúde, entre outros. Essas mudanças são chamadas de Desvio de Conceito, que exige que os modelos se adaptem enquanto permanecem funcionais. Essas mudanças podem ser pontuais ou não, repentinas, incrementais e alternadas. Essa mudança de contexto de dados é inerente ao cenário de voo e, portanto, é crucial para prever atrasos de voos. Essa complexidade deve ser trazida aos modelos de tratamento, de forma ativa ou passiva, para serem mantidos com um nível mínimo de confiança, mitigando problemas operacionais e possibilitando melhorias no sistema de planejamento de voo, rotas e operações gerais. Concluímos que a adoção da análise de Desvio de Conceito não só é viável, mas que começa a ser estudada nos cenários de aviação, ainda incipientemente, mas mostra um claro complemento aos modelos existentes, com ganhos mensuráveis. O principal objetivo do nosso trabalho é desenvolver uma metodologia que possa ser usada para detectar e tratar desvios de conceito em modelos de aprendizado de máquina na área de aviação. Para isso apresentamos o Kairos - Método de detecção e tratamento de desvio de conceito para previsão de atrasos de voos.

O estado da arte foi analisado através de mapeamento sistemático. Os métodos também foram classificados e detalhados para ser possível identificar pontos de melhoria e forma mais adequada de aplicação no setor de aviação. Nesta análise identificamos que os modelos atuais não consideram a área de aplicação, como a aviação. Em sua maioria são agnósticos, com objetivo de aplicação geral. Este foi um dos pontos de interesse e aplicação desta tese, foco na área para obtenção de melhores resultados.

Considerando a classificação apresentada na pesquisa, escolhemos um método que atua por análise de distribuições, e que trabalha com janelas deslizantes. As mudanças nas distribuições originais são determinantes nas mudanças do modelo e podem ser detectadas previamente com menor custo e tempo despendidos. Quanto mais ao final do processamento do modelo de aprendizado

de máquina, maior o custo computacional para detecção de desvio e maior o tempo gasto para o mesmo. Entendemos que a forma prévia e por distribuições também é a melhor forma de tratar fluxos de dados em tempo real. Em complemento, acreditamos que o uso de janelas deslizantes seja o mais indicado para o caso de aviação, por se tratar de área onde ocorrem distribuições sazonais e com fluxos de dados constantes, casos onde esta técnica é aplicada.

Conforme os experimentos preliminares, conforme Apêndice A, mostramos ser possível a aplicação de métodos de detecção de desvios de conceito em modelos de predição de atrasos em voos. Além de aplicáveis, os experimentos mostraram que sua aplicação gera diferencial positivo nos resultados de modelos de predição de atrasos de voo. Também comprovamos ser possível a detecção de desvio através da avaliação de diferenças nas distribuições de entrada, utilizando janelas deslizantes. Com os resultados preliminares positivos, avançamos com o refinamento do método proposto.

Para dar maior abrangência na aplicação nos cenários de classificação de atrasos em voos, implementamos experimentos com o estado da arte em detecção de desvios de conceito. Para esta comparação foram elencados quatro métodos (ADWIN e DDM, HDDM, PAGE HINKLEY), um de cada quadrante da taxonomia apresentada. Comprovamos que a aplicação de métodos ativos de identificação e tratamento de desvios de conceito consegue ser mais efetiva do que a manutenção do mesmo modelo ou algum método passivo, como retreino.

Efetuamos os experimentos em diferentes bases de dados e visões e comprovamos que os resultados são similares. Assim como na visão de malha aérea, os métodos de identificação e tratamento ativos foram prevalentes nas visões por aeroporto e companhia aérea. Os experimentos mostram que o tratamento de desvio foi melhor que o não tratamento. Mas, com janelas de testes e comparações menores, como semanal, o tratamento passivo foi melhor. Isso pode ser explicado pela menor taxa de identificação de desvios pelos métodos ativos, devido à janela curta.

Em vista do cenário amplo de análise para aviação, identificamos o Estado da Arte em aviação. Os resultados mostram que o HDDM e ADWIN são os métodos com melhores métricas de desempenho para os cenários analisados. Ambos métodos de análise com janelas deslizantes, o que foi um indicativo de que seleção deste tipo de mecanismo foi correta para os experimentos prévios, e que então, foi mantido para o desenvolvimento do Kairos.

O Kairos foi implementado com base no método prévio, com a detecção de desvio através da avaliação de diferenças nas distribuições de entrada, e utilizando janelas deslizantes, através da divergência de Kullback-Leibler como base. E o registro histórico de distribuições de referência e modelos treinados para reaproveitamento. O método foi então comparado ao estado da arte em aviação identificado. O Kairos ultrapassa o estado da arte, abrangendo também o tratamento do desvio, e não apenas a detecção. E, diante da sua estrutura de registro de histórico, diferente dos modelos existentes, o nosso modelo tende a fica mais estável conforme o tempo de utilização. A

medida que o histórico de distribuições e medidas de referência são acumuladas, mais resiliente, o sistema se tornará e menos desvios são detectados com o tempo.

Com base nos experimentos, verificamos que o nosso método foi o melhor método ativo, dentre os métodos do estado da arte utilizados para comparação. Os resultados foram também similares em termos de resultados consolidados para todos os cortes de dados.

As principais contribuições deste trabalho são:

- Pesquisa ampla sobre desvio de conceito e estruturação de taxonomia que pode ser aplicada amplamente;
- Trabalho inédito sobre desvio de conceito na aviação com implementação de comparação dos métodos do Estado da Arte da disciplina;
- Identificação do Estado da Arte de desvio de conceito para aviação; e,
- Implementação de novo método que abrange tanto a detecção como o tratamento de desvios de conceito, e que apresentou os melhores resultados que o estado da arte em aviação.

Embora o novo método tenha sido até aqui testado em cenários de classificação de atrasos em voos, não existe impedimento em sua aplicação em quaisquer outros cenários. Toda a ideia pode ser aplicada a qualquer cenário para detecção e tratamento de desvios de conceito. Como evolução deste trabalho, indicamos testes e aplicações do método em cenários diversos para comprovar sua aplicabilidade e eficácia em sentido amplo.

