



DETECCÃO DE ADULTERAÇÃO DE IDENTIFICADOR EM DADOS AIS: UM MÉTODO UTILIZANDO PREVISÃO DE TRAJETÓRIAS E CLASSIFICAÇÃO

Julio Cesar Medeiros dos Anjos Ramos

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientador: Rosa Maria Meri Leão

Rio de Janeiro

Abril de 2025

DETECÇÃO DE ADULTERAÇÃO DE IDENTIFICADOR EM DADOS AIS: UM
MÉTODO UTILIZANDO PREVISÃO DE TRAJETÓRIAS E CLASSIFICAÇÃO

Julio Cesar Medeiros dos Anjos Ramos

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO
ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE
ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO
GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E
COMPUTAÇÃO.

Orientador: Rosa Maria Meri Leão

Aprovada por: Prof. Rosa Maria Meri Leão
Prof. Claudio Miceli de Farias
Prof. Daniel Sadoc Menasche

RIO DE JANEIRO, RJ – BRASIL
ABRIL DE 2025

Medeiros dos Anjos Ramos, Julio Cesar

Detecção de adulteração de identificador em dados AIS: um método utilizando previsão de trajetórias e classificação/Julio Cesar Medeiros dos Anjos Ramos. – Rio de Janeiro: UFRJ/COPPE, 2025.

XIV, 61 p.: il.; 29,7cm.

Orientador: Rosa Maria Meri Leão

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2025.

Referências Bibliográficas: p. 48 – 49.

1. Primeira palavra-chave. 2. Segunda palavra-chave.
3. Terceira palavra-chave. I. Meri Leão, Rosa Maria.
- II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Agradecimentos

À minha orientadora, Prof. Dra. Rosa, pela confiança depositada e por aceitar o desafio de me ter como aluno. Sou imensamente grato por seus ensinamentos, orientações e pelo apoio ao longo de toda a jornada do mestrado. Agradeço, também, pela paciência diante das minhas falhas e pelo incentivo constante ao aprimoramento acadêmico e profissional.

Aos Professores Miceli e Daniel Sadoc, minha sincera gratidão pela disponibilidade em compor a banca da minha defesa e pelas valiosas sugestões que enriqueceram este trabalho.

Ao Prof. Pablo Rangel, pelo apoio fundamental, pelas orientações precisas e pelos ensinamentos que contribuíram significativamente para o desenvolvimento desta pesquisa.

Aos meus pais e ao meu irmão, por toda dedicação, companheirismo e amizade, que sempre foram a base do meu crescimento pessoal e profissional. Um agradecimento especial à minha esposa, pelo apoio incondicional, pelos sábios conselhos, por acreditar em mim mesmo nos momentos de dúvida e pelas palavras de motivação que me impulsionaram a seguir em frente. Sou grato, também, por sua paciência diante das minhas ausências ao longo do curso.

Aos colegas do mestrado, pela amizade e pelo companheirismo, que tornaram essa caminhada mais leve e enriquecedora.

À Marinha do Brasil, pela oportunidade de aprimoramento e pelo suporte fundamental para esta importante etapa da minha carreira.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

DETECÇÃO DE ADULTERAÇÃO DE IDENTIFICADOR EM DADOS AIS: UM MÉTODO UTILIZANDO PREVISÃO DE TRAJETÓRIAS E CLASSIFICAÇÃO

Julio Cesar Medeiros dos Anjos Ramos

Abril/2025

Orientador: Rosa Maria Meri Leão

Programa: Engenharia de Sistemas e Computação

Os sistemas de monitoramento marítimo são amplamente utilizados por autoridades em todo o mundo, com objetivos que vão desde a segurança da navegação até o combate a crimes como o tráfico internacional de drogas. Esses sistemas processam grandes volumes de informações obtidas por diversos sensores e embarcações, o que torna inviável a análise manual por operadores humanos. Por isso, algoritmos computacionais são essenciais para automatizar o processamento e a interpretação desses dados. Na última década, técnicas de aprendizado de máquina têm sido amplamente adotadas para melhorar ainda mais esses sistemas. Cientes da existência do monitoramento, embarcações envolvidas em atividades ilícitas frequentemente adulteram suas identificações ou interrompem temporariamente a transmissão dos dados do sistema *Automatic Identification System* (AIS), visando enganar os sistemas de vigilância e ocultar suas atividades. Neste contexto, este trabalho propõe uma solução baseada em dados AIS para detectar embarcações que adulteraram suas identificações e estimar suas trajetórias reais. A abordagem proposta é dividida em duas etapas principais. Na primeira, utiliza-se um algoritmo de *Deep Learning* para prever as trajetórias das embarcações. Na segunda etapa, algoritmos de classificação (*Random Forest*, SVM e um método desenvolvido pela Marinha do Brasil) são empregados para determinar se trajetórias associadas a identificadores diferentes correspondem, na verdade, a uma mesma embarcação. A solução foi validada usando cinco conjuntos de dados distintos, provenientes de regiões geográficas variadas, permitindo uma avaliação abrangente do desempenho das técnicas propostas.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

IDENTIFIER SPOOFING DETECTION IN AIS DATA: A METHOD USING TRAJECTORY PREDICTION AND CLASSIFICATION

Julio Cesar Medeiros dos Anjos Ramos

April/2025

Advisor: Rosa Maria Meri Leão

Department: Systems Engineering and Computer Science

Maritime monitoring systems are widely used by authorities worldwide for purposes ranging from navigation safety to combating crimes such as international drug trafficking. These systems process large volumes of information collected from various sensors and vessels, making manual analysis by human operators unfeasible. Therefore, computational algorithms are essential for automating the processing and interpretation of this data. Over the past decade, machine learning techniques have been widely adopted to further enhance these systems. Aware of the existence of monitoring, vessels engaged in illicit activities often tamper with their identification or temporarily disable the transmission of data from the Automatic Identification System (AIS) to deceive surveillance systems and conceal their activities. In this context, this work proposes a solution based on AIS data to detect vessels that have altered their identification and estimate their actual trajectories. The proposed approach is divided into two main stages. In the first, a Deep Learning algorithm is used to predict vessel trajectories. In the second stage, classification algorithms (Random Forest, SVM, and a method developed by the Brazilian Navy) are employed to determine whether trajectories associated with different identifiers actually correspond to the same vessel. The solution was validated using five distinct datasets from various geographical regions, allowing for a comprehensive evaluation of the proposed techniques and performance.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xiii
1 Introdução	1
2 Trabalhos relacionados	3
3 Método Proposto	6
3.1 Descrição do problema	6
3.2 Método proposto	7
3.2.1 Técnica para previsão de trajetória: TrAISformer	8
3.2.2 Técnicas de classificação	9
3.2.3 Dados utilizados	11
3.3 Procedimento experimental	20
3.3.1 Treinamento e teste dos modelos de previsão de trajetória	20
3.3.2 Treinamento e teste dos modelos de classificação	21
3.4 Métricas	27
3.4.1 Erro médio	27
3.4.2 Diagrama <i>Boxplot</i>	27
3.4.3 Erro de posição das previsões de trajetórias	28
3.4.4 Erro de velocidade das previsões de trajetórias	28
3.4.5 Erro de rumo das previsões de trajetórias	28
3.4.6 Métricas para etapa de classificação	29
3.5 Implementação	29
4 Resultados e Discussões	31
4.1 Resultados das etapas de previsão de trajetórias	31
4.2 Resultados da etapa de classificação	37
5 Conclusões	46
Referências Bibliográficas	48

Lista de Figuras

3.1	Exemplo de trajetórias	7
3.2	Exemplo de trajetórias em um ambiente real	8
3.3	Exemplo de trajetórias e suas previsões	9
3.4	As imagens 3.4(a), 3.4(b) e 3.4(c) mostram a área geográfica delimitada para coleta de dados AIS para cada um dos conjuntos de dados.	19
3.5	Ilustração da variável GAP.	21
3.6	Amostra de resultados obtidos com as previsões realizadas para o conjunto de dados 1. Os pontos marcados em amarelo nas trajetórias 3.6(b) e 3.6(c) são os 18 pontos de da trajetória original a partir dos quais os modelos realizaram suas previsões.	23
4.1	Resultados dos modelos de previsão de trajetória. As Figuras 4.1(a) e 4.1(b) exibem, no eixo y, o erro médio de posição (EM_k), em Km, sendo o eixo x o instante de tempo da trajetória prevista (equivalente de k em horas), em horas. Os itens 4.1(c) e 4.1(d) exibem o <i>boxplot</i> dos erros de posição, em Km, de cada uma das trajetórias para o instante de previsão de 1 hora ($Err_{i,k=1h}$) sendo o eixo x o número do conjunto de dados e os itens 4.1(e) e 4.1(f) exibem o <i>boxplot</i> dos erros de posição, em Km, de cada uma das trajetórias para o instante de previsão de 3 hora ($Err_{i,k=3h}$) sendo o eixo x o número do conjunto de dados.	34
4.2	Resultados de previsão do rumo dos modelos de previsão de trajetória. Os itens 4.2(a) e 4.2(b) exibem, no eixo y, o erro médio do rumo (EM_k), em graus, sendo o eixo x o instante de tempo da trajetória prevista (equivalente de k em horas), em horas. Os itens 4.2(c) e 4.2(d) exibem o <i>boxplot</i> dos erros de rumo, em graus, de cada uma das trajetórias para o instante de previsão de 1 hora ($Err_{i,k=1h}$) sendo o eixo x o número do conjunto de dados e os itens 4.2(e) e 4.2(f) exibem o <i>boxplot</i> dos erros de rumo, em graus, de cada uma das trajetórias para o instante de previsão de 3 hora ($Err_{i,k=3h}$) sendo o eixo x o número do conjunto de dados.	35

4.3	Resultados de previsão de velocidade dos modelos de previsão de trajetória. Os itens 4.3(a) e 4.3(b) exibem, no eixo y, o erro médio de velocidade (EM_k), em nós, sendo o eixo x o instante de tempo da trajetória prevista (equivalente de k em horas), em horas. Os itens 4.3(c) e 4.3(d) exibem o <i>boxplot</i> dos erros de velocidade, em nós, de cada uma das trajetórias para o instante de previsão de 1 hora ($Err_{i,k=1h}$) sendo o eixo x o número do conjunto de dados e os itens 4.3(e) e 4.3(f) exibem o <i>boxplot</i> dos erros de velocidade, em nós, de cada uma das trajetórias para o instante de previsão de 3 hora ($Err_{i,k=3h}$) sendo o eixo x o número do conjunto de dados.	36
4.4	Matrizes de confusão com os resultados obtidos pelos classificadores com melhor acurácia para cada um dos conjuntos de dados. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	43
4.5	Resultado do experimento de análise do GAP com diferentes algoritmos.	44
4.6	Resultado do experimento de análise do GAP com diferentes pré-processamentos.	45
A.1	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 1 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	51
A.2	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 1 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	52
A.3	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 1 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	52
A.4	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 1 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	53
A.5	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 2 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	53

A.6	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 2 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	54
A.7	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 2 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	54
A.8	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 2 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	55
A.9	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 3 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	55
A.10	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 3 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	56
A.11	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 3 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	56
A.12	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 3 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	57
A.13	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 4 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	57
A.14	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 4 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	58

A.15	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 4 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	58
A.16	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 4 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	59
A.17	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 5 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	59
A.18	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 5 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	60
A.19	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 5 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	60
A.20	Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 5 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos	61

Lista de Tabelas

3.1	Exemplo do funcionamento da janela deslizante considerando uma janela de tamanho 5, sendo D_n um vetor (x,y) com a diferença em x e em y entre as enésimas amostras do par de trajetórias. As posições marcadas em amarelo e verde são as posições que estão dentro da janela e utilizadas no cálculo da matriz de covariância da equação 3.1, sendo a posição verde o ponto D na equação 3.1 no qual está sendo testada a hipótese	12
3.2	Tabela Resumo das características dos conjuntos de dados utilizados.	19
3.3	Tabela resumo de características dos conjuntos de dados para etapa de classificação.	26
3.4	Número de trajetórias para cada um dos conjuntos de dados em cada um dos experimentos com diferentes valores de GAP	26
4.1	Descrição dos tipos de pré-processamentos utilizados nas <i>features</i> extraídas na etapa de classificação	38
4.2	Esta tabela apresenta a acurácia obtida nos modelos de classificação utilizando diferentes pré-processamentos nos dados em cada um dos 5 conjuntos de dados. Demarcados em negrito são os maiores valores de acurácia obtido em cada um dos conjuntos de dados.	39
4.3	Esta tabela apresenta a taxa de verdadeiro positivo obtida nos modelos de classificação utilizando diferentes pré-processamentos nos dados em cada um dos 5 conjuntos de dados. Demarcados em negrito são os maiores valores da taxa de verdadeiro positivo obtido em cada um dos conjuntos de dados.	39
4.4	Esta tabela apresenta a taxa de falso positivo obtida nos modelos de classificação utilizando diferentes pré-processamentos nos dados em cada um dos 5 conjuntos de dados. Demarcados em negrito são os menores valores da taxa de falso positivo obtido em cada um dos conjuntos de dados.	40

4.5 Esta tabela apresenta a taxa de f1-score obtida nos modelos de classificação utilizando diferentes pré-processamentos nos dados em cada um dos 5 conjuntos de dados. Demarcados em negrito são os maiores valores de f1-score obtido em cada um dos conjuntos de dados. . . . 40

Capítulo 1

Introdução

O AIS (*Automatic Identification System*) é um sistema colaborativo de identificação e geolocalização de embarcações, desenvolvido com o objetivo de melhorar a segurança na navegação, sendo utilizado na prevenção de acidentes marítimos. Esse sistema consiste em um equipamento instalado a bordo de navios e transmite continuamente informações estáticas sobre a própria embarcação, como identificação (*Maritime Mobile Service Identity* - MMSI), porto de origem, porto de destino e outras características, e também dados dinâmicos relacionados à cinemática, como posição em coordenadas geográficas, velocidade e rumo (direção). Essas informações são transmitidas via VHF (*Very High Frequency*) de forma pública e podem ser recebidas por outras embarcações que navegam na mesma região. Assim, cada embarcação tem acesso às informações da posição e de navegação das embarcações próximas, permitindo a prevenção de colisões.

Além disso, estações localizadas na costa também podem receber esses dados e disponibilizá-los para serem utilizados em sistemas de monitoramento de tráfego marítimo ou para controle de áreas portuárias, entre outras aplicações. No entanto, a coleta de informações via VHF possui limitações no alcance, que depende das condições climáticas e normalmente varia entre 15 e 20 milhas náuticas (aproximadamente 40 km) [1]. Com o avanço da tecnologia satelital, foi possível lançar satélites capazes de capturar mensagens AIS, ampliando significativamente a cobertura e permitindo o monitoramento global do tráfego marítimo.

A partir dessa evolução, autoridades marítimas ao redor do mundo passaram a utilizar os dados do AIS para uma variedade de finalidades, como monitorar atividades ilegais (por exemplo, pesca clandestina e tráfico de drogas), realizar operações de busca e salvamento, além de outras atividades relacionadas à segurança e controle marítimo.

Em 2004, a Organização Marítima Internacional (IMO) regulamentou o uso obrigatório de dispositivos AIS para navios com arqueação bruta acima de 300 toneladas em viagens internacionais, acima de 500 toneladas em viagens domésticas, além de

todos os navios de passageiros [2]. Apesar dessa regulamentação e da ampla adoção do sistema, a confiabilidade do AIS é frequentemente questionada [3]. Além das possíveis falhas de comunicação ou mau funcionamento dos equipamentos – que podem gerar dados ausentes ou imprecisos –, o fato de ser um sistema colaborativo permite que os dispositivos sejam deliberadamente desligados ou que as informações transmitidas sejam adulteradas para ludibriar os sistemas de monitoramento, muitas vezes com o objetivo de ocultar atividades ilícitas [3]. É comum a detecção, por exemplo, de embarcações utilizando o mesmo identificador simultaneamente, o que indica que alguma das embarcações está com o seu identificador adulterado [4].

O AIS também é suscetível a ataques externos dos tipos negação de GPS ou criação de navios inexistentes, conforme demonstrado em [4], o que representa uma preocupação significativa, especialmente em cenários de conflito, onde a manipulação ou falsificação de dados pode ter consequências graves.

Diante do exposto, este trabalho tem o objetivo de apresentar um método para identificar a continuidade da trajetória de uma embarcação que tenha a sua identificação adulterada. O método proposto envolve uma combinação entre técnicas de *Deep Learning* para realizar previsão de trajetórias de embarcações combinada com técnicas de classificação utilizando dados históricos de navegação e da cinemática da embarcação provenientes do sistema AIS.