Trabalhos futuros:

- Complementação do sistema de detecção de desvio de conceito com uma análise de desempenho abrangente. Isso envolveria a avaliação das métricas de desempenho do método de detecção de desvio de conceito, como taxa de detecção, taxa de falsos positivos e tempo de resposta. Além disso, comparar o desempenho do sistema proposto com outros métodos existentes de detecção de desvio de conceito, utilizando conjuntos de dados de referência ou conjuntos de dados sintéticos. Essa análise permitiria uma compreensão mais completa da eficácia e confiabilidade do sistema, auxiliando na sua melhoria contínua.
- Realização de experimentos em novos cenários fora da aviação, a fim de avaliar a aplicação genérica do método como uma ferramenta de detecção e tratamento de desvio de conceito. Isso envolveria a aplicação do sistema em diferentes domínios, como finanças, medicina, manufatura, entre outros, para verificar sua efetividade em lidar com diferentes tipos de dados

e contextos. Esses experimentos em cenários diversificados permitiriam uma análise mais abrangente das capacidades e limitações do método, bem como a identificação de possíveis ajustes ou adaptações necessárias para sua aplicação em contextos específicos.

- Para facilitar a adoção do método de detecção e tratamento de desvio de conceito proposto, desenvolvimento de uma biblioteca ou *framework* que disponibilize uma implementação pronta e fácil de usar. Essa biblioteca poderia conter algoritmos, funções e recursos específicos para a detecção e tratamento de desvio de conceito, bem como documentação abrangente e exemplos de uso. Isso permitiria que pesquisadores e profissionais de diferentes áreas utilizassem o método de forma mais acessível, acelerando sua aplicação em diversas aplicações práticas e promovendo seu avanço como uma ferramenta útil no combate ao desvio de conceito.

Apêndice A Pesquisa prévia

Este capítulo apresentará histórico de pesquisa, direcionamentos e resultados prévios que nos levaram ao estado atual. O conjunto com o referencial teórico e trabalhos relacionados abordados, servirá de base e motivador para as escolhas tomadas ao estruturarmos o novo modelo. Abaixo, seguem experimentos prévios de detecção de desvio de conceito ativo na aviação. Estes experimentos serviram para direcionar e embasar a proposta apresentada no Capítulo V.

O objetivo deste trabalho prévio foi o de testar premissas e hipóteses básicas para o modelo proposto, quais sejam:

- Possibilidade da aplicação de métodos de detecção de desvio de conceito em modelos de predição de atrasos em voos;
- Que a aplicação de métodos de detecção de desvio de conceito geram diferencial positivo nos resultados de modelos de predição de atrasos de voo; e,
- Mostrar ser possível a detecção de desvio através da avaliação de diferenças nas distribuições de entrada, sem o processamento do modelo, conforme proposta de um modelo prévio e ativo.

Buscamos atendimento desses objetivos com um modelo simplificado, em relação ao proposto, com a detecção prévia e ativa usando formas de medida de distância entre distribuições padrão, como média e variância. E aqui sem a seleção ou armazenamento de contextos. Para dessa forma conseguir, como descrito e detalhado nas seções abaixo, demonstrar que existe base sólida para as hipóteses que serão desenvolvidas com maior completude e comparações com estado da arte na área de desvio de conceito.

A.1 Metodologia reduzida aplicada

Em Giusti et al. [2022] apresentamos trabalho prévio com o objetivo de estruturar método de detecção de desvio de conceito, especificamente, três estratégias *ativas* diferentes e uma estratégia *passiva*. Também apresentamos experimentos em diferentes escalas de treinamento, observando dados baseados em aeroportos isoladamente e em dados de todo o sistema, onde foi considerando informações dos dez maiores aeroportos do Brasil, apenas voos comerciais.

O algoritmo 5 descreve a metodologia utilizada para previsão de atrasos de voos com detecção de desvio de conceito. O mesmo requer sete parâmetros: D, $aeroporto$, mlm, t, b, dd e dh. O

parâmetro D corresponde ao conjunto de dados de entrada. O parâmetro *aeroporto* identifica um único aeroporto se a escala AB for usada ou *nulo* se SB for usada. O parâmetro mlm corresponde ao método de aprendizado de máquina. O parâmetro t está relacionado ao tempo e b corresponde ao tamanho da sequência do lote. Finalmente, dd e dh correspondem ao método de detecção, desvio de conceito e estratégia de tratamento de desvio, respectivamente. Esses parâmetros são descritos na Tabela A.1.

Algoritmo 5 Metodologia geral

1: **função** $metodologia(D, aeroporto, mlm, t, b, dd, dh)$
2: $D \leftarrow preprocessamento(D, aeroporto)$
3: $D_i \leftarrow selecionaPeriodoTreinamento(D, t, b)$
4: $D_j \leftarrow selecionaPeriodoTreinamento(D, t-1, b)$
5: $desvioDetectado \leftarrow detectaDesvio(dd, dh, D_i, D_j)$
6: **se** $desvioDetectado$ **então**
7: $mdl \leftarrow trainModel(mlm, D_i)$
8: $armazenaModel(mdl, airport, mlm, dd, dh)$
9: **senão**
10: $mdl \leftarrow carregaModelo(airport, mlm, dd, dh)$
11: **fim se**
12: $D_{t+1} \leftarrow selecionaPeriodoTreinamento(D, t+1)$
13: $resultados \leftarrow teste(mdl, D_{t+1})$
14: return $resultados$
15: **fim função**

Tabela A.1: Parâmetros usados para o experimento

Nome	Descrição [valores]
D	Base de dados integrada [Teixeira et al., 2020] contendo todos os voos brasileiros (ANAC) com registros meteorológicos (ASOS) [de 2000 a 2018]
$aeroporto$	Contendo todos os voos brasileiros (ANAC) com registros meteorológicos (ASOS) [de 2000 a 2018] AB; or *nulo*, for SB analisa [*nulo*, $SBBR$, $SBPA$, $SBSV$, $SBGL$, $SBCT$, $SBKP$, $SBGR$, $SBCF$, $SBRJ$, $SBSP$]
mlm	Modelos de aprendizagem de máquina: Naive Bayes (NB), Neural Networks (NN), Random Forest (RF) [NB, NN, RF]
t	Tempo (intervalos de tempo com base anual) [de 2003 a 2017]
b	BSS Tempo (intervalos de tempo com base anual) [de 2003 a 2017]
dd	detecção Desvio [$median$, $variancia$, $media/variancia$]
dh	Manipulação do Desvio [$linhadebase$, $ativo$, $passivo$]

A primeira etapa do Algoritmo 5 faz o pré-processamento dos dados (função *preprocessamento*). Se *aeroporto* diferir de *nulo*, o conjunto de dados D será filtrado para esse aeroporto. Caso contrário, estuda todo o SB. As linhas 3 e 4 preparam os lotes de treinamento D_i e D_j do BSS de b. D_j está atrasado uma posição para permitir a detecção de desvio de conceito.

A linha 5 calcula a ação de desvio. Ele considera o método de detecção de desvio (dd), a estratégia de tratamento de desvio (dh) e as duas sequências em lote (D_i e D_j). Especificamente, implementamos as estratégias de tratamento $linhadebase$, $ativa$ e $passiva$. A $linhadebase$ cor-

responde ao treinamento usando a primeira sequência de lote para prever todos os outros lotes. Ou seja, um tratamento padrão mais comum aplicado ao métodos de aprendizado de máquina. A estratégia *passiva* treinou um modelo para cada sequência de lote para prever o próximo lote. Finalmente, a estratégia *ativa* comparou a sequência do lote de treinamento atual com a anterior para detectar se ocorreu um desvio. No caso de não haver desvio, opta-se pelo modelo previamente treinado. Caso contrário, um novo modelo é treinado usando a sequência de lote de treinamento atual. A previsão do próximo lote é feita a partir do modelo escolhido.

Para estratégias *ativa*, implementamos três métodos de detecção com base na análise de distribuição de dados. Eles avaliam a ocorrência de desvios conforme a proporção de atrasos ($p(Y)$) apresentados em duas sequências em lote consecutivas (D_i e D_{i-1}). Os três métodos foram (i) *media*, (ii) *variancia* e (iii) *media/variancia*. Em *media* e *variancia*, respectivamente, a *media* e *variancia* de ambas as sequências de lotes de treinamento foram comparados para uma diferença estatisticamente significativa. Finalmente, em *media/variancia*, uma diferença significativa em *media* ou *variancia* indica um desvio. Para esses testes, a proporção de atrasos foi agregada em semanas e testada para normalidade com os testes de Kolmogorov Smirnov e Shapiro-Wilk [Yap and Sim, 2011]. Para distribuições normais, o teste t médio foi usado, e o teste F foi usado para a *variancia*. Quando as distribuições não eram normais, o teste de Wilcoxon foi usado para *media* e o teste de Levene foi usado para *variancia*. O valor de p utilizado foi de 0,05 para todos os testes.

A saída da ação desvio é atribuída a *desvioDetectado*, indicando se o treinamento é necessário no momento t. Se necessário, um método de aprendizado de máquina é treinado usando a sequência em lote D_i (linha 7). A saída é um modelo treinado (*mdl*). O modelo é armazenado para *aeroporto*, *mlm*, *dd* e *dh* selecionados. Se nenhum treinamento for necessário, o modelo previamente treinado *mdl* é recuperado para o *aeroporto*, *mlm*, *dd* e *dh* selecionados.

A linha 12 seleciona o lote para teste (D_{t+1}). O modelo *mdl* é usado para prever atrasos. A metodologia retorna os resultados da previsão. Os experimentos executados com a lógica do Algoritmo 5 são descritos na Seção A.2.

A.2 Experimento

Considerando a metodologia geral descrita no Algoritmo 5 e os parâmetros descritos na Tabela A.1, o Algoritmo 6 descreve toda a análise de desvio de conceito. Ele executa a metodologia considerando o produto cruzado para todos os valores possíveis dos parâmetros. O processo de mineração de dados foi implementado em R para os métodos de pré-processamento e aprendizado de máquina [Han et al., 2012; James et al., 2021]. Eles estão disponíveis como pacotes R (caret, nnet, randomForest, e1071, dplyr, PerformanceAnalysis).

Primeiro, ocorre a seleção dos dez maiores aeroportos do país. São carregados os três modelos de aprendizado de máquina previamente treinados. Modelos usando Naive Bayes (NB), Neural Networks (NN) e Random Forest (RF). Foi selecionado o período compreendido entre 2003 e 2017, e na sequência os três tamanhos de janelas para os lotes. Foram listados os três tipos de aferição de distância para a detecção de desvio ativa, *media*, *variancia* e *media/variancia*. Assim como as três formas de comparação, *linhadebase*, *passiva*, *ativa*. Todos os parâmetros são utilizados de forma cruzada para processamento e posterior comparação.

Algoritmo 6 Experimento

função $analise_desvio(D)$
    $A \leftarrow \{nulo, SBBR, SBPA, SBSV, SBGL, SBCT, SBKP, SBGR, SBCF, SBRJ, SBSP\}$
    $MLM \leftarrow \{NB, NN, RF\}$
    $T \leftarrow \{2003, \cdots, 2017\}$
    $B \leftarrow \{1, 2, 3\}$
    $DD \leftarrow \{media, variancia, media/variancia\}$
    $DH \leftarrow \{linhadebase, passiva, ativa\}$
    $resultados \leftarrow nulo$
    para each $a \in A, mlm \in MLM, t \in T, b \in B, dd \in DD, dh \in DH \leftarrow$
        até faça$resultados \leftarrow resultados \cup metodologia(D, a, mlm, t, b, dd, dh)$
    fim para
    return $results$
fim função

A.3 Resultados

Considerando os métodos *media* e *media/variancia*, o número médio de desvios detectados ao longo de todo o período foi de 9,33 e 9,66, respectivamente. Esses números correspondem a 62,2% e 64% de todos os desvios possíveis, indicando uma alta prevalência de desvios entre os lotes. Além disso, o método *variancia* apresentou uma média de 3,66 (24%), o que indica que este método é menos sensível a desvios em atrasos de voos.

Os resultados de experimentos feitos em níveis de aeroportos e em níveis de malha aérea mostraram similaridade, tanto nos percentuais de desvio detectados, como nas diferenças entre métodos de detecção. O mesmo pôde ser verificado sobre a estratégia de detecção de desvio.