As principais contribuições desse trabalho são:

1. Proposta de um método que combina um modelo *deep learning* com algoritmos de classificação para identificar se duas trajetórias pertencem à mesma embarcação.
2. Experimentos com o modelo de *deep learning* para avaliar sua robustez em diferentes cenários, considerando cinco conjuntos de dados distintos, provenientes de regiões geográficas variadas.
3. Avaliação do desempenho de três algoritmos de classificação usados para identificar se duas rotas pertencem à mesma embarcação usando cinco conjuntos de dados.
4. Definição de critérios para limpeza, filtragem e pré-processamentos para a preparação para utilização de conjuntos de dados AIS.

Capítulo 2

Trabalhos relacionados

O sistema AIS (*Automatic Identification System*) apresenta diversas vulnerabilidades conhecidas, que vão desde a falta de controle na instalação dos equipamentos nas embarcações até a ausência de autenticação e verificação de integridade em seus protocolos de comunicação [5]. Essas fragilidades possibilitam diferentes formas de adulteração nos dados transmitidos pelo sistema, como a criação de informações sobre navios inexistentes, a falsificação da posição da embarcação e manipulações mais simples, como o desligamento intencional do equipamento ou a alteração do identificador único da embarcação (*Maritime Mobile Service Identity* - MMSI) [6]. Diante desse cenário, este trabalho busca contribuir para a detecção de adulterações no identificador MMSI realizadas pela tripulação do próprio navio.

Diversas técnicas são comumente empregadas na literatura com o objetivo de detecção de adulteração de dados do sistema. Uma estratégia comum é detectar o desligamento intencional do equipamento AIS. O estudo [7] propõe um método para diferenciar falhas de comunicação de desligamentos intencionais do AIS, caracterizando a área de cobertura de uma estação costeira com base na intensidade do sinal recebido. No entanto, essa abordagem apresenta limitações importantes: (1) é aplicável apenas a sensores AIS costeiros, tornando-se inviável para monitoramento oceânico; e (2) depende da informação da intensidade do sinal, que não costuma estar disponível. O estudo [8] utiliza uma estimativa da velocidade da embarcação com base nas posições recebidas pelo receptor, combinada com o algoritmo DBSCAN, para detectar o desligamento intencional do AIS e a adulteração do identificador MMSI. O método parte da premissa que embarcações que falsificam seus identificadores utilizam um MMSI pertencente a outra embarcação real, geralmente localizada em uma região geograficamente distante. Assim, ao calcular a velocidade com base nas diferenças de posição entre amostras, valores anormalmente altos e incompatíveis com uma embarcação real, indicam uma possível falsificação. Contudo, o algoritmo não é eficaz em três cenários principais: (1) quando a embarcação altera seu MMSI para um identificador inválido ou de uma embarcação inativa; (2) quando

a embarcação falsifica seu MMSI utilizando o identificador de um navio próximo, pois a velocidade resultante pode parecer plausível; e (3) em casos de *spoofing* de latitude/longitude, onde o AIS transmite posições manipuladas sem alterar o MMSI. Este método efetivamente identifica *spoofing* baseado em mudanças abruptas de velocidade, mas sua eficácia é reduzida em cenários onde o identificador falso não gera discrepâncias evidentes na trajetória.

Outra abordagem amplamente utilizada para detecção de adulteração de MMSI são os algoritmos de associação de acompanhamentos radar com AIS. Em [9], são comparadas técnicas tradicionais numéricas de associação com abordagens baseadas em *deep learning*, enquanto [10] utiliza MLE (*Maximum Likelihood Estimation*) para associar acompanhamentos AIS e radar. O trabalho [11] propõe um método de associação radar-AIS para detectar desligamentos do AIS e adulteração de MMSI. Como o acompanhamento por radar é mais confiável, quando um contato AIS é associado a um contato radar, a embarcação pode ser monitorada mesmo que o AIS seja desligado ou tenha seu identificador alterado. No entanto, todas essas técnicas de que utilizam a estratégia de associação de contato radar e AIS apresentam uma desvantagem significativa: sua aplicabilidade depende da presença de um radar na região de análise, o que limita seu uso em alto-mar ou em áreas sem infraestrutura de monitoramento.

Além das técnicas já citadas, em [12], é proposta uma técnica baseada em *deep learning* para associar novas amostras AIS a uma trajetória existente, utilizando apenas a informação de posição e ignorando o identificador MMSI. Isso permitiria acompanhar a embarcação mesmo após a alteração do identificador. Contudo, essa abordagem possui duas limitações principais: (1) exige alto poder computacional, tornando-se inviável em regiões de tráfego intenso devido ao grande volume de novas amostras AIS sendo coletadas continuamente; e (2) não foi testada em cenários onde a embarcação mantém seu transmissor AIS desligado por longos períodos, deixando incertezas sobre sua eficácia em tais situações.

Técnicas de previsão de trajetória também são utilizadas tanto para prevenção de acidentes quanto para estimar a rota percorrida por embarcações que interrompem sua transmissão AIS, seja por falha ou manipulação intencional. Essas técnicas variam desde abordagens tradicionais de aprendizado não supervisionado até modelos modernos baseados em *deep learning*.

O estudo [13] apresenta uma revisão da literatura comparando cinco técnicas tradicionais de aprendizado de máquina e sete técnicas de *deep learning*, comparando os resultados utilizando métricas de desempenho e necessidade de dados para treinamento. Os resultados mostram um desempenho significativamente superior das técnicas de *deep learning* o que corrobora com a aplicação desse tipo de técnica neste trabalho. Em [14], o autor utiliza um modelo de redes adversárias generativas

(GAN - *Generative Adversarial Networks*) aliado a um mapa de calor gerado a partir do histórico de trajetórias para prever a posição da embarcação até 10 minutos à frente. No entanto, a curta janela de previsão torna essa abordagem inadequada para aplicação em adulteração de AIS pelos períodos prolongados de tempo envolvido.

Por outro lado, [15] propõe a arquitetura *TrAISformer*, que combina técnicas de codificação com camadas de *transformers* para prever trajetórias de longo prazo. O modelo foi testado para previsões de até 10 (dez) horas, apresentando um erro médio de aproximadamente 10 (dez) milhas náuticas, e superou outras arquiteturas de redes neurais propostas em outros estudos na literatura no cenário estudado do trabalho. Apesar dos resultados promissores, o modelo foi testado apenas em um cenário específico, sendo desconhecido o seu desempenho em diferentes condições geográficas e em conjuntos de dados com diferentes características.

Diante das limitações identificadas nas abordagens existentes, este trabalho propõe uma solução baseada em previsão de trajetórias combinada com técnicas de associação de trajetórias. A técnica de previsão utilizada foi inspirada em [15], enquanto enquanto foram utilizados algoritmos classificadores para realizar a associação de trajetórias. A descrição detalhada da solução proposta e dos resultados obtidos será apresentada nos capítulos 3 e 4 deste trabalho.

Capítulo 3

Método Proposto

3.1 Descrição do problema

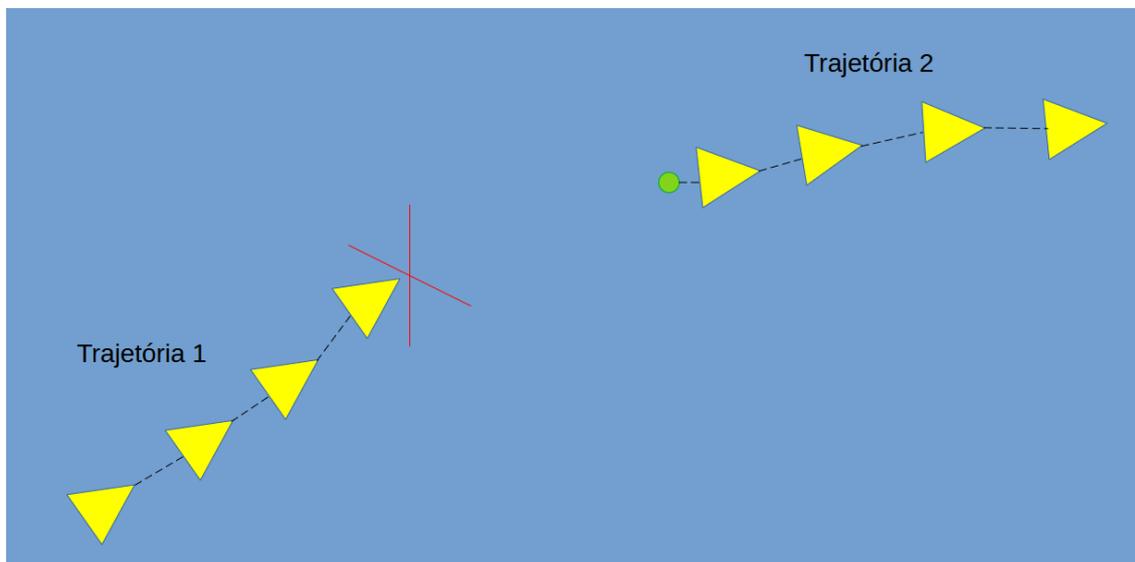
Suponha que um navio tenha a intenção de infringir alguma norma de navegação ou cometer algum crime em alto mar. Neste contexto, o equipamento AIS instalado dentro do navio revelaria o seu comportamento infracional e o navio poderia ser identificado e autuado pela sua infração. Desta forma, é possível que o navio simplesmente altere o seu identificador(MMSI) registrado no equipamento AIS, fazendo-se passar por alguma outra embarcação e possivelmente fugindo de eventuais penalidades. Durante monitoramentos das mensagens AIS realizados ao redor do mundo, não raramente, são identificadas momentos onde mais de um navio estava utilizando o mesmo identificador MMSI, o que sugere que esse tipo de situação descrita aconteça com certa frequência [3].

A Figura 3.1 representa esse tipo de situação, a trajetória 1 representa a trajetória realizada por uma determinada embarcação até o momento do desligamento do equipamento AIS. Algum tempo após o desligamento do equipamento AIS, a mesma embarcação religa o equipamento, porém com um identificador diferente do anterior. A trajetória 2 representa a trajetória da embarcação após o religamento do equipamento AIS, já com uma identificação diferente e que se iniciou algum tempo depois do término de transmissão da trajetória da embarcação 1. Neste cenário, como é possível identificar se as trajetórias 1 e 2 são, na verdade, uma única trajetória da mesma embarcação?

Em um cenário ainda mais realista, poderiam existir diversas embarcações navegando na mesma região conforme Figura 3.2, o que dificulta ainda mais a identificação da continuidade da trajetória do navio. Sendo assim, este trabalho tem como objetivo identificar qual das trajetórias, aparentemente não relacionadas, podem ser a continuação da trajetória 1 e também, como resultado secundário, estimar a trajetória completa realizada pelo navio no período em que o equipamento esteve

desligado.

Figura 3.1: Exemplo de trajetórias



3.2 Método proposto

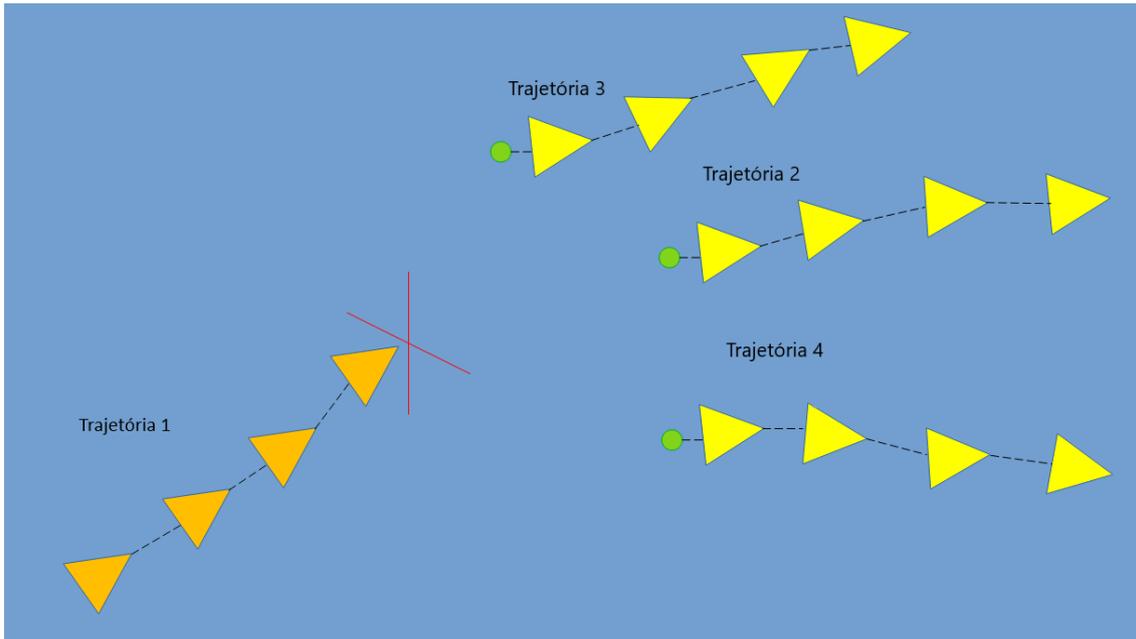
A estratégia adotada neste trabalho para resolver o problema descrito consiste em duas etapas distintas. Conforme ilustrado na Figura 3.2, a primeira etapa envolve a previsão da continuação da trajetória 1 após a interrupção do fornecimento de sua localização. Paralelamente, também são feitas previsões do "passado" das trajetórias que potencialmente podem representar sua continuação, neste caso, as trajetórias 2, 3 e 4.

Adotamos a nomenclatura "previsão para frente" para a projeção futura da trajetória 1 e "previsão para trás" para a reconstrução do passado das trajetórias candidatas (2, 3 e 4). Após a realização dessas previsões, inicia-se a segunda etapa, que consiste na comparação, em pares, entre a trajetória 1 e as trajetórias 2, 3 e 4, utilizando um classificador. O objetivo dessa etapa é verificar se alguma dessas trajetórias pode, de fato, ser a continuação da trajetória 1.

A Figura 3.3 ilustra as previsões realizadas, destacando a previsão para frente da trajetória 1 e as previsões para trás das trajetórias 2, 3 e 4.

Com base na literatura acadêmica apresentada, foi selecionada uma técnica para etapa de previsão e três técnicas para etapa de classificação.

Figura 3.2: Exemplo de trajetórias em um ambiente real



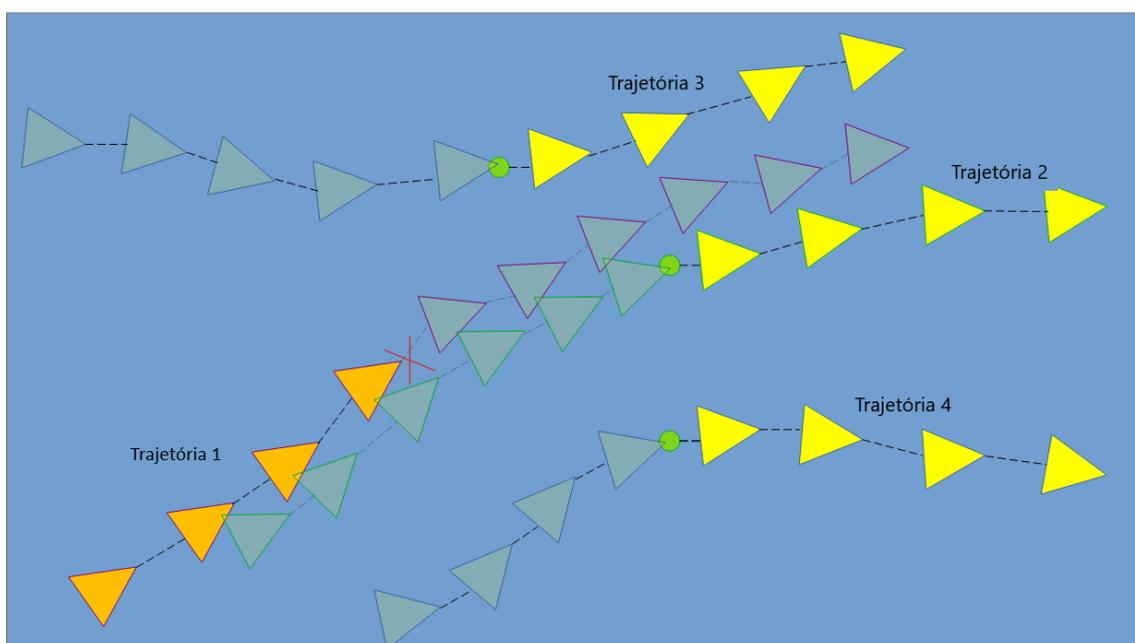
3.2.1 Técnica para previsão de trajetória: TrAISformer

A técnica utilizada na previsão foi a técnica de *deep learning* com a arquitetura denominada *TrAISformer* proposta em [15]. Esta arquitetura utiliza uma combinação de técnicas de codificação de dados e rede de *transformers* para realizar as previsões de trajetórias.