As estratégias *ativa* e *passiva* apresentaram melhor desempenho do que a *linhadebase*, esta última sem retreinamento. Especificamente, a estratégia *passiva* se mostrou a melhor em diversos casos, exatamente onde se tinha uma detecção de desvio muito frequente, o que acabou influenciando o modelo e indicadores de desempenho. Mas as estratégias *ativas* tenderam a balancear indicadores de desempenho e acurácia em demais cenários. Esses resultados indicam que os modelos de aprendizado de máquina de retreinamento (sejam *ativa* ou *passiva*) podem frequentemente oferecer modelos melhores do que o treinamento apenas uma vez no início, como no método

linhadebase.

Os experimentos também mostraram que o número de desvios pode estar relacionado ao desempenho de previsão. Especificamente, o número de desvios do método de detecção de desvio com base na comparação *media* mostrou uma correlação positiva significativa com *precisao*. Os desvios do método *media/variancia* também mostrou correlação com *precisao*, o que era esperado considerando a alta correlação com o método *media*. Esses resultados indicam que quanto maior a frequência de desvios detectados por *media*, maiores são as pontuações de *precisao*.

A.4 Discussão e direcionamento

Conforme as pesquisas anteriores, experimentos e resultados, concluímos que:

- O estudo mostrou ser possível a aplicação de métodos de detecção de Desvios de Conceito em modelos de predição de atrasos em voos;
- O estudo e aplicações de métodos de detecção de desvios de conceito geram diferencial positivo nos resultados de modelos de predição de atrasos de voo;
- Mostrou ser possível a detecção de desvio através da avaliação de diferenças nas distribuições de entrada, sem o processamento do modelo; e
- A princípio, houve similaridade entre os resultados quando aplicados em dimensões diferentes de dados, notadamente as dimensões aeroporto e malha aérea.

O que não foi contemplado nas pesquisas e experimentos até o momento:

- Comparação com demais métodos do estado da arte na detecção de desvio;
- Experimentos utilizando métodos do estado da arte na detecção de desvio;
- Diferentes escalas de tempo, considerando diferentes janelas, como em meses ou semanas; e
- Diferentes parâmetros para delimitação do que se considera desvio.

Diante dos resultados, mostrou-se ser possível a aplicação de métodos de detecção de desvio de conceito em modelos de predição de atrasos em voos. Também de que a avaliação de diferenças nas distribuições de entrada, sem o processamento do modelo, poderia ser um modelo válido para detecção de desvios. E de que todas as aplicações de métodos para detecção de desvio, sejam passivos ou ativos, geraram impacto positivo nos resultados. Dessa forma, decidimos por dar continuidade à pesquisa. Embora não seja exaustiva, como o próprio estudo apontou, as pesquisas prévias serviram para mostrar a validade e viabilidade da pesquisa e aplicação desta disciplina na área de aviação, na complementação de modelos de predição de voos, tornando-os mais robustos.

Com o aprofundamento da pesquisa foi possível identificar e classificar o que identificamos aqui como o estado da arte na detecção de desvio, conforme detalhado no Capítulo III. Verificamos também que a pesquisa carecia de diferentes bases de comparação. E que poderia ser enriquecida com melhor seleção de parâmetros e análise de diferentes períodos para janelas. Consideramos que a evolução na pesquisa deveria contemplar experimentos e comparações com demais métodos do estado da arte na detecção de desvio. Para robustecer as hipótese e manter uma análise abrangente paramétrica com demais pesquisas e publicações, utilizar e ponderar sobre diferentes escalas de tempo, considerando diferentes janelas, como em meses ou semanas.

Objetivos de continuidade da pesquisa em tempo de qualificação:

- Implementar novo método de detecção e tratamento de Desvio de Conceito conforme Capítulo V;
- Comparar com estado da arte, em termos de desempenho e tratamento dos diferentes tipos de desvio de conceito; e
- Utilizar e ponderar sobre diferentes escalas de tempo, considerando diferentes janelas, como em meses ou semanas.

Referências

Abdel-Aty, M., Lee, C., Bai, Y., Li, X., and Michalak, M. Detecting periodic patterns of arrival delay. *Journal of Air Transport Management*, 13(6):355–361, 2007.

Abdelghany, K., Shah, S., Raina, S., and Abdelghany, A. A model for projecting flight delays during irregular operation conditions. *Journal of Air Transport Management*, 10(6):385–394, 2004.

Ai, Y., Pan, W., Yang, C., Wu, D., and Tang, J. A deep learning approach to predict the spatial and temporal distribution of flight delay in network. *Journal of Intelligent and Fuzzy Systems*, 37(5):6029–6037, 2019.

Alloghani, M., Al-Jumeily, D., Mustafina, J., Hussain, A., and Aljaaf, A. J. A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science. In Berry, M. W., Mohamed, A., and Yap, B. W., editors, *Supervised and Unsupervised Learning for Data Science*, Unsupervised and Semi-Supervised Learning, pages 3–21. Springer International Publishing, Cham, 2020.

Alonso, H. and Loureiro, A. Predicting flight departure delay at porto airport: A preliminary study. In *IJCCI 2015 - Proceedings of the 7th International Joint Conference on Computational Intelligence*, volume 3, pages 93–98, 2015.

Amari, S. *Information Geometry and Its Applications*. Springer, 2016.

ANAC. The Brazilian National Civil Aviation Agency. Technical report, http://www.anac.gov.br/, 2021.

ANAC. Voo Regular Ativo (VRA). Technical report, https://sistemas.anac.gov.br/dadosabertos/Voos/, 2023.

Baena-Garcia, M., del Campo-Ávila, J., Fidalgo, R., Bifet, A., Gavalda, R., and Morales-Bueno, R. Early drift detection method. In *Fourth international workshop on knowledge discovery from data streams*, volume 6, pages 77–86. Citeseer, 2006.

Ball, M., Barnhart, C., Nemhauser, G., and Odoni, A. Chapter 1 Air Transportation: Irregular Operations and Control. *Handbooks in Operations Research and Management Science*, 14(C):1–67, 2007.

Barros, R., Cabral, D., Gonçalves P.M., J., and Santos, S. RDDM: Reactive drift detection method. *Expert Systems with Applications*, 90:344–355, 2017.

Basterrech, S. and Woźniak, M. Tracking changes using Kullback-Leibler divergence for the continual learning, 2022.

Bentkus, V. On Hoeffding's inequalities. *Annals of Probability*, 32(2):1650–1673, 2004.

Bifet, A. and Gavaldà, R. Learning from time-changing data with adaptive windowing. In *Proceedings of the 7th SIAM International Conference on Data Mining*, pages 443–448, 2007.

Bifet, A. and Gavaldà, R. Adaptive learning from evolving data streams. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5772 LCNS:249–260, 2009.

Bifet, A., Holmes, G., Kirkby, R., and Pfahringer, B. MOA: Massive Online Analysis. *Journal of Machine Learning Research*, 11:1601–1604, 2010.

Bisong, E. *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*. Apress, 2019.

Breiman, L. Random forests. *Machine Learning*, 45(1):5–32, 2001.

Britto, R., Dresner, M., and Voltes, A. The impact of flight delays on passenger demand and societal welfare. *Transportation Research Part E: Logistics and Transportation Review*, 48(2):460–469, 2012.

BTS. The Bureau of Transportation Statistics databases. Technical report, http://www.rita.dot.gov/bts/home, 2019.

Carvalho, L. Kairos. Technical report, https://github.com/leonardocarvalho77/Kairos, 2023.

Carvalho, L., Sternberg, A., Maia Gonçalves, L., Beatriz Cruz, A., Soares, J., Brandão, D., Carvalho, D., and Ogasawara, E. On the relevance of data science for flight delay research: a systematic review. *Transport Reviews*, 41(4):499–528, 2021.

Ceravolo, P., Tavares, G., Junior, S., and Damiani, E. Evaluation Goals for Online Process Mining: A Concept Drift Perspective. *IEEE Transactions on Services Computing*, 15(4):2473–2489, 2022.

Chakrabarty, N. A data mining approach to flight arrival delay prediction for American airlines. In *IEMECON 2019 - 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference*, pages 102–107, 2019.

Chandramouleeswaran, K. R. and Tran, H. T. Data-driven resilience quantification of the US Air transportation network. In *12th Annual IEEE International Systems Conference, SysCon 2018 - Proceedings*, pages 1–7, 2018.

Chen, H., Wang, J., and Yan, X. A fuzzy support vector machine with weighted margin for flight delay early warning. In *Proceedings - 5th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2008*, volume 3, pages 331–335, 2008.

Cheng, J., Rong, C., Ye, H., and Zheng, X. Risk management using big real time data. In *Proceedings - IEEE 7th International Conference on Cloud Computing Technology and Science, CloudCom 2015*, pages 542–547, 2016.

Cheng, J. and Xing, Y. A positive research on flight delays based on data mining. In *ICTIS 2011: Multimodal Approach to Sustained Transportation System Development - Information, Technology, Implementation - Proceedings of the 1st Int. Conf. on Transportation Information and Safety*, pages 2166–2173, 2011.

Choi, S., Kim, Y., Briceno, S., and Mavris, D. Prediction of weather-induced airline delays based on machine learning algorithms. In *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, volume 2016-December, 2016.

Cover, T. M. and Thomas, J. A. *Elements of Information Theory.* John Wiley & Sons, 2012.

Dai, X., Hu, M., Tian, W., and Liu, H. Modeling Congestion Propagation in Multistage Schedule within an Airport Network. *Journal of Advanced Transportation*, 2018, 2018.

Dami, S. and Yahaghizadeh, M. Efficient event prediction in an IOT environment based on LDA model and support vector machine. In *2018 6th Iranian Joint Congress on Fuzzy and Intelligent Systems, CFIS 2018*, volume 2018-January, pages 135–138, 2018.

de Lima e Silva, P., Severiano C.A., J., Alves, M., Silva, R., Weiss Cohen, M., and Guimarães, F. Forecasting in non-stationary environments with fuzzy time series. *Applied Soft Computing Journal*, 97, 2020.

De Vito, S., Esposito, E., Castell, N., Schneider, P., and Bartonova, A. On the robustness of field calibration for smart air quality monitors. *Sensors and Actuators, B: Chemical*, 310, 2020.

Deshpande, V. and Arikan, M. The impact of airline flight schedules on flight delays. *Manufacturing and Service Operations Management*, 14(3):423–440, 2012.

Diana, T. Predicting arrival delays: An application of spatial analysis. *Journal of Aircraft*, 48(2):462–467, 2011.

Diebold, F. *Elements of Forecasting*. Cengage Learning, 2006.

Ding, Y. Predicting flight delay based on multiple linear regression. In *IOP Conference Series: Earth and Environmental Science*, volume 81, 2017.

DIsabato, S. and Roveri, M. Learning Convolutional Neural Networks in presence of Concept Drift. In *Proceedings of the International Joint Conference on Neural Networks*, volume 2019-July, 2019.

Dong, X., Yu, Z., Cao, W., Shi, Y., and Ma, Q. A survey on ensemble learning. *Frontiers of Computer Science*, 14(2):241–258, 2020.

Dornadula, V. N. and Geetha, S. Credit Card Fraud Detection using Machine Learning Algorithms. In *Procedia Computer Science*, volume 165 of *2nd International Conference on Recent Trends in Advanced Computing ICRTAC -DISRUP - TIV INNOVATION , 2019 November 11-12, 2019*, pages 631–641, 2019.

Dos Reis, D., Flach, P., Matwin, S., and Batista, G. Fast unsupervised online drift detection using incremental kolmogorov-smirnov test. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 13-17-August-2016, pages 1545–1554, 2016.

Du, W.-B., Zhang, M.-Y., Zhang, Y., Cao, X.-B., and Zhang, J. Delay causality network in air transport systems. *Transportation Research Part E: Logistics and Transportation Review*, 118:466–476, 2018.

Duda, P., Jaworski, M., Cader, A., and Wang, L. On Training Deep Neural Networks Using a Streaming Approach. *Journal of Artificial Intelligence and Soft Computing Research*, 10(1):15–26, 2020.

EUROCONTROL. European Organisation for the Safety of Air Navigation. Technical report, https://www.eurocontrol.int/, 2021.

FAA. Federal Aviation Administration. Technical report, http://www.faa.gov/, 2017.