A arquitetura *TrAISformer* recebe como entrada até 120 amostras sequenciais de uma trajetória, sendo que a taxa de coleta das amostras deve ser fixa. Cada amostra contém quatro atributos: latitude, longitude, curso e velocidade. Inicialmente, os atributos de cada amostra são codificados utilizando *one-hot encoding*. Em seguida, esses dados passam por uma camada de *embedding*, que transforma cada atributo em um vetor de *embedding*. Após essa etapa, os dados seguem para uma rede de *transformers*, responsável pela previsão da trajetória. O resultado gerado mantém a sequência de entrada, mas inclui uma amostra adicional prevista, totalizando até 121 amostras na saída. Cada amostra de saída possui os mesmos atributos das amostras de entrada (latitude, longitude, curso e velocidade), porém, na amostra prevista, em vez de valores binários (0 ou 1) na codificação *one-hot*, o vetor contém probabilidades associadas a cada possível valor. Para definir os valores finais da previsão, utiliza-se um sorteio multinomial, para cada um dos atributos, baseado nas probabilidades geradas pela rede.

A rede possui um tamanho de entrada fixo de 120 amostras e um tamanho de saída fixo de 121 amostras. No entanto, é possível utilizar trajetórias com menos de 120 amostras, completando os dados com zeros até atingir o total necessário. O

Figura 3.3: Exemplo de trajetórias e suas previsões



mesmo mecanismo é aplicado na saída: caso a previsão gere menos de 121 amostras, a própria rede preenche os vetores de saída com zeros para manter o formato padronizado.

A arquitetura *TrAISformer* é capaz de prever apenas uma amostra por iteração. Para obter previsões mais longas, a rede deve ser acionada iterativamente, utilizando a saída da previsão anterior como entrada para a próxima iteração. Dessa forma, é possível gerar trajetórias estendidas conforme necessário. Por exemplo, ao inserir uma trajetória de 36 amostras, a rede retorna uma sequência de 37 amostras, sendo 36 da entrada original e 1 amostra prevista. Para prever trajetórias mais longas, a rede é acionada recursivamente, adicionando uma nova previsão a cada iteração, até atingir o tamanho desejado.

As redes neurais foram treinadas utilizando os dados descritos no item 3.2.3 deste trabalho. O objetivo do treinamento é permitir que a rede identifique a trajetória futura (ou passada) da embarcação, com base no comportamento de navegação atual e no histórico de navegação local das embarcações.

3.2.2 Técnicas de classificação

A segunda etapa da solução proposta envolve o uso de algoritmos de classificação para comparar as trajetórias previstas nas etapas anteriores (previsão "para frente" e "para trás"). O objetivo é identificar quais trajetórias podem representar a continuação da trajetória original da embarcação que supostamente adulterou seu identificador. Para esta tarefa, foram testados três algoritmos de classificação. O

primeiro e o segundo modelos são os modelos *Random Forest* e o *SVM*, que são dois modelos de classificação amplamente conhecidos, estudados e utilizados na literatura, já o terceiro modelo foi desenvolvido pelo Instituto de Pesquisas da Marinha do Brasil(IPqM) com o objetivo de realizar a detecção e combinação de dois contatos detectados por radares diferentes, porém que são dois contatos de uma mesma embarcação. Este algoritmo funciona de forma "online", porém, foi adaptado para ser utilizado de forma estática para que seu resultado pudesse ser comparado com o resultado dos outros algoritmos utilizados.

Modelo de classificação da Marinha do Brasil

Este modelo de classificação foi desenvolvido para analisar contatos detectados por radares diferentes com objetivo de identificar se esses contatos são, na verdade, a mesma embarcação detectada por diferentes sensores. Desta forma, a tarefa original em que se utiliza o algoritmo é diferente da tarefa pretendida neste trabalho. Apesar disso, as duas tarefas apresentam semelhanças, motivo pelo qual este trabalho realizou uma adaptação no algoritmo para que pudesse ser aplicado no desafio em questão e analisou o desempenho deste algoritmo na tarefa que este trabalho propõe.

Este modelo de classificação é baseado no teste de hipótese χ^2 , com fundamento na distância de Mahalanobis e na teoria de testes estatísticos para dados multivariados. Este método é aplicado para verificar se duas trajetórias (como pares de posições ou velocidades) pertencem à mesma embarcação, assumindo-se que pequenas variações entre elas são resultantes de ruído ou erro de medição. Sendo assim, o teste sempre se utiliza de um par de trajetórias para comparação.

A estatística utilizada no teste é dada por:

$$\chi^2 = D\Sigma^{-1}D^T \tag{3.1}$$

onde:

- D é um vetor de diferenças entre os valores de duas trajetórias (por exemplo, posições ou velocidades em x e y);
- Σ^{-1} é a matriz de covariância inversa dos dados históricos;
- D^T é a transposta de D .

Essa fórmula representa o **quadrado da distância de Mahalanobis**, que mede a distância entre vetores levando em conta a variância e covariância entre as variáveis. Diferentemente da distância euclidiana, ela é sensível à estrutura estatística dos dados.

Se os dados seguem uma distribuição normal multivariada, a distância de Mahalanobis ao quadrado segue uma distribuição χ^2 com k graus de liberdade, onde k é

o número de variáveis analisadas (por exemplo, $k = 2N$ em uma janela deslizante com N amostras em 2 dimensões).

O seguinte teste de hipótese é formulado:

- **Hipótese nula (H_0):** As duas trajetórias pertencem à mesma embarcação. As diferenças observadas são atribuídas a ruídos e erros de medição, seguindo uma distribuição de probabilidade constante.
- **Hipótese alternativa (H_1):** As trajetórias pertencem a embarcações distintas, sendo as diferenças estatisticamente significativas.

Para avaliar a hipótese nula, é definida uma janela deslizante de tamanho N na qual, a cada iteração é calculada uma nova estatística χ^2 utilizando a matriz de covariância inversa (Σ^{-1}), calculada a partir dos dados dos últimos N pontos, e o ponto mais recente (D), conforme equação 3.1. Em seguida, compara-se a média das estatísticas χ^2 dos últimos N pontos com um **limiar crítico** da distribuição χ^2 com nível de confiança de 99,5% e $2N$ graus de liberdade. Se a média das estatísticas for inferior ao valor crítico, a hipótese nula é aceita para aquela janela.

A cada iteração da janela deslizante dois testes independentes são realizados: um com base na diferença de posições ($D = [D_x, D_y]$) (convertendo as coordenadas geodésicas em coordenadas cartesianas) e outro com base na diferença de velocidades convertidas para os eixos cartesianos ($DV = [DV_x, DV_y]$). A hipótese nula é aceita apenas se ambos os testes forem positivos.

A Tabela 3.1 ilustra o funcionamento da janela deslizante. Observando a Tabela 3.1, é possível notar que o algoritmo funciona de maneira *online*, realizando vários testes em sequência a medida que a janela deslizante avança. No exemplo da Tabela 3.1, após 5 amostras do vetor D_n é realizado o cálculo da estatística χ^2 e um teste de hipótese com a média das últimas 5 estatísticas χ^2 calculadas. Este processo é repetido a cada atualização da janela deslizante, resultando, no caso deste exemplo, em 8 resultados dos testes de hipótese. Sendo assim, neste trabalho, foi introduzido o conceito de limiar de detecção, que representa o número mínimo de resultados positivos no teste de hipótese necessários para que um par de trajetórias seja considerado como pertencente à mesma embarcação. Dessa forma, em cada experimento realizado, dois parâmetros precisam ser ajustados: o tamanho da janela deslizante e o limiar de detecção.

O algoritmo de classificação MB está descrito em pseudocódigo no algoritmo 1

3.2.3 Dados utilizados

Neste trabalho, foram utilizados cinco conjuntos de dados provenientes de diferentes localidades, cada um com características distintas, com o objetivo de analisar as

1	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	D_{12}
	-	-	-	-	χ_5^2	-	-	-	-	-	-	-
2	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	D_{12}
	-	-	-	-	χ_5^2	χ_6^2	-	-	-	-	-	-
3	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	D_{12}
	-	-	-	-	χ_5^2	χ_6^2	χ_7^2	-	-	-	-	-
4	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	D_{12}
	-	-	-	-	χ_5^2	χ_6^2	χ_7^2	χ_8^2	-	-	-	-
5	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	D_{12}
	-	-	-	-	χ_5^2	χ_6^2	χ_7^2	χ_8^2	χ_9^2	-	-	-
6	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	D_{12}
	-	-	-	-	χ_5^2	χ_6^2	χ_7^2	χ_8^2	χ_9^2	χ_{10}^2	-	-
7	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	D_{12}
	-	-	-	-	χ_5^2	χ_6^2	χ_7^2	χ_8^2	χ_9^2	χ_{10}^2	χ_{11}^2	-
8	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}	D_{12}
	-	-	-	-	χ_5^2	χ_6^2	χ_7^2	χ_8^2	χ_9^2	χ_{10}^2	χ_{11}^2	χ_{12}^2

Tabela 3.1: Exemplo do funcionamento da janela deslizante considerando uma janela de tamanho 5, sendo D_n um vetor (x,y) com a diferença em x e em y entre as n -ésimas amostras do par de trajetórias. As posições marcadas em amarelo e verde são as posições que estão dentro da janela e utilizadas no cálculo da matriz de covariância da equação 3.1, sendo a posição verde o ponto D na equação 3.1 no qual está sendo testada a hipótese

Algorithm 1: algoritmo de classificação MB

Data: x, n, l
Result: y
 $D \leftarrow x$;
 $N \leftarrow n$; /* Tamanho da janela */
 $I \leftarrow N$;
 $L \leftarrow l$; /* Limiar de detecção */
 $J \leftarrow N$;
 $Pos \leftarrow 0$; /* Número de detecções positivas */
 $\chi_{arr}^2 \leftarrow$ Array vazio;
while $I \leq$ tamanho de D **do**
 $\Sigma^{-1} =$ calculaMatrizCovarianciaInversa($D[I - N : I]$)
 $\chi^2 = D[I]\Sigma^{-1}D[I]^T$;
 $\chi_{arr}^2[I - N] \leftarrow \chi^2$;
 $I \leftarrow I + 1$;
end
while $J \leq$ tamanho de χ_{arr}^2 **do**
 $M =$ calculaMedia($\chi_{arr}^2[J - N : J]$) ;
 if $M \leq$ distChiSquare($2N, 99, 5\%$) **then**
 $Pos \leftarrow Pos + 1$;
 end
 $J \leftarrow J + 1$;
end
if $Pos \geq L$ **then**
 $y \leftarrow True$;
else
 $y \leftarrow False$;
end

capacidades e limitações dos modelos empregados tanto na previsão de trajetórias quanto na classificação. Todos os conjuntos de dados são compostos por informações obtidas a partir de sensores AIS.

Devido a sua natureza, os dados oriundos do sistema AIS são amplamente conhecidos por possuírem uma série de imprecisões e erros que podem ter diversas origens como falhas na comunicação, imprecisão na determinação de posição ou de velocidade no sistema GPS, falha no próprio equipamento, entre outras origens. Sendo assim, os dados são bastante ruidosos e requerem uma série de medidas de pré-processamento para mitigar os efeitos destes ruídos.

Cada amostra do sensor AIS contém seis atributos: latitude, longitude, curso, velocidade, tempo e MMSI. A latitude e a longitude representam a posição geográfica da embarcação em coordenadas geográficas (graus). O curso indica a direção do deslocamento da embarcação em relação ao norte (medido em graus). A velocidade, expressa em nós, corresponde à taxa de deslocamento da embarcação na direção do curso. O tempo representa o instante em que essas informações foram registradas. Por fim, o MMSI é o código identificador único da embarcação de origem dos dados. Uma sequência temporal de amostras pertencentes a uma mesma embarcação forma uma trajetória.

Os dados AIS possuem um intervalo variável entre mensagens, que pode oscilar entre 2 segundos e 6 minutos, dependendo da velocidade da embarcação e da origem da transmissão (AIS satelital ou costeiro). Essa variação resulta em um grande volume de dados, que frequentemente são descartados para economizar espaço de armazenamento.

Além disso, a arquitetura da rede neural *TrAISformer*, utilizada neste trabalho, requer um intervalo de tempo fixo entre as amostras. Para atender a essa exigência, os dados coletados foram submetidos ao método de interpolação linear, mesmo método utilizado no trabalho que propôs a arquitetura *TrAISformer* ([15]), garantindo a uniformização do tempo entre as amostras. Com o intuito de avaliar os efeitos da redução da taxa de amostragem na coleta dos dados, diferentes intervalos entre amostras foram testados nos distintos conjuntos de dados.

Dessa forma, uma trajetória (Traj) é composta por T amostras de uma mesma embarcação (x_t), com espaçamentos temporais uniformes entre as observações.

$$x_i = [Lat_i, Long_i, curso_i, velocidade_i] \quad (3.2)$$

$$Traj = [x_1, x_2, x_3, \dots, x_T] \quad (3.3)$$

A técnica de interpolação linear, utilizada para fixar o tempo entre as amostras AIS e reduzir os efeitos das perdas de dados, consiste em estimar o valor em um

determinado instante de tempo traçando uma reta entre a medida anterior e a medida posterior mais próximas desse instante. A partir dessa reta, obtém-se a estimativa do valor no instante desejado. Dessa forma, a amostragem é realizada em intervalos fixos de tempo utilizando valores estimados para os atributos.

Além da interpolação linear para manter fixo o intervalo temporal entre as amostras, foram aplicados os seguintes pré-processamentos em todos os conjuntos de dados:

- Remoção de dados AIS com velocidade maior que 30 nós;
 - Dadas as características dos navios cujos dados foram utilizados neste estudo (navios tanque e navios cargueiros), é sabido que estes navios não navegam em velocidades altas, desta forma velocidades acima de 30 nós foram consideradas erros de medição e removidos do conjunto de dados.
- remoção de embarcações atracadas e ancoradas;
 - Embarcações atracadas e ancoradas estão relativamente paradas e não possuem relevância para este estudo.
- remoção de embarcações que não são do tipo tanque ou cargueiro;
 - A arquitetura de rede neural adotada neste trabalho (*TrAISformer*) foi desenvolvida e testada utilizando apenas dados de navios tanque e cargueiro. Navios de pequeno porte podem apresentar uma dinâmica de movimentação mais "aleatória", dificultando o treinamento da rede neural.
- remoção de embarcações navegando a menos de uma milha de distância da costa;
 - Embarcações do tipo tanque e do tipo cargueiro navegando em mar aberto não se movimentam a menos de uma milha da costa a não ser em casos de entrada e saída do porto, o que são situações não relevantes para este trabalho.
- separação em duas rotas diferentes caso a embarcação fique mais de duas horas sem emitir dado AIS;
 - Trajetórias com longo período sem dados acarretaria aumento no erro na estimativa feita na etapa de interpolação linear, o que não é desejado. Desta forma, estipulou-se de maneira empírica o limite de duas horas.
- remoção de trajetórias com duração menor que 3 horas e 10 minutos.

- Estas trajetórias foram removidas por terem uma duração menor que o tamanho inicial de trajetória escolhido para ser utilizado como entrada nos modelos de previsão para que ele realize as previsões. Este tamanho de trajetória inicial foi escolhido com base no artigo inicial que propôs o modelos TrAISformer.
- divisão de trajetórias longas em trajetórias menores na etapa de treinamento dos modelos de previsão.
 - Para os conjuntos de dados 1, 2 e 3, foi utilizado um limite de 121 pontos na duração da trajetória. Para o conjunto de dados 4 e 5 foi utilizado o limite de 300 pontos. Esta divisão de trajetórias foi utilizada para que houvesse um maior número de trajetórias para a etapa de treinamento. O limite de 121 pontos foi escolhido por ser o mesmo limite de entrada dos modelos de previsão. Porém, como os conjuntos de dados 4 e 5 possuem uma taxa de amostragem maior, uma trajetória com 121 pontos seria muito curta (apenas 4 horas), por esse motivo foi utilizado um limite de 300 pontos.

Conjunto de dados 1

O conjunto de dados 1 consiste em dados públicos de AIS disponibilizados pela Autoridade Marítima Dinamarquesa (DMA - *Danish Maritime Authority*). Foram utilizados dados coletados no período de 01 de janeiro de 2019 a 31 de março de 2019, abrangendo uma região delimitada por um retângulo entre as coordenadas geográficas de latitude $55,5^\circ$ e longitude $10,3^\circ$, e latitude $58,0^\circ$ e longitude $13,0^\circ$ (Figura 3.4(a)).

A divisão do conjunto de dados seguiu a seguinte estrutura: os dados de 01 de janeiro de 2019 a 10 de março de 2019 foram utilizados para treinamento, os dados de 11 de março a 20 de março de 2019 foram reservados para validação e ajuste de hiper parâmetros dos modelos, e os dados de 21 de março a 31 de março de 2019 foram empregados na etapa de testes dos modelos.