Feuser Fernandes, H. and Müller, C. Optimization of the waiting time and makespan in aircraft departures: A real time non-iterative sequencing model. *Journal of Air Transport Management*, 79, 2019.

Fleurquin, P., Ramasco, J., and Eguiluz, V. Data-driven modeling of systemic delay propagation under severe meteorological conditions. In *Proceedings of the 10th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2013*, 2013.

Fleurquin, P., Ramasco, J. J., and Eguíluz, V. M. Characterization of delay propagation in the US air-transportation network. *Transportation Journal*, 53(3):330–344, 2014.

Frías-Blanco, I., Del Campo-Ávila, J., Ramos-Jiménez, G., Morales-Bueno, R., Ortiz-Díaz, A., and Caballero-Mota, Y. Online and non-parametric drift detection methods based on Hoeffding's bounds. *IEEE Transactions on Knowledge and Data Engineering*, 27(3):810–823, 2015.

Gama, J., Medas, P., Castillo, G., and Rodrigues, P. Learning with drift detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3171:286–295, 2004.

Gama, J., Zliobaite, I., Bifet, A., Pechenizkiy, M., and Bouchachia, A. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4), 2014.

Garcia, K. D., Poel, M., Kok, J. N., and de Carvalho, A. C. P. L. F. Online clustering for novelty detection and concept drift in data streams. In Moura Oliveira, P., Novais, P., and Reis, L. P., editors, *Progress in Artificial Intelligence*, pages 448–459, Cham. Springer International Publishing, 2019.

García, S., Ramírez-Gallego, S., Luengo, J., Benítez, J. M., and Herrera, F. Big data preprocessing: methods and prospects. *Big Data Analytics*, 1(1):9, 2016.

Giusti, L., Carvalho, L., Gomes, A., Coutinho, R., Soares, J., and Ogasawara, E. Analyzing flight delay prediction under concept drift. *Evolving Systems*, 13(5):723–736, 2022.

Google. Google colaboratory. Technical report, https://colab.research.google.com/, 2023.

Gui, G., Liu, F., Sun, J., Yang, J., Zhou, Z., and Zhao, D. Flight delay prediction based on aviation big data and machine learning. *IEEE Transactions on Vehicular Technology*, 69(1):140–150, 2020.

Han, J., Kamber, M., and Pei, J. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2012.

Hastie, T., Tibshirani, R., and Friedman, J. H. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.

Hoeltgebaum, H., Adams, N., and Fernandes, C. Estimation, Forecasting, and Anomaly Detection for Nonstationary Streams Using Adaptive Estimation. *IEEE Transactions on Cybernetics*, 52(8):7956–7967, 2022.

Hoens, T., Polikar, R., and Chawla, N. Learning from streaming data with concept drift and imbalance: An overview. *Progress in Artificial Intelligence*, 1(1):89–101, 2012.

IATA. International Air Transport Association. Technical report, https://www.iata.org/, 2021.

IEM, I. S. U. Iowa Environmental Mesonet. Technical report, https://mesonet.agron.iastate.edu/request/download.phtml, 2023.

Iwashita, A. and Papa, J. An Overview on Concept Drift Learning. *IEEE Access*, 7:1532–1547, 2019.

James, G. M., Witten, D., Hastie, T., and Tibshirani, R. *An Introduction to Statistical Learning: With Applications in R*. Springer Nature, 2021.

Jedrzejowicz, J. and Jedrzejowicz, P. GEP-based classifiers with drift-detection. *Expert Systems*, 2020.

Jiang, J., Liu, F., Ng, W., Tang, Q., Wang, W., and Pham, Q.-V. Dynamic Incremental Ensemble Fuzzy Classifier for Data Streams in Green Internet of Things. *IEEE Transactions on Green Communications and Networking*, 6(3):1316–1329, 2022.

Khamassi, I. and Sayed-Mouchaweh, M. Drift detection and monitoring in non-stationary environments. In *2014 IEEE Conference on Evolving and Adaptive Intelligent Systems, EAIS 2014 - Conference Proceedings*, 2014.

Khanmohammadi, S., Chou, C.-A., Lewis H.W., I., and Elias, D. A systems approach for scheduling aircraft landings in JFK airport. In *IEEE International Conference on Fuzzy Systems*, pages 1578–1585, 2014.

Kim, Y., Choi, S., Briceno, S., and Mavris, D. A deep learning approach to flight delay prediction. In *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, volume 2016-December, 2016.

Krishnan, R., Jagannathan, S., and Samaranayake, V. A Minimax Approach for Classification with Big-data. In *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, pages 1437–1444, 2019.

Kullback, S. *Information Theory and Statistics*. Courier Corporation, 2012.

Kuncheva, L. Change detection in streaming multivariate data using likelihood detectors. *IEEE Transactions on Knowledge and Data Engineering*, 25(5):1175–1180, 2013.

Le, T., Sabir, B., and Babar, M. Automated software vulnerability assessment with concept drift. In *IEEE International Working Conference on Mining Software Repositories*, volume 2019-May, pages 371–382, 2019.

Lee, D. D. and Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Li, X., Zhou, Y., Jin, Z., Yu, P., and Zhou, S. A Classification and Novel Class Detection Algorithm for Concept Drift Data Stream Based on the Cohesiveness and Separation Index of Mahalanobis Distance. *Journal of Electrical and Computer Engineering*, 2020, 2020a.

Li, Z., Huang, W., Xiong, Y., Ren, S., and Zhu, T. Incremental learning imbalanced data streams with concept drift: The dynamic updated ensemble algorithm. *Knowledge-Based Systems*, 195, 2020b.

Lin, C.-C., Deng, D.-J., Kuo, C.-H., and Chen, L. Concept drift detection and adaption in big imbalance industrial IoT data using an ensemble learning method of offline classifiers. *IEEE Access*, 7:56198–56207, 2019.

Liu, S., Xue, S., Wu, J., Zhou, C., Yang, J., Li, Z., and Cao, J. Online Active Learning for Drifting Data Streams. *IEEE Transactions on Neural Networks and Learning Systems*, 34(1):186–200, 2023.

Liu, Y. and Willsky, A. S. Learning Gaussian Graphical Models with observed or latent FVSs. In *Advances in Neural Information Processing Systems*, 2013.

Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., and Zhang, G. Learning under Concept Drift: A Review. *IEEE Transactions on Knowledge and Data Engineering*, 31(12):2346–2363, 2019.

Mahdi, O., Pardede, E., Ali, N., and Cao, J. Diversity measure as a new drift detection method in data streaming. *Knowledge-Based Systems*, 191, 2020.

McDiarmid, C. On the method of bounded differences. In Siemons, J., editor, *Surveys in Combinatorics, 1989: Invited Papers at the Twelfth British Combinatorial Conference*, London Mathematical Society Lecture Note Series, pages 148–188. Cambridge University Press, Cambridge, 1989.

Miller, C. R. Opportunity, opportunism, and progress:Kairos in the rhetoric of technology. *Argumentation*, 8(1):81–96, 1994.

Mitchell, T. M. *Machine Learning*. McGraw-Hill, 1997.

Moreira, L., Dantas, C., Oliveira, L., Soares, J., and Ogasawara, E. On Evaluating Data Preprocessing Methods for Machine Learning Models for Flight Delays. In *Proceedings of the International Joint Conference on Neural Networks*, volume 2018-July, 2018.

Munoz Hernandez, A., Scarlatti, D., and Costas, P. Real-Time Estimated Time of Arrival Prediction System using Historical Surveillance Data. In *Proceedings - 45th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2019*, pages 174–177, 2019.

Namitha, K. and Santhosh Kumar, G. Learning in the presence of concept recurrence in data stream clustering. *Journal of Big Data*, 7(1), 2020.

Nayak, N. and Zhang, Y. Estimation and comparison of impact of single airport delay on national airspace system with multivariate simultaneous models. *Transportation Research Record*, 1(2206):52–60, 2011.

Ng, W., Zhang, J., Lai, C., Pedrycz, W., Lai, L., and Wang, X. Cost-sensitive weighting and imbalance-reversed bagging for streaming imbalanced and concept drifting in electricity pricing classification. *IEEE Transactions on Industrial Informatics*, 15(3):1588–1597, 2019.

Nigam, R. and Govinda, K. Cloud based flight delay prediction using logistic regression. In *Proceedings of the International Conference on Intelligent Sustainable Systems, ICISS 2017*, pages 662–667, 2018.

Ogasawara, E. Data Analytics Lab (DAL). Technical report, https://eic.cefet-rj.br/~dal/brazilian-flight-dataset-description/, 2023.

Page, E. S. Continuous Inspection Schemes. *Biometrika*, 41(1/2):100–115, 1954.

Pears, R., Sakthithasan, S., and Koh, Y. Detecting concept change in dynamic data streams: A sequential approach based on reservoir sampling. *Machine Learning*, 97(3):259–293, 2014.

Pesaranghader, A. and Viktor, H. Fast hoeffding drift detection method for evolving data streams. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9852 LNAI:96–111, 2016.

Pesaranghader, A., Viktor, H., and Paquet, E. McDiarmid Drift Detection Methods for Evolving Data Streams. In *Proceedings of the International Joint Conference on Neural Networks*, volume 2018-July, 2018.

Post, J., Murphy, D., and Bonn, J. A regression model of national airspace system delay. In *Collection of Technical Papers - 6th AIAA Aviation Technology, Integration, and Operations Conference*, volume 1, pages 441–449, 2006.

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M., Shyu, M.-L., Chen, S.-C., and Iyengar, S. A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys*, 51(5), 2018.

Python Software Foundation, P. Python programming language. Technical report, https://www.python.org, 2023.

R Core Team, R. C. T. The R project for statistical computing. Technical report, https://www.r-project.org, 2023.

Raab, C., Heusinger, M., and Schleif, F.-M. Reactive Soft Prototype Computing for Concept Drift Streams. *Neurocomputing*, 416:340–351, 2020.

Ramírez-Gallego, S., Krawczyk, B., García, S., Woźniak, M., and Herrera, F. A survey on data preprocessing for data stream mining: Current status and future directions. *Neurocomputing*, 239:39–57, 2017.

Rebollo, J. and Balakrishnan, H. Characterization and prediction of air traffic delays. *Transportation Research Part C: Emerging Technologies*, 44:231–241, 2014.

Rodríguez-Sanz, A., Comendador, F., Valdés, R., Pérez-Castán, J., Montes, R., and Serrano, S. Assessment of airport arrival congestion and delay: Prediction and reliability. *Transportation Research Part C: Emerging Technologies*, 98:255–283, 2019.

Rodriguez-Sanz, A., Comendador, F. G., Valdes, R. A., and Perez-Castan, J. A. Characterization and prediction of the airport operational saturation. *Journal of Air Transport Management*, 69:147–172, 2018.

Rong, F., Qianya, L., Bo, H., Jing, Z., and Dongdong, Y. The prediction of flight delays based the analysis of Random flight points. In *Chinese Control Conference, CCC*, volume 2015-September, pages 3992–3997, 2015.

Ross, G., Adams, N., Tasoulis, D., and Hand, D. Exponentially weighted moving average charts for detecting concept drift. *Pattern Recognition Letters*, 33(2):191–198, 2012.

Saadallah, A., Moreira-Matias, L., Sousa, R., Khiari, J., Jenelius, E., and Gama, J. BRIGHT - Drift-Aware Demand Predictions for Taxi Networks. *IEEE Transactions on Knowledge and Data Engineering*, 32(2):234–245, 2020.

Sathanur, A., Sripimonwan, B., Halappanavar, M., Chatterjee, S., Ganguly, A., and Clark, K. Identification of Critical Airports from the Perspective of Delay and Disruption Propagation in Air Travel Networks. In *2019 IEEE International Symposium on Technologies for Homeland Security, HST 2019*, 2019.

Sathya, R. and Abraham, A. Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. *International Journal of Advanced Research in Artificial Intelligence (IJARAI)*, 2(2), 2013.

Sato, D., De Freitas, S., Barddal, J., and Scalabrin, E. A Survey on Concept Drift in Process Mining. *ACM Computing Surveys*, 54(9), 2022.

Scarpel, R. A. and Pelicioni, L. A data analytics approach for anticipating congested days at the São Paulo International Airport. *Journal of Air Transport Management*, 72:1–10, 2018.