Após a execução de todo o pré-processamento, o conjunto final de dados contém 11888 trajetórias, distribuídas da seguinte forma: 9144 trajetórias para treinamento, 1291 trajetórias para validação e 1453 trajetórias para teste. O intervalo entre amostras nesse conjunto de dados é de 10 minutos, e a duração das trajetórias varia entre 3 horas e 10 minutos (19 pontos) e 20 horas e 10 minutos (121 pontos). O treinamento e as previsões dos modelos de previsão de trajetória são realizados a partir de trajetórias iniciais de 3 horas (18 pontos).

Conjunto de dados 2

O conjunto de dados 2 consiste em dados AIS coletados pela Marinha do Brasil no período de 01 de junho a 31 de agosto de 2024, em uma região próxima ao porto de Santos. A área de coleta é delimitada por um retângulo cujos vértices possuem as seguintes coordenadas: (Lat: $-24,48^\circ$, Long: $-46,94^\circ$), (Lat: $-23,27^\circ$, Long: $-43,97^\circ$), (Lat: $-24,62^\circ$, Long: $-43,32^\circ$) e (Lat: $-25,89^\circ$, Long: $-46,34^\circ$) (Figura 3.4(b)).

A divisão do conjunto de dados seguiu a seguinte estrutura: os dados de 01 de junho a 10 de agosto de 2024 foram utilizados para treinamento, os dados de 11 a 20 de agosto de 2024 foram reservados para validação e ajuste de hiper parâmetros dos modelos, e os dados de 21 a 31 de agosto de 2024 foram empregados na etapa de testes dos modelos.

Após a execução de todo o pré-processamento, o conjunto final de dados contém 2143 trajetórias, distribuídas da seguinte forma: 1328 trajetórias para treinamento, 379 trajetórias para validação e 438 trajetórias para teste. O intervalo entre amostras nesse conjunto de dados é de 10 minutos, e a duração das trajetórias varia entre 3 horas e 10 minutos (19 pontos) e 20 horas e 10 minutos (121 pontos). O treinamento e as previsões dos modelos de previsão de trajetória são realizados a partir de trajetórias iniciais de 3 horas (18 pontos).

Conjunto de dados 3

O conjunto de dados 3 é composto pelos mesmos dados utilizados no conjunto de dados 2, diferenciando-se apenas pelo tempo entre as amostras AIS, que, neste caso, foi reduzido para 5 minutos. O objetivo dessa alteração é avaliar os efeitos da redução do intervalo entre amostras nos resultados das previsões geradas pelos modelos treinados, tanto os modelos de previsão quanto os de classificação.

Após o pré-processamento, o conjunto final de dados contém 3106 trajetórias, distribuídas da seguinte forma: 1875 trajetórias para treinamento, 567 trajetórias para validação e 664 trajetórias para teste. O intervalo entre amostras nesse conjunto é de 5 minutos, e a duração das trajetórias varia entre 3 horas e 10 minutos (38 pontos) e 10 horas e 5 minutos (121 pontos). O treinamento e previsão dos modelos de previsão de trajetória são realizados a partir de trajetórias iniciais de 3 horas (36 pontos).

Conjunto de dados 4

O conjunto de dados 4 consiste em dados AIS coletados e disponibilizados pela organização *Marine Cadastre* (<https://hub.marinecadastre.gov/pages/vesseltraffic>), uma parceria entre o *Bureau of Ocean Energy Management* (BOEM) e a *National Oceanic and Atmospheric Administration* (NOAA), ambos órgãos do governo dos

Estados Unidos. Os dados foram coletados no período de 01 de janeiro a 31 de dezembro de 2022, na região do Golfo do México, delimitada por um retângulo cujos vértices são: (Lat: 30,14° , Long: -77,16°), (Lat: 30,14° , Long: -88,60°), (Lat: 20,0° , Long: -77,16°) e (Lat: 20,0° , Long: -88,60°) (Figura 3.4(c)).

Os dados de 01 de janeiro a 24 de novembro de 2022 foram utilizados para treinamento dos modelos, os dados de 25 de novembro a 09 de dezembro de 2022 foram reservados para validação e ajuste de hiperparâmetros, enquanto os dados de 10 de dezembro a 31 de dezembro de 2022 foram empregados na fase de testes.

Após o pré-processamento, o conjunto final de dados contém 73946 trajetórias, distribuídas da seguinte forma: 66656 trajetórias para treinamento, 3257 trajetórias para validação e 4033 trajetórias para teste. O intervalo entre as amostras neste conjunto é de 2 minutos, e a duração das trajetórias varia entre 3 horas e 10 minutos (95 pontos) e 10 horas e 10 minutos (300 pontos). O treinamento e previsão dos modelos de previsão de trajetória são realizados a partir de trajetórias iniciais de 3 horas (90 pontos).

Conjunto de dados 5

O conjunto de dados 5 é composto pelos mesmos dados do conjunto de dados 4, mantendo o mesmo intervalo entre amostras e, conseqüentemente, o mesmo número de trajetórias para treinamento, validação e teste. A única diferença entre os dois conjuntos é que, no conjunto de dados 5, a trajetória inicial utilizada tem duração de 2 horas (60 pontos), enquanto no conjunto de dados 4 foi utilizada uma trajetória inicial de 3 horas (90 pontos). O objetivo dessa modificação foi avaliar o impacto do tamanho da trajetória inicial na acurácia dos modelos.

Após o pré-processamento, o conjunto final de dados contém 73946 trajetórias, distribuídas da seguinte forma: 66656 trajetórias para treinamento, 3257 trajetórias para validação e 4.033 trajetórias para teste. O intervalo entre as amostras permanece de 2 minutos, e a duração das trajetórias varia entre 3 horas e 10 minutos (95 pontos) e 10 horas e 10 minutos (300 pontos). O treinamento e previsão dos modelos de previsão de trajetória são realizados a partir de trajetórias iniciais de 2 horas (60 pontos).

A Tabela 3.2 resume as principais informações sobre os cinco conjuntos de dados utilizados. A coluna "Região" indica a região geográfica em que os dados foram coletados, a coluna "Amostragem" indica o intervalo entre as amostras do conjunto de dados, a coluna "Tempo de coleta" indica a duração da coleta dos dados, a coluna "Trajetórias" indica o número de trajetórias contidas no conjunto de dados após o pré-processamento, a coluna "Início de previsões" indica o tamanho mínimo inicial das trajetórias, em horas, a partir do qual o modelo realiza as previsões.

Conjunto	Região	Amostragem	Tempo de coleta	Trajetórias	Início de previsões
1	Dinamarca	10 min	3 meses	11888	3 horas
2	Brasil	10 min	3 meses	2143	3 horas
3		5 min	3 meses	3106	3 horas
4	Golfo do	2 min	12 meses	73946	3 horas
5	México	2 min	12 meses	73946	2 horas

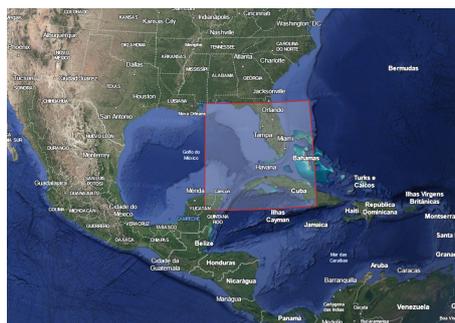
Tabela 3.2: Tabela Resumo das características dos conjuntos de dados utilizados.



(a) Conjunto de dados 1.



(b) Conjunto de dados 2 e 3.



(c) Conjunto de dados 4 e 5.

Figura 3.4: As imagens 3.4(a), 3.4(b) e 3.4(c) mostram a área geográfica delimitada para coleta de dados AIS para cada um dos conjuntos de dados.

3.3 Procedimento experimental

3.3.1 Treinamento e teste dos modelos de previsão de trajetória

O experimento teve início com o treinamento das redes neurais baseadas na arquitetura *TrAISformer*. Para isso, os dados foram pré-processados e separados em conjuntos de treinamento, validação e teste, conforme descrito na Seção 3.2.3.

Durante o treinamento, para cada trajetória do conjunto de treinamento, foi apresentado à rede o segmento inicial da trajetória com o tamanho mínimo necessário para realizar previsões (3 horas para os conjuntos de dados 1, 2, 3 e 4, e 2 horas para o conjunto de dados 5). A rede então gerava a previsão de uma única amostra, que era concatenada ao segmento inicial, sendo esse novo segmento reapresentado à rede. Esse processo foi repetido até que a trajetória prevista tivesse o mesmo comprimento da trajetória real da embarcação.

Com a trajetória real e a estimada, foi calculado o erro ponto a ponto, seguindo o mesmo critério da proposta original da arquitetura *TrAISformer*. Esse erro era então propagado pela rede via o algoritmo de *backpropagation*, com o objetivo de ajustar os pesos da rede e aprimorar as previsões. Após processar todas as trajetórias do conjunto de treinamento, o modelo era avaliado com as trajetórias do conjunto de validação, calculando-se o erro de previsão. Esse ciclo de treinamento e validação era repetido até que o erro no conjunto de validação fosse minimizado, momento em que a rede era considerada treinada para aquele conjunto de dados.

Esse procedimento foi realizado individualmente para cada um dos cinco conjuntos de dados. Para cada conjunto, foram treinadas duas redes: uma para realizar previsões "para frente" (futuras) e outra "para trás" (passadas). A única diferença entre elas é a ordem de apresentação dos dados: a rede "para frente" recebeu os dados em ordem cronológica (do mais antigo para o mais recente), enquanto a rede "para trás" recebeu os dados em ordem inversa (do mais recente para o mais antigo). Ambas utilizaram os mesmos conjuntos de dados.

Essa abordagem permite à rede neural capturar tanto padrões históricos de navegação na região quanto o comportamento típico das embarcações, habilitando o modelo a prever trajetórias a partir do histórico local de navegação.

Com os modelos treinados, realizou-se o estudo de precisão das previsões. Para isso, foram utilizadas as trajetórias do conjunto de teste. Apresentou-se à rede o segmento inicial com o tamanho mínimo necessário, e a rede gerava uma previsão que era concatenada ao segmento inicial, repetindo o processo até que a trajetória prevista tivesse o mesmo tamanho da trajetória real. Esse procedimento foi repetido para todas as trajetórias do conjunto de teste. As trajetórias previstas foram então

comparadas com as reais, e as métricas descritas na Seção 3.4 foram utilizadas para avaliar a precisão das previsões realizadas.

Em um cenário de utilização destas previsões em um sistema em funcionamento, a etapa de treinamento das redes neurais é uma etapa que deve ser executada de forma "offline" com dados históricos de navegação oriundas do sistema AIS já que esta etapa tem um alto custo computacional envolvido. Porém, a realização de estimativa de trajetórias pode ser realizada de forma "online" realizando, assim, uma estimativa futura ou passada da trajetória da embarcação.

3.3.2 Treinamento e teste dos modelos de classificação

Após o treinamento e a análise dos resultados dos modelos de previsão, a próxima etapa consistiu no treinamento e teste dos modelos de classificação, cuja finalidade é identificar trajetórias pertencentes à mesma embarcação. Para essa etapa, foi necessário a criação de novos conjuntos de dados compostos por pares de trajetórias, onde uma trajetória corresponde a uma previsão "para frente" e a outra a uma previsão "para trás", ambas obtidas a partir dos conjuntos de dados utilizados na fase de testes. Essa abordagem garante a coerência com o problema inicial abordado neste trabalho.

Neste contexto, para facilitar o entendimento, é necessária a definição de uma variável denominada *GAP*. Considerando um par de trajetórias utilizado para classificação, o *GAP* representa a distância temporal entre o fim da trajetória original utilizada na previsão para frente e o início da trajetória original utilizada na previsão para trás. No cenário real, o *GAP* corresponde ao intervalo de tempo em que a embarcação supostamente permaneceu com o equipamento AIS desligado. A duração do *GAP* é um fator crítico, pois acreditamos influenciar diretamente o desempenho e a aplicabilidade do algoritmo de classificação. A Figura 3.5 ilustra a variável *GAP*.

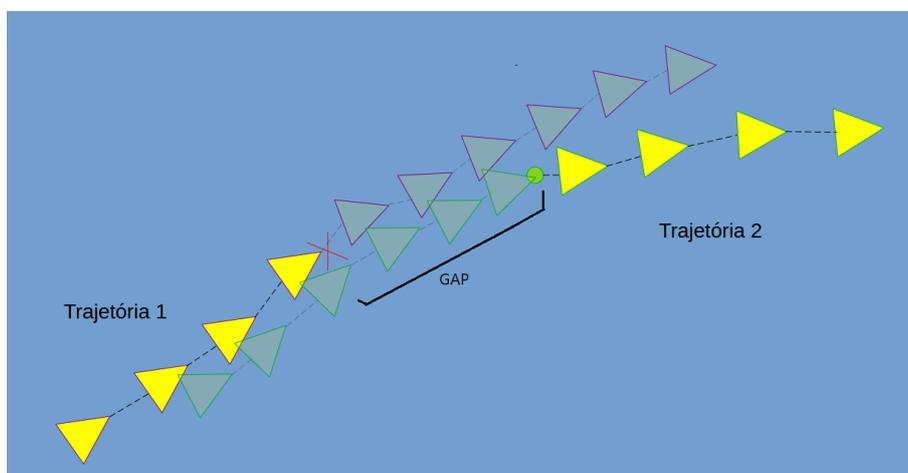


Figura 3.5: Ilustração da variável *GAP*.

Neste ponto, foram realizados dois experimentos. O primeiro teve como objetivo comparar o desempenho de diferentes algoritmos de classificação em um mesmo cenário de *GAP*, enquanto o segundo buscou analisar a influência do *GAP* no desempenho dos algoritmos de classificação.

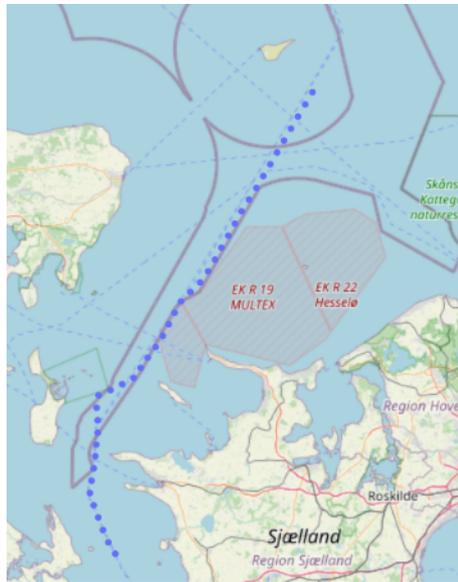
Para a construção dos novos conjuntos de dados utilizados no primeiro experimento, o primeiro passo foi a definição do tamanho das trajetórias a serem consideradas. Foi estabelecido que o experimento utilizaria um *GAP* fixo de 1 hora, o que resultou no uso de trajetórias com duração de 7 horas, exceto para o conjunto de dados 5, onde a duração foi de 5 horas. Dessa forma, a partir do conjunto de dados de teste de cada um dos cinco conjuntos de dados, foram removidas todas as trajetórias com duração inferior a 7 horas (ou 5 horas no caso do conjunto de dados 5). Já as trajetórias com duração superior a esses valores foram truncadas para o tempo máximo estabelecido.

Com as trajetórias ajustadas, procedeu-se à obtenção das previsões de trajetória utilizando os modelos treinados na etapa anterior. A previsão "para frente" foi gerada a partir das primeiras 3 horas da trajetória original (ou 2 horas para o conjunto de dados 5), estendendo-se até completar 7 horas (ou 5 horas para o conjunto de dados 5). Da mesma forma, a previsão "para trás" foi gerada a partir das últimas 3 horas da trajetória original (ou 2 horas no caso do conjunto de dados 5), estendendo-se até atingir a duração total definida.

A Figura 3.6 ilustra esse procedimento, exibindo uma trajetória original e os resultados obtidos com as previsões "para frente" e "para trás" no contexto do conjunto de dados 1.

Neste ponto, para cada trajetória original, obtivemos uma previsão "para frente" e uma previsão "para trás", formando assim um par de trajetórias classificado como "verdadeiro positivo", ou seja, um par que os classificadores devem identificar como pertencente à mesma trajetória. No entanto, para que os treinamentos e testes dos classificadores possam ser realizados, também são necessários pares de trajetórias "verdadeiro negativo", ou seja, pares que os classificadores devem identificar como trajetórias pertencentes a embarcações distintas.

Para obter os pares de trajetórias do tipo "verdadeiro negativo", foram combinadas previsões "para frente" de uma trajetória com previsões "para trás" de uma trajetória diferente, gerando assim pares de trajetórias que não pertencem à mesma embarcação ("verdadeiros negativos"). No entanto, observou-se que a simples combinação aleatória dessas trajetórias poderia resultar em pares facilmente identificáveis como verdadeiros negativos. Por exemplo, a combinação de uma trajetória em que o navio está navegando para o norte com outra em que a embarcação está navegando para o leste tornaria a tarefa trivial. Para reduzir esse efeito, foram adotados alguns critérios para a seleção dos pares de trajetórias.



(a) Trajetória original.



(b) Trajetória prevista "para frente".



(c) Trajetória prevista "para trás".

Figura 3.6: Amostra de resultados obtidos com as previsões realizadas para o conjunto de dados 1. Os pontos marcados em amarelo nas trajetórias 3.6(b) e 3.6(c) são os 18 pontos de da trajetória original a partir dos quais os modelos realizaram suas previsões.

Suponha que temos um par de trajetórias composto pelas trajetórias T_1 e T_2 . A trajetória T_1 corresponde à previsão "para frente", em que as primeiras 3 horas (ou 2 horas no caso do conjunto de dados 5) representam a parte conhecida da trajetória original, enquanto as 4 horas seguintes correspondem ao trecho previsto pelo modelo. Já a trajetória T_2 corresponde à previsão "para trás", em que as últimas 3 horas (ou 2 horas no conjunto de dados 5) representam a parte conhecida da trajetória original, enquanto as 4 horas iniciais correspondem à projeção gerada pelo modelo "para trás".

Para manter a coerência com o problema estudado, em que se propõe a comparar trajetórias que estão próximas geograficamente da trajetória prevista para a embarcação que desligou o equipamento AIS, a seleção dos pares de trajetórias "verdadeiro negativo" foi realizada comparando a parte conhecida da trajetória T_2 com as últimas horas da trajetória T_1 (parte prevista "para frente"). Considerando $R_{T_x}^i$ como o rumo da embarcação no ponto i da trajetória T_x , e $P_{T_x}^i$ como a localização geográfica (latitude e longitude) da embarcação nesse mesmo ponto, foram estabelecidos critérios específicos para a seleção.

O primeiro critério adotado baseia-se na direção de navegação das embarcações. Foi estabelecido que a média das diferenças, ponto a ponto, entre os cursos das trajetórias deveria ser inferior a 30 graus, conforme descrito na Equação 3.4. Esse critério foi definido para remover pares de trajetórias em que as embarcações estivessem navegando em direções completamente diferentes.

$$\frac{\sum_{i=N-n}^N D(R_{T_1}^i, R_{T_2}^i)}{n} < 30^\circ \quad (3.4)$$

Seja $D(A, B)$ a diferença, em graus, entre os ângulos A e B . Além disso, considere N como o número total de amostras da trajetória e n como o número de amostras correspondentes a um intervalo de 3 horas (ou 2 horas no caso do conjunto de dados 5).

O segundo critério estabelecido diz respeito à distância entre as embarcações. Foi definido que a média das distâncias, calculadas ponto a ponto entre as trajetórias, deveria ser inferior a 20 km, conforme a Equação 3.5. Esse critério tem como objetivo eliminar pares de trajetórias em que as embarcações estejam muito distantes uma da outra, garantindo que as trajetórias comparadas estejam espacialmente próximas e, portanto, representem cenários mais realistas para a tarefa de classificação.

$$\frac{\sum_{i=N-n}^N D_h(P_{T_1}^i, P_{T_2}^i)}{n} < 20Km \quad (3.5)$$

Seja $D_h(A, B)$ a distância, em quilômetros, entre os pontos A e B em coordenadas geográficas, calculada utilizando a fórmula de *haversine* (Equação 3.9).

O terceiro critério estabelecido determina que a distância entre os pontos $N - n$

de cada trajetória deve ser inferior a 10 km, conforme a Equação 3.6. Esse critério foi aplicado para remover pares de trajetórias que se iniciavam muito afastadas, garantindo uma melhor coerência espacial entre as trajetórias comparadas.

Todos os três critérios utilizados para a seleção dos pares de trajetórias foram definidos de maneira empírica, com base na observação dos resultados obtidos após a aplicação dos filtros.

$$D_h(P_{T_1}^{N-n}, P_{T_2}^{N-n}) < 10Km \quad (3.6)$$

Aplicando esses três critérios, foi possível obter um conjunto de pares de trajetórias classificadas como verdadeiros negativos para cada um dos conjuntos de dados. Com o objetivo de construir conjuntos de dados balanceados para a etapa de classificação, foram selecionados aleatoriamente, dentro do conjunto de verdadeiros negativos, uma quantidade equivalente à dos pares de verdadeiros positivos já reservados. Dessa forma, para cada um dos cinco conjuntos de dados, foi gerado um conjunto de pares de trajetórias contendo a mesma quantidade de pares verdadeiro positivo e verdadeiro negativo, garantindo equilíbrio na fase de treinamento e teste dos modelos de classificação.

Após a construção desses conjuntos de dados, foi necessário extrair características (*features*) de cada par de trajetórias para que pudessem ser utilizados nos algoritmos de classificação. As principais *features* extraídas foram:

- Distância ponto a ponto entre as duas trajetórias do par, calculada utilizando a fórmula de *haversine* (Equação 3.9).
- Módulo da diferença de velocidade ponto a ponto entre as trajetórias do par.
- Módulo da diferença de rumo entre as trajetórias do par.

Um detalhe importante sobre o cálculo da diferença de rumo é que, como o ângulo pode variar entre 0° e 360° , a subtração direta dos valores pode gerar distorções quando as embarcações navegam próximas a 0° ou 360° . Para corrigir esse problema, a diferença entre os rumos foi calculada subtraindo-se o maior ângulo do menor. Caso a diferença fosse superior a 180° , utilizava-se o complemento do ângulo para obter a menor diferença angular entre os rumos.

A Tabela 3.3 apresenta um resumo das principais informações sobre os conjuntos de dados de pares de trajetórias utilizados na etapa de classificação.

O mesmo procedimento utilizado para a criação dos conjuntos de pares de trajetórias no experimento com *GAP* de uma hora foi repetido para os conjuntos de dados 1, 3 e 5, abrangendo a obtenção de pares de trajetórias com valores de *GAP* variando de 0,5 até 7 horas. Esse processo incluiu a aplicação dos critérios para

Conjunto de dados	1	2	3	4	5
Numero de pares de trajetórias	2844	732	910	5726	5726
Número de pontos em cada trajetória	42	42	84	210	180
Número de <i>features</i> extraídas de cada par	126	126	252	630	540

Tabela 3.3: Tabela resumo de características dos conjuntos de dados para etapa de classificação.

	Conjunto de dados		
GAP (horas)	1	3	5
0,5	2914	938	5974
1	2844	910	5726
2	2704	846	5338
3	2550	740	4866
4	2340	644	4488
5	2120	484	3546
6	1916	-	3284
7	1696	-	3008

Tabela 3.4: Número de trajetórias para cada um dos conjuntos de dados em cada um dos experimentos com diferentes valores de GAP

a combinação de trajetórias classificadas como verdadeiros negativos, bem como a extração dos atributos de distância, diferença de velocidade e diferença de rumo ponto a ponto, exatamente como no procedimento anterior.

A Tabela 3.4 apresenta o número de trajetórias utilizadas em cada experimento para os diferentes valores de *GAP* em cada conjunto de dados. Nota-se que, para o conjunto de dados 3, não foi possível realizar os experimentos com *GAP* de 6 e 7 horas, devido à ausência de um número relevante de trajetórias com duração suficiente para serem utilizadas nesses cenários.

Após a construção dos conjuntos de dados necessários para os dois experimentos realizados na etapa de classificação, foram treinados modelos utilizando os algoritmos *SVM*, *Random Forest* e *MB*. Os treinamentos foram conduzidos com 80% das trajetórias destinadas ao treinamento e 20% para os testes, além da aplicação da técnica de validação cruzada *5-Fold* para treinamento e otimização dos hiperparâmetros dos modelos.

Em um cenário de utilização destas classificações em um sistema em funcionamento, a etapa de treinamento dos classificadores é uma etapa que deve ser executada de forma "offline" com dados históricos de navegação oriundas do sistema AIS já que esta etapa tem um alto custo computacional envolvido. Porém, a realização das classificações podem ser realizadas de forma "online" devido à utilização de algoritmos de classificação de baixo custo computacional.

As métricas utilizadas para analisar e comparar o desempenho dos algoritmos de classificação foram acurácia, taxa de verdadeiros positivos, taxa de falsos positivos e *F1-score*, todas descritas na Seção 3.4. Nos experimentos de análise do impacto do GAP, a métrica principal utilizada para avaliação dos resultados foi o *F1-score*.

3.4 Métricas

Para a análise de desempenho dos algoritmos utilizados neste trabalho, foram adotadas diversas métricas de avaliação nas diversas etapas do trabalho.

3.4.1 Erro médio

O **erro médio** é utilizado para avaliar a precisão das previsões realizadas pelos modelos propostos. Ele é um vetor que contém a média do erro para cada amostra prevista na trajetória.

Seja n o número total de trajetórias no conjunto de dados de teste e j o número de amostras previstas por trajetória. Para cada trajetória $i \in \{1, \dots, n\}$, define-se um vetor de erro $\text{Err}_i \in \mathbb{R}^j$, em que cada componente $\text{Err}_{i,k}$ representa o erro absoluto na k -ésima amostra prevista. Nas seções 3.4.3, 3.4.4 e 3.4.5, será apresentado como foram estimados os vetores Err_i para cada uma variáveis.

O vetor de erro médio $\mathbf{EM} \in \mathbb{R}^j$ é então definido por:

$$EM_k = \frac{1}{n} \sum_{i=1}^n \text{Err}_{i,k}, \quad \text{para } k = 1, 2, \dots, j \quad (3.7)$$

Ou, equivalentemente, em notação vetorial:

$$\mathbf{EM} = \frac{1}{n} \sum_{i=1}^n \mathbf{Err}_i \quad (3.8)$$

Essa média vetorial permite a análise da tendência de erro ao longo do tempo de previsão. O vetor \mathbf{EM} é utilizado para gerar as curvas de erro médio apresentadas na Seção 4.1, contemplando os erros de *posição*, *velocidade* e *rumo*.

3.4.2 Diagrama *Boxplot*

O diagrama *boxplot* foi elaborado utilizando os erros nas previsões ($\text{Err}_{i,k}$, com $i = 1, 2, \dots, n$) nos instantes de 1 e 3 horas (k equivalente a 1 hora e 3 horas). Foram elaborados diagramas *boxplot* para os erros de posição, velocidade e rumo e apresentados na seção 4.1.

3.4.3 Erro de posição das previsões de trajetórias

O erro nas previsões de posições das trajetórias foi definido como a distância ponto a ponto entre a posição prevista e a posição real da embarcação. Essa distância foi calculada através da fórmula de *Haversine*, descrita na Equação 3.9, que determina a distância, em quilômetros, entre duas coordenadas geodésicas (latitude e longitude). Assim, uma previsão de k pontos de uma trajetória gera k valores de erro correspondentes.

$$d = 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\varphi_2 - \varphi_1}{2} \right) + \cos(\varphi_1) \cos(\varphi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right) \quad (3.9)$$

Sendo:

d = distância calculada

r = 6371 (raio médio aproximado da Terra)

φ_1 e φ_2 = latitude dos pontos 1 e 2 em radianos

λ_1 e λ_2 = longitude dos pontos 1 e 2 em radianos

3.4.4 Erro de velocidade das previsões de trajetórias

O erro nas velocidades previstas das trajetórias foi definido como a diferença ponto a ponto entre a velocidade prevista e a velocidade real da embarcação. Assim, uma previsão com k pontos de uma trajetória gera k valores de erro correspondentes, calculados em nós.

3.4.5 Erro de rumo das previsões de trajetórias

O erro nos rumos das previsões das trajetórias foi definido como a diferença ponto a ponto entre o rumo previsto e o rumo real da embarcação. Desta forma, uma previsão de k pontos de uma trajetória gera k valores de erro correspondentes, calculados em graus.

Um detalhe importante é que, como o rumo varia de 0° a 360° , a simples subtração entre os ângulos pode gerar distorções quando as embarcações navegam próximo a 0° ou 360° . Para evitar esse problema, a diferença angular foi calculada subtraindo-se o maior rumo pelo menor; caso essa diferença fosse maior que 180° , utilizava-se o complemento do ângulo para obter a menor diferença possível entre os rumos.

3.4.6 Métricas para etapa de classificação

Para a análise de desempenho dos algoritmos de classificação, foram utilizadas quatro métricas: acurácia, taxa de verdadeiros positivos, taxa de falsos positivos e *F1-score*.

A acurácia, calculada conforme a Equação 3.10, representa a proporção de classificações corretas realizadas pelo algoritmo em relação ao total de classificações feitas. A taxa de verdadeiros positivos, calculada conforme a Equação 3.12, expressa a porcentagem de classificações corretas entre os casos que deveriam ser classificados como positivos.

A taxa de falsos positivos, calculada conforme a Equação 3.11, indica a proporção de classificações incorretas como positivas em relação ao total de casos que deveriam ser classificados como negativos.

Por fim, o *F1-score*, calculado conforme a Equação 3.13, é uma métrica que equilibra a importância entre a taxa de verdadeiros positivos e a taxa de falsos positivos, sendo especialmente útil para avaliação de algoritmos de classificação em cenários com classes desbalanceadas.

$$\text{Acurácia} = \frac{\text{Previsões corretas}}{\text{Total de previsões}} \quad (3.10)$$

$$\text{Taxa de falso positivo} = \frac{\text{Falsos Positivos}}{\text{Falsos Positivos} + \text{Verdadeiros Negativos}} \quad (3.11)$$

$$\text{Taxa de verdadeiro positivo} = \frac{\text{Verdadeiros Positivos}}{\text{Verdadeiros Positivos} + \text{Falso Negativo}} \quad (3.12)$$

$$\text{F1 - score} = \frac{\text{Verdadeiros Positivos}}{\text{Verdadeiros Positivos} + \frac{1}{2}(\text{Falso Positivo} + \text{Falso Negativo})} \quad (3.13)$$

3.5 Implementação

Todo o software desenvolvido neste trabalho foi implementado em linguagem *Python* com o auxílio da biblioteca *Pandas* e *GeoPandas* para manipulação dos conjuntos de dados, *PyTorch* para manipulação das redes neurais, *scikit-learn* para para os algoritmos de classificação tradicionais utilizados (*Random Forest* e *SVM*) e métricas para os modelos de classificação, *Scipy* para implementação do teste estatístico do

modelo de classificação MB, além da biblioteca *NumPy* para manipulação numérica em geral.

Capítulo 4

Resultados e Discussões

4.1 Resultados das etapas de previsão de trajetórias

A etapa de previsão das trajetórias treinou dois modelos de rede neural com arquitetura *TrAISformer* para cada um dos cinco conjuntos de dados: um modelo para realizar previsão "para frente" e outro para realizar previsão "para trás". Utilizando os dados separados para teste, foram obtidos os resultados apresentados nas Figuras 4.1, 4.3 e 4.2.

A Figura 4.1 apresenta as métricas de desempenho das previsões de posição realizadas pelos modelos. As subfiguras 4.1(a) e 4.1(b) mostram o erro médio de posição (**EM**, Seções 3.4.1 e 3.4.3) comparando as trajetórias previstas e as trajetórias realizadas pelas embarcações ao longo do tempo para cada um dos modelos treinados. Nessas figuras, observa-se que o erro médio das previsões aumenta à medida que o tempo de projeção avança, o que é um comportamento esperado.

As subfiguras 4.1(c) e 4.1(d) exibem diagramas do tipo *boxplot* (Seção 3.4.2), em escala logarítmica, do erro de posição (Seção 3.4.3) medido entre a trajetória prevista e a trajetória realizada para o instante de tempo de 1 hora em cada um dos conjuntos de dados ($Err_{i,k=1h}$). Da mesma forma, as subfiguras 4.1(e) e 4.1(f) apresentam gráficos do tipo *boxplot*, também em escala logarítmica, do erro de posição entre a trajetória prevista e a trajetória realizada para o instante de tempo de 3 horas para cada conjunto de dados ($Err_{i,k=3h}$).

Observando os resultados apresentados, nota-se que, em todos os casos analisados, a mediana dos erro de posição ($Err_{i,k}$) nos instantes de 1 e 3 horas (apresentados nos *boxplots* 4.1(c), 4.1(d), 4.1(e) e 4.1(f)) são menores que o erro médio em todos os cinco conjuntos de dados (apresentados nas Figuras 4.1(a) e 4.1(b)), devido à influência do grande número de *outliers* superiores presentes nas previsões.

Ao comparar os resultados dos modelos treinados com os conjuntos de dados 2 e 3, observa-se uma melhora no desempenho do modelo 3 em relação ao modelo

2. Como a única diferença entre esses dois modelos é o tempo entre as amostras das trajetórias, esse resultado sugere que o intervalo entre amostras pode afetar significativamente o desempenho da arquitetura *TrAISformer*, intervalos menores entre amostras aumentam a precisão dos resultados.