Schmidhuber, J. Deep Learning in neural networks: An overview. *Neural Networks*, 61:85–117, 2015.

Schultz, M. and Reitmann, S. Machine learning approach to predict aircraft boarding. *Transportation Research Part C: Emerging Technologies*, 98:391–408, 2019a.

Schultz, M. and Reitmann, S. Prediction of aircraft boarding time using LSTM network. In *Proceedings - Winter Simulation Conference*, volume 2018-December, pages 2330–2341, 2019b.

Sebastião, R. and Gama, J. Change detection in learning histograms from data streams. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4874 LNAI:112–123, 2007.

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*, volume 9781107057135. Cambridge University Press, 2013.

Song, Y., Lu, J., Lu, H., and Zhang, G. Fuzzy Clustering-Based Adaptive Regression for Drifting Data Streams. *IEEE Transactions on Fuzzy Systems*, 28(3):544–557, 2020.

Song, Y., Zhang, G., Lu, H., and Lu, J. A Noise-tolerant Fuzzy c-Means based Drift Adaptation Method for Data Stream Regression. In *IEEE International Conference on Fuzzy Systems*, volume 2019-June, 2019.

Sternberg, A., Carvalho, D., Murta, L., Soares, J., and Ogasawara, E. An analysis of Brazilian flight delays based on frequent patterns. *Transportation Research Part E: Logistics and Transportation Review*, 95:282–298, 2016.

Teixeira, C., Tavares, L., Soares, J., dos Santos, J., Amorim, G., and Ogasawara, E. Brazilian Flights Dataset. Technical report, https://doi.org/10.21227/k10b-qn21, 2020.

Tejasviram, V., Solanki, H., Ravi, V., and Kamaruddin, S. Auto associative Extreme Learning Machine based non-linear principal component regression for big data applications. In *The 10th*

International Conference on Digital Information Management, ICDIM 2015, pages 223–228, 2016.

Thiagarajan, B., Srinivasan, L., Sharma, A. V., Sreekanthan, D., and Vijayaraghavan, V. A machine learning approach for prediction of on-time performance of flights. In *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, volume 2017-September, 2017.

Tian, H., Khoa, N., Anaissi, A., Wang, Y., and Chen, F. Concept drift adaption for online anomaly detection in structural health monitoring. In *International Conference on Information and Knowledge Management, Proceedings*, pages 2813–2821, 2019.

Tu, Y., Ball, M., and Jank, W. Estimating flight departure delay distributions - A statistical approach with long-term trend and short-term pattern. *Journal of the American Statistical Association*, 103(481):112–125, 2008.

Wang, C., Chen, M.-H., Schifano, E., Wu, J., and Yan, J. Statistical methods and computing for big data. *Statistics and its Interface*, 9(4):399–414, 2016.

Wang, K., Li, J., and Tian, Y. Airport Delay Prediction Method based on Improved Weather Impacted Traffic Index. In *Proceedings of 2019 IEEE 1st International Conference on Civil Aviation Safety and Information Technology, ICCASIT 2019*, pages 73–78, 2019.

Wang, K., Lu, J., Liu, A., Song, Y., Xiong, L., and Zhang, G. Elastic gradient boosting decision tree with adaptive iterations for concept drift adaptation. *Neurocomputing*, 491:288–304, 2022.

Wang, X., Kang, Q., Zhou, M., and Yao, S. A Multiscale Concept Drift Detection Method for Learning from Data Streams. In *IEEE International Conference on Automation Science and Engineering*, volume 2018-August, pages 786–790, 2018.

Webb, G., Hyde, R., Cao, H., Nguyen, H., and Petitjean, F. Characterizing concept drift. *Data Mining and Knowledge Discovery*, 30(4):964–994, 2016.

Wieland, F. Limits to growth: results from the detailed policy assessment tool [air traffic congestion]. In *16th DASC. AIAA/IEEE Digital Avionics Systems Conference. Reflections to the Future. Proceedings*, volume 2, pages 9.2–1, 1997.

Woodburn, A. and Ryerson, M. Airport capacity enhancement and flight predictability. *Transportation Research Record*, 1(2400):87–97, 2014.

Wu, C.-L. and Truong, T. Improving the IATA delay data coding system for enhanced data analytics. *Journal of Air Transport Management*, 40:78–85, 2014.

Xu, N., Donohue, G., Laskey, K., and Chen, C.-H. Estimation of delay propagation in the national aviation system using Bayesian networks. In *Proceedings of the 6th USA/Europe Air Traffic Management Research and Development Seminar, ATM 2005*, pages 353–363, 2005.

Xu, N., Sherry, L., and Laskey, K. B. Multifactor model for predicting delays at U.S. airports. *Transportation Research Record*, 1(2052):62–71, 2008.

Xu, X., Yuan, H., and Qian, Y. Analyzing the system features of the flight delays: A network perspective. In *2015 12th International Conference on Service Systems and Service Management, ICSSSM 2015*, 2015.

Yang, Z., Al-Dahidi, S., Baraldi, P., Zio, E., and Montelatici, L. A Novel Concept Drift Detection Method for Incremental Learning in Nonstationary Environments. *IEEE Transactions on Neural Networks and Learning Systems*, 31(1):309–320, 2020.

Yanying, Y., Mo, H., and Haifeng, L. A Classification Prediction Analysis of Flight Cancellation Based on Spark. In *Procedia Computer Science*, volume 162, pages 480–486, 2019.

Yap, B. and Sim, C. Comparisons of various types of normality tests. *Journal of Statistical Computation and Simulation*, 81(12):2141–2155, 2011.

Yu, B., Guo, Z., Asian, S., Wang, H., and Chen, G. Flight delay prediction for commercial air transport: A deep learning approach. *Transportation Research Part E: Logistics and Transportation Review*, 125:203–221, 2019.

Zhang, M. and He, C. Survey on association rules mining algorithms. *Lecture Notes in Electrical Engineering*, 56 LNEE:111–118, 2010.

Zonglei, L., Jiandong, W., and Yunfeng, Z. A new technology for combining small samples based on clustering and its applications. In *Proceedings - 2008 International Symposium on Knowledge Acquisition and Modeling, KAM 2008*, pages 735–740, 2008.