Outra conclusão importante pode ser obtida ao comparar o modelo do conjunto de dados 1 com o modelo 2. Apesar de os conjuntos de dados 1 e 2 terem sido coletados no mesmo período de tempo (3 meses), o conjunto de dados 1 contém um número muito maior de trajetórias do que o conjunto de dados 2. Isso indica que a região correspondente ao conjunto de dados 1 possui um tráfego de embarcações mais intenso, sendo, portanto, esperado que o modelo treinado com esse conjunto tenha um desempenho superior ao modelo treinado com o conjunto 2, uma vez que ambos utilizam o mesmo intervalo entre amostras (10 minutos). Esse resultado esperado ocorre apenas para previsões curtas, de até aproximadamente 2,5 a 3 horas. A partir desse ponto, o erro médio de distância (**EM**, Seções 3.4.1 e 3.4.3), Figuras 4.1(a) e 4.1(b)) do modelo 1 passa a ser maior do que a do modelo 2. O mesmo fenômeno é observado ao comparar o modelo 1 com o modelo 3.

Além disso, ao analisar os gráficos de *boxplot*, observa-se que o modelo 1 apresenta maior variabilidade nos resultados em comparação com os modelos 2 e 3, incluindo um maior número de *outliers*. Considerando a área geográfica de onde esses dados foram coletados, conforme apresentado na Figura 3.4, nota-se que a região do conjunto de dados 1 possui uma maior complexidade de navegação devido à presença de ilhas e múltiplos destinos. Esse fator obriga as embarcações a realizarem trajetórias mais complexas em comparação com as trajetórias das regiões dos conjuntos de dados 2 e 3.

Dessa forma, a maior variabilidade dos resultados do modelo 1 e sua maior dificuldade em realizar previsões de longo prazo, em comparação com os modelos 2 e 3, podem ser explicadas pela complexidade geográfica para navegação da região correspondente ao conjunto de dados 1. Assim, os resultados indicam que a arquitetura *TrAISformer* pode apresentar um desempenho inferior em regiões de navegação mais complexa.

Uma última conclusão pode ser obtida analisando os resultados da métrica de distância, ao comparar os desempenhos dos modelos 4 e 5. Esses dois conjuntos de dados possuem a maior quantidade de trajetórias em comparação com os demais conjuntos, o que, como esperado, resultou nos melhores desempenhos. No entanto, uma conclusão mais relevante pode ser extraída ao comparar diretamente esses dois modelos.

Os conjuntos de dados 4 e 5 são idênticos, ambos possuem um tempo entre amostras de 2 minutos, diferenciando-se apenas no tamanho da trajetória inicial utilizada para que o modelo inicie as previsões. No conjunto de dados 4, a trajetória

inicial possui 3 horas, de modo que, para uma previsão de 4 horas, a trajetória completa totaliza 7 horas, o que equivale a 210 amostras considerando a taxa de 2 minutos entre amostras. A arquitetura *TrAISformer* utilizada, no entanto, possui uma entrada limitada a apenas 120 amostras, correspondendo a 4 horas de trajetória nessa taxa de amostragem. Dessa forma, a partir de 1 hora de previsão, os dados da trajetória original na entrada do modelo começam a ser substituídos por amostras resultantes das próprias previsões, que acumulam erro ao longo do tempo.

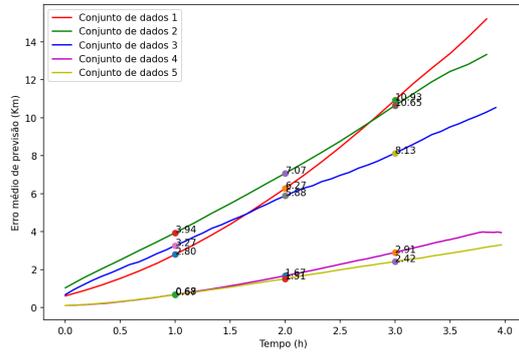
Já no conjunto de dados 5, a trajetória inicial utilizada possui 2 horas, ao invés de 3. Isso faz com que a remoção dos dados originais na entrada da rede neural ocorra apenas a partir de 2 horas de previsão, e não 1 hora como no conjunto 4. O impacto dessa diferença pode ser observado nos resultados apresentados na Figura 4.1, onde os modelos 4 e 5 apresentam desempenhos equivalentes para previsões de até 1 hora. No entanto, a partir desse ponto — quando o modelo 4 começa a substituir dados reais por previsões — o modelo 5 passa a apresentar um desempenho superior.

Com isso, conclui-se que, devido à limitação da rede neural quanto ao tamanho da entrada, trajetórias iniciais muito longas podem impactar negativamente o desempenho do modelo em previsões que ultrapassem o tamanho da entrada da rede.

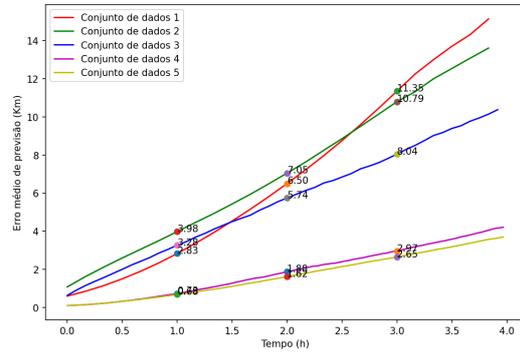
As métricas de desempenho dos modelos para previsão de rumo são exibidas na Figura 4.2. As Figuras 4.2(a) e 4.2(b) apresentam os erros médios de rumo (**EM**, Seções 3.4.1 e 3.4.5) comparando os rumos estimados pelos diversos modelos e os rumos executados pelas embarcações ao longo das trajetórias. Já as Figuras 4.2(c) e 4.2(d) exibem os diagramas *boxplot* (Seção 3.4.2), em escala logarítmica, dos erros de rumo ($Err_{i,k}$, Seção 3.4.5) entre os rumos previstos e os rumos reais das embarcações no instante de tempo de 1 hora para cada um dos modelos ($Err_{i,k=1h}$). Da mesma forma, as Figuras 4.2(e) e 4.2(f) mostram os *boxplots* do erro de rumo ($Err_{i,k}$, Seção 3.4.5) no instante de tempo de 3 horas para cada conjunto de dados ($Err_{i,k=3h}$).

O comportamento dos erros na previsão do rumo foi similar ao comportamento observado na análise do erro em distância, tanto para os erros médios (**EM**) quanto para os erros por amostra ($Err_{i,k}$, apresentando características semelhantes na comparação entre os modelos. No entanto, observando os diagramas de *boxplot*, nota-se que, na maioria dos casos, o erro médio de rumo (EM_k) é pelo menos duas vezes, e em alguns casos até três vezes, maior que a mediana, devido à influência de um grande número de *outliers*. Esta característica sugere uma maior dificuldade do modelo em estimar os rumos realizados pelas embarcações ao longo das trajetórias se comparado com as estimativas de posição.

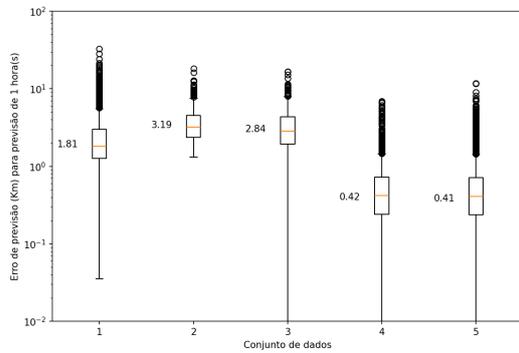
As métricas de desempenho dos modelos para estimativa de velocidade são exibidas na Figura 4.3. As Figuras 4.3(a) e 4.3(b) apresentam o erro médio (**EM**,



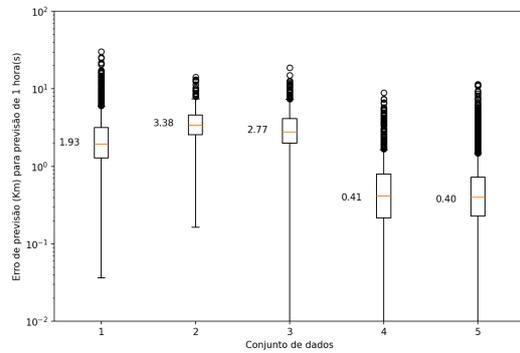
(a) Previsão "para frente".



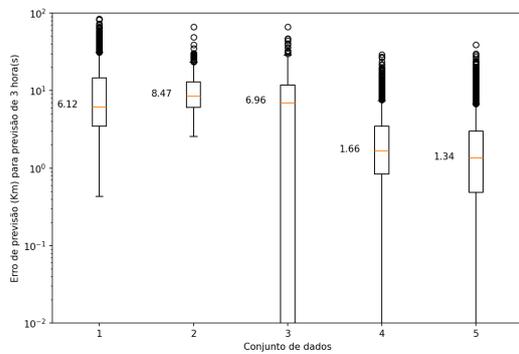
(b) Previsão "para trás".



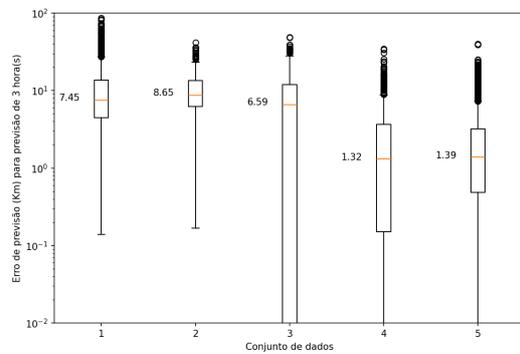
(c) Previsão "para frente" no instante de 1 hora.



(d) Previsão "para trás" no instante de 1 hora.

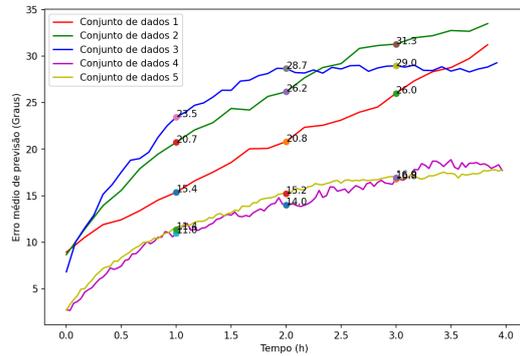


(e) Previsão "para frente" no instante de 3 horas.

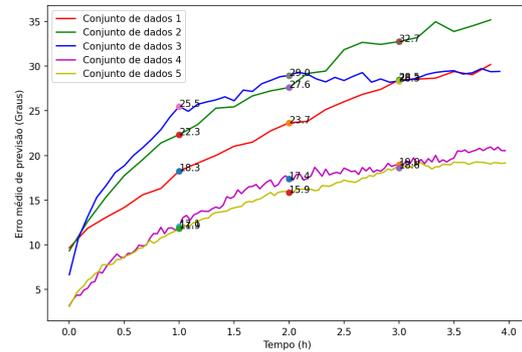


(f) Previsão "para trás" no instante de 3 horas.

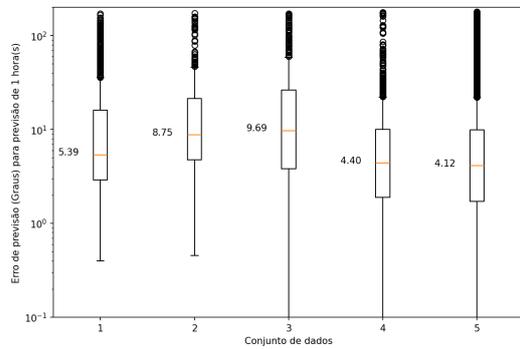
Figura 4.1: Resultados dos modelos de previsão de trajetória. As Figuras 4.1(a) e 4.1(b) exibem, no eixo y, o erro médio de posição (EM_k), em Km, sendo o eixo x o instante de tempo da trajetória prevista (equivalente de k em horas), em horas. Os itens 4.1(c) e 4.1(d) exibem o *boxplot* dos erros de posição, em Km, de cada uma das trajetórias para o instante de previsão de 1 hora ($Err_{i,k=1h}$) sendo o eixo x o número do conjunto de dados e os itens 4.1(e) e 4.1(f) exibem o *boxplot* dos erros de posição, em Km, de cada uma das trajetórias para o instante de previsão de 3 hora ($Err_{i,k=3h}$) sendo o eixo x o número do conjunto de dados.



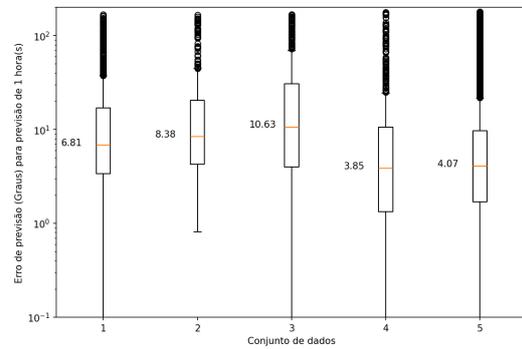
(a) Previsão do rumo "para frente".



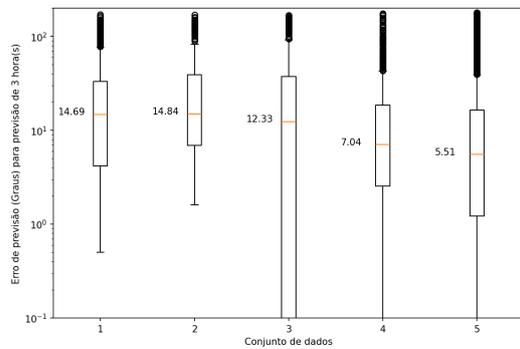
(b) Previsão do rumo "para trás".



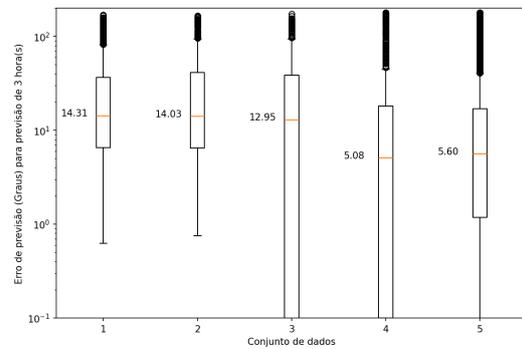
(c) Previsão do rumo "para frente" no instante de 1 hora.



(d) Previsão do rumo "para trás" no instante de 1 hora.

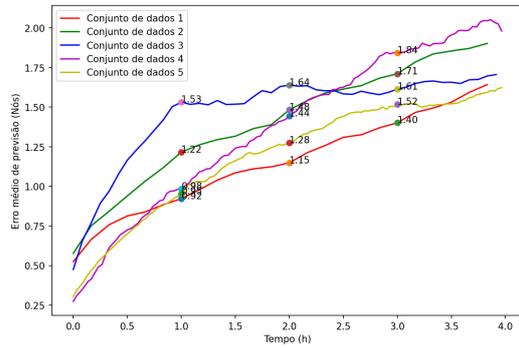


(e) Previsão do rumo "para frente" no instante de 3 horas.

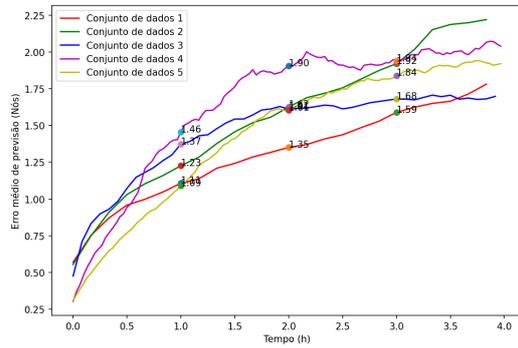


(f) Previsão do rumo "para trás" no instante de 3 horas.

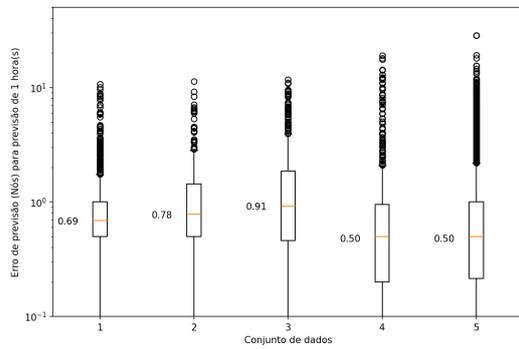
Figura 4.2: Resultados de previsão do rumo dos modelos de previsão de trajetória. Os itens 4.2(a) e 4.2(b) exibem, no eixo y, o erro médio do rumo (EM_k), em graus, sendo o eixo x o instante de tempo da trajetória prevista (equivalente de k em horas), em horas. Os itens 4.2(c) e 4.2(d) exibem o *boxplot* dos erros de rumo, em graus, de cada uma das trajetórias para o instante de previsão de 1 hora ($Err_{i,k=1h}$) sendo o eixo x o número do conjunto de dados e os itens 4.2(e) e 4.2(f) exibem o *boxplot* dos erros de rumo, em graus, de cada uma das trajetórias para o instante de previsão de 3 hora ($Err_{i,k=3h}$) sendo o eixo x o número do conjunto de dados.



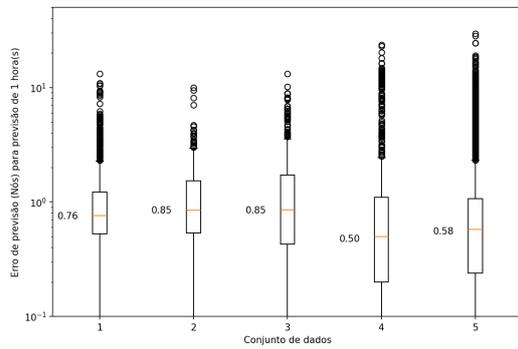
(a) Previsão da Velocidade "para frente".



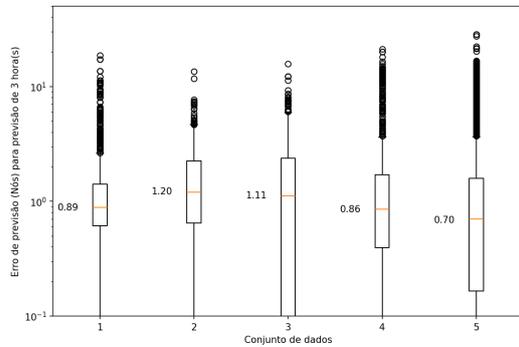
(b) Previsão da velocidade "para trás".



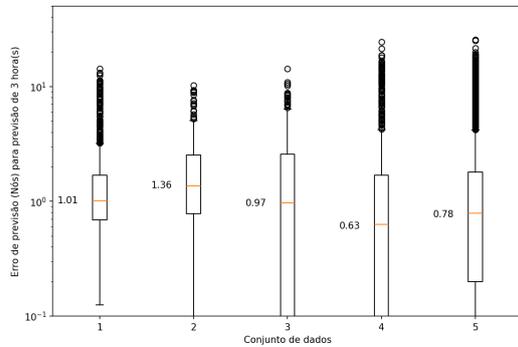
(c) Previsão da Velocidade "para frente" no instante de 1 hora.



(d) Previsão da velocidade "para trás" no instante de 1 hora.



(e) Previsão da Velocidade "para frente" no instante de 3 horas.



(f) Previsão da velocidade "para trás" no instante de 3 horas.

Figura 4.3: Resultados de previsão de velocidade dos modelos de previsão de trajetória. Os itens 4.3(a) e 4.3(b) exibem, no eixo y, o erro médio de velocidade (EM_k), em nós, sendo o eixo x o instante de tempo da trajetória prevista (equivalente de k em horas), em horas. Os itens 4.3(c) e 4.3(d) exibem o *boxplot* dos erros de velocidade, em nós, de cada uma das trajetórias para o instante de previsão de 1 hora ($Err_{i,k=1h}$) sendo o eixo x o número do conjunto de dados e os itens 4.3(e) e 4.3(f) exibem o *boxplot* dos erros de velocidade, em nós, de cada uma das trajetórias para o instante de previsão de 3 hora ($Err_{i,k=3h}$) sendo o eixo x o número do conjunto de dados.

Seções 3.4.1 e 3.4.4) das estimativas de velocidade comparando as velocidades previstas pelos diversos modelos e as velocidades reais das embarcações ao longo das trajetórias. Já as Figuras 4.3(c) e 4.3(d) exibem os diagramas *boxplot* (Seção 3.4.2), em escala logarítmica, dos erros de velocidade ($Err_{i,k}$, Seção 3.4.4) entre as velocidades previstos e as velocidades reais das embarcações no instante de tempo de 1 hora para cada um dos modelos ($Err_{i,k=1h}$). Da mesma forma, as Figuras 4.3(e) e 4.3(f) mostram os *boxplots* deste mesmo erro de velocidade para o instante de tempo de 3 horas para cada um dos cinco conjuntos de dados ($Err_{i,k=3h}$).

As análises dos *boxplots* (Figuras 4.3(c), 4.3(d), 4.3(e) e 4.3(f)) mostram que os modelos 4 e 5 apresentam uma mediana menor em comparação com os demais modelos, enquanto o modelo 3, em geral, exibe melhorias em relação ao modelo 2. No entanto, os gráficos de erro médio (**EM** de velocidade (Figuras 4.3(a) e 4.3(b)) exibem uma dificuldade maior em distinguir quais modelos apresentaram melhores resultados. Isso mostra uma maior dificuldade do modelo em realizar estimativas para a variável velocidade, possivelmente devido à variável velocidade ser uma variável instantânea e influenciada por fatores externos, como correntes marítimas e condições meteorológicas.

4.2 Resultados da etapa de classificação

Com o objetivo de analisar a eficácia da aplicação dos algoritmos de classificação na solução do cenário estudado neste trabalho, foram utilizados os algoritmos *Random Forest* (RF), *Support Vector Machine* (SVM) e o algoritmo desenvolvido pela Marinha do Brasil (MB) nos conjuntos de pares de trajetórias obtidos conforme o procedimento descrito na Seção 3.3.

Além da aplicação direta dos algoritmos, foram exploradas técnicas de normalização e redução dos atributos (*features*) de entrada, visando aprimorar o desempenho da classificação. Entre as técnicas empregadas, foi testada a normalização logarítmica, a qual pode ser realizada aplicando a Equação 4.1 às *features* dos conjuntos de dados. Esta técnica foi aplicada em todas as *features* dos conjuntos de dados.

$$y' = \log_{10}(y + 1) \quad (4.1)$$

Conforme apresentado na Tabela 3.3, o número de *features* nos conjuntos de dados de pares de trajetórias é bastante elevado. Dessa forma, foram testadas duas técnicas diferentes para reduzir o número de *features* sem comprometer o desempenho dos algoritmos de classificação.

A primeira técnica consistiu na utilização de estatísticas resumidas dos dados em vez de todas as variáveis originais. Foram consideradas as seguintes estatísticas:

Tipo de pré-processamento	Descrição
Completo	Utilização de todas as <i>features</i> extraídas (descrita na seção 3.3) sem nenhum pré-processamento
Normalização logarítmica	Utilização de todas as <i>features</i> extraídas aplicando a normalização logarítmica
10 <i>features</i> estatísticas	Utilização de dez estatísticas calculadas a partir das <i>features</i> extraídas: máximo, mínimo, média e desvio padrão das <i>features</i> de distância, máximo, mínimo e média das <i>features</i> de diferença de velocidade e rumo.
Normalização + SFS 10 <i>features</i>	Aplicando a normalização logarítmica em todas as <i>features</i> e posterior aplicação do algoritmo SFS para seleção das 10 <i>features</i> mais relevantes

Tabela 4.1: Descrição dos tipos de pré-processamentos utilizados nas *features* extraídas na etapa de classificação

máximo, mínimo, média e desvio padrão das distâncias ponto a ponto entre as trajetórias; máximo, mínimo e desvio padrão da diferença de velocidade ponto a ponto entre as trajetórias; e máximo, mínimo e desvio padrão da diferença de rumo ponto a ponto entre as trajetórias, totalizando 10 variáveis.

A segunda técnica empregada foi o uso do algoritmo *Sequential Feature Selection* (SFS) para a seleção automática de 10 *features*, em substituição ao conjunto completo de variáveis.

Além disso, todos os experimentos foram conduzidos utilizando a técnica de validação cruzada *5-Fold* para a seleção dos hiperparâmetros dos modelos de classificação.

A Tabela 4.1 apresenta um resumo explicativo dos diferentes pré-processamentos utilizados.

As Tabelas 4.2, 4.3, 4.4 e 4.5 apresentam, respectivamente, as métricas de acurácia, taxa de verdadeiros positivos, taxa de falsos positivos e *F1-score* obtidas nos modelos de classificação para cada um dos cinco conjuntos de dados, considerando as diferentes técnicas de pré-processamento testadas.

De forma geral, os melhores resultados foram obtidos com o conjunto de dados 5, seguido pelos conjuntos 4, 1, 3 e 2. Esse comportamento sugere uma relação direta entre o desempenho das previsões na primeira etapa da solução proposta, conforme apresentado na Seção 4.1, e os resultados obtidos na etapa de classificação. Assim, verifica-se que, quanto melhores as previsões, melhores tendem a ser os resultados da classificação, conforme esperado.

No cenário de utilização dos dados sem nenhum pré-processamento (classificado como "Completo" na tabela), os melhores resultados foram obtidos, em todas as métricas, com a aplicação do algoritmo *Random Forest*, quando comparado tanto

		Dados 1	Dados 2	Dados 3	Dados 4	Dados 5
Completo	RF	0,84	0,75	0,76	0,93	0,96
	SVM	0,80	0,71	0,72	0,90	0,94
	MB	0,72	0,74	0,79	0,83	0,75
Normalização Logarítmica	RF	0,84	0,79	0,77	0,93	0,96
	SVM	0,84	0,75	0,81	0,93	0,94
	MB	0,73	0,62	0,73	0,82	0,83
10 <i>features</i> estatísticas	RF	0,78	0,76	0,74	0,90	0,93
	SVM	0,79	0,75	0,78	0,88	0,94
Normalização + SFS 10 <i>features</i>	RF	0,83	0,77	0,80	0,93	0,96
	SVM	0,81	0,76	0,79	0,93	0,95

Tabela 4.2: Esta tabela apresenta a acurácia obtida nos modelos de classificação utilizando diferentes pré-processamentos nos dados em cada um dos 5 conjuntos de dados. Demarcados em negrito são os maiores valores de acurácia obtido em cada um dos conjuntos de dados.

		Dados 1	Dados 2	Dados 3	Dados 4	Dados 5
Completo	RF	0,87	0,75	0,78	0,95	0,96
	SVM	0,84	0,80	0,69	0,93	0,94
	MB	0,71	0,73	0,78	0,83	0,75
Normalização Logarítmica	RF	0,88	0,83	0,79	0,94	0,96
	SVM	0,88	0,81	0,76	0,94	0,95
	MB	0,75	0,63	0,73	0,81	0,86
10 <i>features</i> estatísticas	RF	0,82	0,79	0,76	0,91	0,93
	SVM	0,90	0,87	0,87	0,94	0,93
Normalização + SFS 10 <i>features</i>	RF	0,87	0,81	0,80	0,95	0,96
	SVM	0,84	0,79	0,79	0,95	0,95

Tabela 4.3: Esta tabela apresenta a taxa de verdadeiro positivo obtida nos modelos de classificação utilizando diferentes pré-processamentos nos dados em cada um dos 5 conjuntos de dados. Demarcados em negrito são os maiores valores da taxa de verdadeiro positivo obtido em cada um dos conjuntos de dados.

		Dados 1	Dados 2	Dados 3	Dados 4	Dados 5
Completo	RF	0,23	0,25	0,26	0,09	0,04
	SVM	0,28	0,49	0,20	0,13	0,06
	MB	0,28	0,25	0,20	0,17	0,25
Normalização Logarítmica	RF	0,23	0,28	0,25	0,09	0,04
	SVM	0,22	0,36	0,11	0,09	0,06
	MB	0,30	0,40	0,28	0,16	0,43
10 <i>features</i> estatísticas	RF	0,29	0,31	0,28	0,11	0,07
	SVM	0,41	0,44	0,33	0,19	0,06
Normalização + SFS 10 <i>features</i>	RF	0,23	0,31	0,18	0,10	0,04
	SVM	0,25	0,28	0,18	0,10	0,05

Tabela 4.4: Esta tabela apresenta a taxa de falso positivo obtida nos modelos de classificação utilizando diferentes pré-processamentos nos dados em cada um dos 5 conjuntos de dados. Demarcados em negrito são os menores valores da taxa de falso positivo obtido em cada um dos conjuntos de dados.

		Dados 1	Dados 2	Dados 3	Dados 4	Dados 5
Completo	RF	0,82	0,75	0,76	0,93	0,96
	SVM	0,79	0,70	0,73	0,90	0,94
	MB	0,71	0,74	0,79	0,83	0,75
Normalização Logarítmica	RF	0,83	0,79	0,77	0,93	0,96
	SVM	0,84	0,75	0,81	0,93	0,94
	MB	0,73	0,62	0,73	0,82	0,75
10 <i>features</i> estatísticas	RF	0,78	0,75	0,75	0,90	0,93
	SVM	0,78	0,75	0,79	0,88	0,93
Normalização + SFS 10 <i>features</i>	RF	0,83	0,76	0,81	0,93	0,96
	SVM	0,80	0,76	0,80	0,93	0,95

Tabela 4.5: Esta tabela apresenta a taxa de f1-score obtida nos modelos de classificação utilizando diferentes pré-processamentos nos dados em cada um dos 5 conjuntos de dados. Demarcados em negrito são os maiores valores de f1-score obtido em cada um dos conjuntos de dados.

ao algoritmo SVM, que apresentou o segundo melhor desempenho, quanto ao algoritmo MB (desenvolvido pela Marinha do Brasil). As únicas exceções ocorreram nos conjuntos de dados 2 e 3, onde o algoritmo SVM obteve uma maior taxa de verdadeiros positivos para o conjunto de dados 2, e os algoritmos MB e SVM apresentaram uma menor taxa de falsos positivos para o conjunto de dados 3 em comparação com o *Random Forest*, resultando em um *F1-score* superior para o algoritmo MB neste caso. Esses resultados sugerem que, em cenários com menor disponibilidade de dados, como nos conjuntos de dados 2 e 3, o algoritmo MB pode se sobressair em relação aos demais algoritmos testados.

A utilização da normalização logarítmica nos dados não resultou em alterações significativas nas métricas do *Random Forest* na maioria dos conjuntos de dados, com exceção do conjunto 2, onde foi observado um aumento na taxa de verdadeiros positivos, uma redução na taxa de falsos negativos e, conseqüentemente, um aumento no *F1-score*. Para o classificador SVM, a normalização logarítmica resultou em uma melhora ou equivalência dos resultados em todas as métricas e conjuntos de dados, com destaque para a redução da taxa de falsos positivos e conseqüente aumento no *F1-score*. Essa melhora pode ser atribuída à capacidade da normalização logarítmica de reduzir o impacto dos *outliers*, conforme observado nos resultados de previsão de trajetórias (Seção 4.1), aproximando numericamente os dados e beneficiando o algoritmo SVM, uma vez que esses conjuntos de dados apresentam maior dispersão numérica. Já para o algoritmo MB, a normalização logarítmica apresentou resultados mistos, com melhorias em alguns conjuntos de dados e piora em outros.

A utilização das 10 *features* estatísticas no *Random Forest* resultou em uma piora ou, no máximo, equivalência dos resultados em todas as métricas quando comparado tanto aos dados brutos quanto aos dados normalizados, com destaque para a significativa piora na taxa de falsos positivos em todos os conjuntos de dados. A única exceção ocorreu no conjunto de dados 2, onde houve uma melhora na taxa de verdadeiros positivos em relação aos dados completos, mas uma leve redução em comparação com os dados normalizados. Para o algoritmo SVM, de modo geral, houve uma melhora na taxa de verdadeiros positivos e uma piora significativa na taxa de falsos positivos em relação tanto aos dados completos quanto aos normalizados. Isso resultou em um desempenho misto nas métricas de acurácia e *F1-score*, com melhorias nos conjuntos de dados 2 e 3 e pioras nos conjuntos de dados 1, 4 e 5. Considerando a significativa redução do número de atributos em relação aos conjuntos de dados completos, o uso das 10 variáveis estatísticas demonstra potencial para cenários em que há restrições computacionais para o processamento dos algoritmos de classificação.

A combinação da normalização logarítmica com o algoritmo *Sequential Feature Selection* (SFS) para a redução da quantidade de atributos para 10 permitiu uma

redução dos atributos sem impactar significativamente os resultados obtidos com os dados normalizados, que apresentaram o melhor desempenho no experimento. De modo geral, foram observadas variações pouco significativas, tanto positivas quanto negativas, ao utilizar os algoritmos SVM e *Random Forest*, com destaque para a redução da taxa de falsos positivos no conjunto de dados 2 ao utilizar o algoritmo SVM em comparação com os dados normalizados. Assim, a aplicação do algoritmo SFS nos dados normalizados demonstrou ser uma técnica eficaz para a redução de atributos, preservando o desempenho dos classificadores nesta aplicação específica. A redução de atributos é um aspecto relevante neste caso, dado o alto número de variáveis presentes nos dados utilizados neste trabalho, conforme evidenciado na Tabela 3.3.

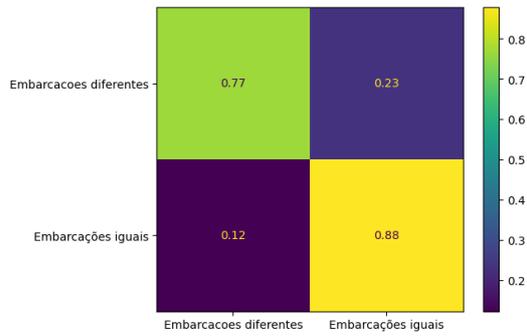
A Figura 4.4 exibe as matrizes de confusão dos classificadores que apresentaram a maior acurácia para cada um dos conjuntos de dados utilizados no experimento. Os classificadores com os melhores resultados para cada conjunto de dados estão assinalados em negrito na Tabela 4.2.

Com base nos resultados obtidos no experimento de comparação entre os algoritmos de classificação, foi conduzido um experimento adicional para analisar o impacto da duração do *GAP* (tempo em que o equipamento AIS da embarcação permanece desligado) no desempenho dos diferentes algoritmos de classificação utilizados. Foram testados *GAPs* variando entre 0,5 e 7 horas, e os resultados estão exibidos nos gráficos da Figura 4.5. Os gráficos apresentam os resultados obtidos utilizando os algoritmos *Random Forest*, SVM e MB, considerando, em todos os casos, os dados normalizados com a normalização logarítmica, conforme realizado no experimento anterior. No eixo y, é apresentada a métrica *F1-score* obtida, enquanto no eixo x é representado o *GAP* medido em horas.

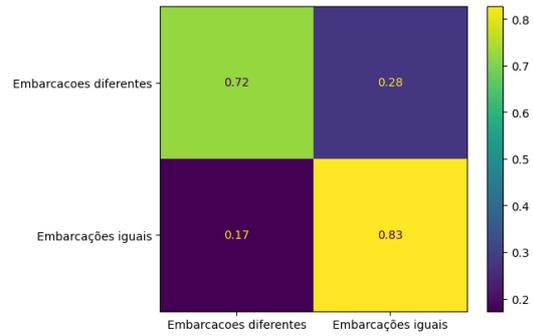
Observa-se, nesses gráficos, que, de modo geral, o *F1-score* dos modelos de classificação diminui à medida que o *GAP* aumenta. Além disso, os três algoritmos apresentam um comportamento similar nos gráficos, sugerindo que nenhum deles possui imunidade ao aumento do *GAP*. Esse resultado indica que a indisponibilidade dos dados AIS por períodos prolongados impacta negativamente o desempenho da classificação, independentemente do algoritmo utilizado.

Também foi analisado o efeito do *GAP* para cada um dos pré-processamentos realizados neste trabalho. Para essa análise, foi selecionado o algoritmo *Random Forest*, ao qual foram aplicados os mesmos quatro pré-processamentos utilizados no experimento anterior. O comportamento dos classificadores foi analisado para diferentes valores de *GAP*, variando entre 0,5 e 7 horas.

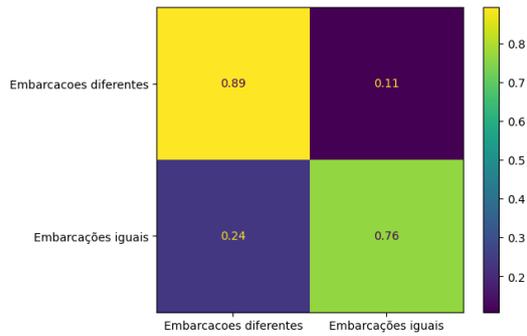
A Figura 4.6 apresenta os resultados obtidos, onde o eixo x representa o *GAP* em horas e o eixo y a métrica *F1-score*. Observa-se que, em todos os casos, os resultados seguem um comportamento similar, sugerindo que não há diferenças estatisticamente



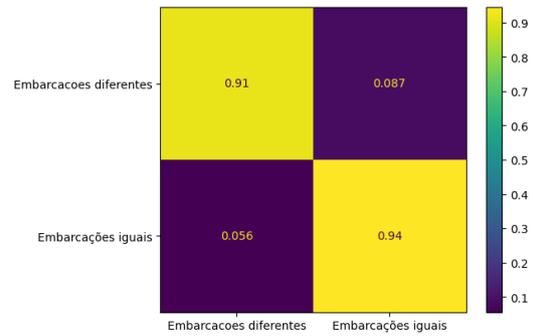
(a) Conjunto de dados 1



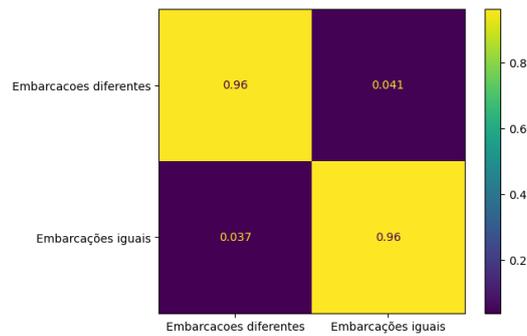
(b) Conjunto de dados 2



(c) Conjunto de dados 3

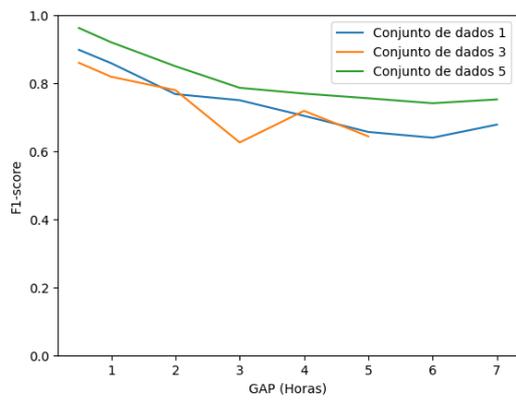


(d) Conjunto de dados 4

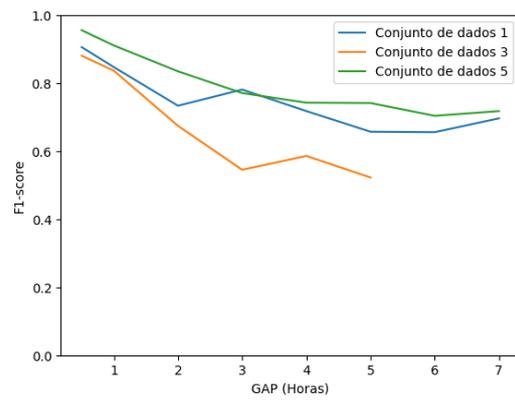


(e) Conjunto de dados 5

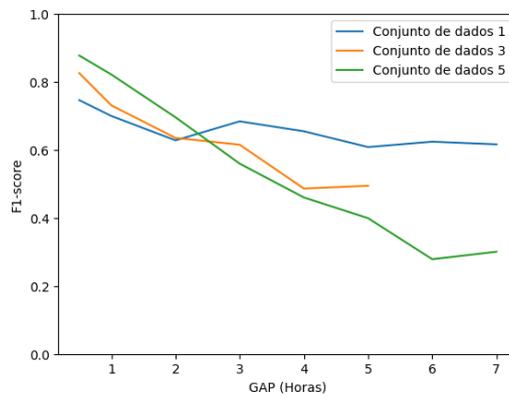
Figura 4.4: Matrizes de confusão com os resultados obtidos pelos classificadores com melhor acurácia para cada um dos conjuntos de dados. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



(a) *Random Forest*



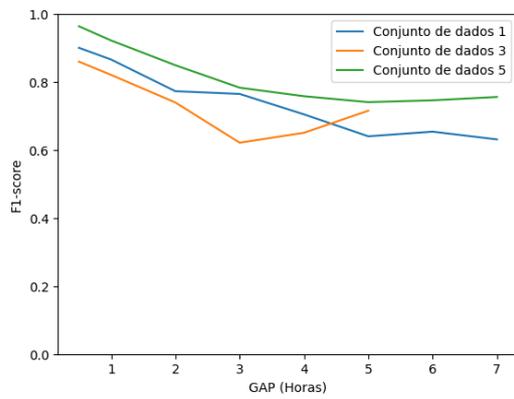
(b) *SVM*



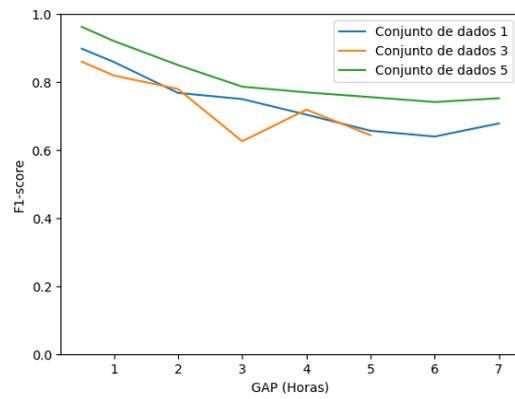
(c) *MB*

Figura 4.5: Resultado do experimento de análise do GAP com diferentes algoritmos.

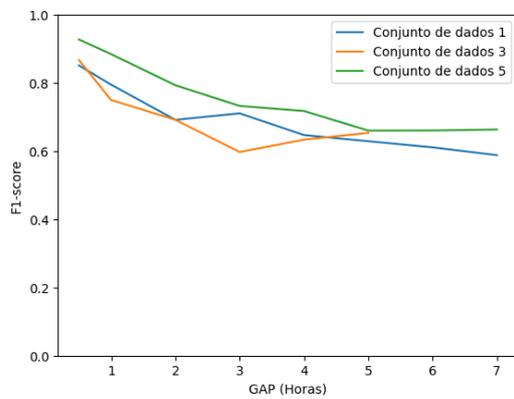
relevantes entre os diferentes pré-processamentos em relação ao aumento do GAP no contexto do experimento proposto.



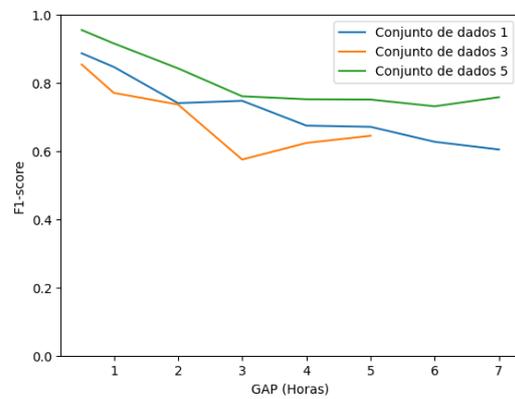
(a) Completo



(b) Normalização logarítmica



(c) 10 *features* estatísticas



(d) Normalização + SFS 10 *features*

Figura 4.6: Resultado do experimento de análise do GAP com diferentes pré-processamentos.

Capítulo 5

Conclusões

Esta dissertação apresentou um método para identificar e correlacionar trajetórias marítimas de embarcações em cenários de adulteração de identificadores no sistema AIS. A abordagem combinou técnicas de *Deep Learning* para previsão de trajetórias, utilizando-se da arquitetura *TrAISformer*, e algoritmos de classificação para identificar qual trajetória representa a continuidade da trajetória realizada por um navio que teve o seu equipamento AIS desligado e posteriormente religado utilizando um identificador adulterado. Isso demonstrou como soluções baseadas em inteligência artificial podem ser aplicadas de maneira robusta para resolver problemas críticos em sistemas de monitoramento do tráfego marítimo.

O principal objetivo deste trabalho foi desenvolver e validar um método que identificasse a continuidade de trajetórias de embarcações em casos de manipulação de identificadores, contribuindo para a segurança da navegação e combate a atividades ilícitas. O método proposto envolve duas etapas:

- **Previsão de trajetórias:** utilizando a arquitetura de rede neural *TrAISformer* e o histórico de navegação de diversas regiões, o método foi capaz de prever trajetórias com erro médio aceitável. Os experimentos mostraram que a precisão das previsões depende da quantidade de dados históricos, do intervalo entre amostras e da complexidade geográfica da região. O melhor desempenho foi observado em regiões menos complexas e com mais dados, com erros médios de 0,68 km, 1,51 km e 2,42 km para previsões de 1, 2 e 3 horas, respectivamente.
- **Classificação de trajetórias:** a segunda etapa utilizou três algoritmos (*Random Forest*, SVM e o modelo da Marinha do Brasil) para comparar pares de trajetórias — uma com previsão para frente e outra para trás — e decidir se pertencem à mesma embarcação. A acurácia obtida evidenciou a capacidade dos algoritmos em distinguir entre trajetórias reais e adulteradas, com destaque para o desempenho do *Random Forest* em diversos cenários.

A dissertação também analisou o impacto de diferentes parâmetros no desempenho dos modelos, como:

- o tempo entre amostras, que influenciou significativamente os resultados das previsões (modelo 3 superou o modelo 2 apenas por usar intervalo menor);
- a complexidade geográfica, que afetou a estabilidade e precisão do modelo em regiões com tráfego denso e obstáculos naturais (modelo 1 apresentou mais variabilidade e erros do que os modelos 2 e 3);
- o tamanho da trajetória inicial, que determinou o momento em que o modelo começa a substituir dados reais por previsões acumuladas, afetando diretamente o erro nas previsões longas (modelo 5 teve desempenho superior ao modelo 4 nesse aspecto).

Além disso, foram avaliadas estratégias de normalização e redução de atributos para melhorar a eficiência dos classificadores. A combinação de normalização logarítmica com seleção de atributos (SFS) mostrou-se promissora, especialmente em cenários com limitações computacionais.

Como perspectivas futuras, este trabalho pode ser ampliado com:

- a integração de dados de múltiplas fontes (como radar ou satélite) para complementar o AIS em situações de ocultamento intencional;
- o uso de arquiteturas mais recentes de redes neurais com maior capacidade de generalização;
- o estudo de abordagens que reduzam o erro acumulado nas previsões iterativas do modelo *TrAISformer*.

Com isso, conclui-se que o método proposto representa uma contribuição significativa para o monitoramento marítimo, oferecendo uma solução eficaz, flexível e robusta para a identificação de comportamentos suspeitos no tráfego de embarcações.

Referências Bibliográficas

- [1] OF HOMELAND SECURITY, U. D. “Automatic Identification System (AIS) Overview”. 2019. Disponível em: <<https://www.navcen.uscg.gov/automatic-identification-system-overview>>.
- [2] *SOLAS CHAPTER V: SAFETY OF NAVIGATION*. IMO, 2002.
- [3] HE, W., LEI, J., CHU, X., et al. “A Visual Analysis Approach to Understand and Explore Quality Problems of AIS Data”, *Journal of Marine Science and Engineering*, v. 9, n. 2, 2021. ISSN: 2077-1312. doi: 10.3390/jmse9020198. Disponível em: <<https://www.mdpi.com/2077-1312/9/2/198>>.
- [4] FORSBERG, J. *Cybersecurity of Maritime Communication Systems: Spoofing attacks against AIS and DSC*. Tese de Doutorado, 2022. Disponível em: <<https://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-189419>>.
- [5] KESSLER, G. C., ZORRI, D. M. “AIS Spoofing: A Tutorial for Researchers”. In: *2024 IEEE 49th Conference on Local Computer Networks (LCN)*, pp. 1–7, 2024. doi: 10.1109/LCN60385.2024.10639747.
- [6] ANDROJNA, A., PERKOVIC, M. “Impact of Spoofing of Navigation Systems on Maritime Situational Awareness”, *Transactions on Maritime Science*, v. 10, 09 2021. doi: 10.7225/toms.v10.n02.w08.
- [7] MAZZARELLA, F., VESPE, M., TARCHI, D., et al. “AIS reception characterisation for AIS on/off anomaly detection”. In: *2016 19th International Conference on Information Fusion (FUSION)*, pp. 1867–1873, 2016.
- [8] PRASAD, P., VATSAL, V., ROY CHOWDHURY, R. “Maritime Vessel Route Extraction and Automatic Information System (AIS) Spoofing Detection”. In: *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, pp. 1–11, 2021. doi: 10.1109/ICAECT49130.2021.9392536.
- [9] KAZIMIERSKI, W. “Proposal of neural approach to maritime radar and automatic identification system tracks association”, *IET Radar, Sonar*

- Navigation*, v. 11, n. 5, pp. 729–735, 2017. doi: <https://doi.org/10.1049/iet-rsn.2016.0409>. Disponível em: <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-rsn.2016.0409>.
- [10] SUN, W., LI, X., PANG, Z., et al. “Track-to-Track Association Based on Maximum Likelihood Estimation for T/R-R Composite Compact HFSWR”, *IEEE Transactions on Geoscience and Remote Sensing*, v. 61, pp. 1–12, 2023. doi: 10.1109/TGRS.2023.3253784.
- [11] ZHANG, T., ZHAO, S., CHENG, B., et al. “Detection of AIS Closing Behavior and MMSI Spoofing Behavior of Ships Based on Spatiotemporal Data”, *Remote Sensing*, v. 12, n. 4, 2020. ISSN: 2072-4292. doi: 10.3390/rs12040702. Disponível em: <https://www.mdpi.com/2072-4292/12/4/702>.
- [12] YU, J. Y., SGHAIER, M. O., GRABOWIECKA, Z. “Deep learning approaches for AIS data association in the context of maritime domain awareness”. In: *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, pp. 1–8, 2020. doi: 10.23919/FUSION45008.2020.9190283.
- [13] LI, H., JIAO, H., YANG, Z. “AIS data-driven ship trajectory prediction modeling and analysis based on machine learning and deep learning methods”, *Transportation Research Part E: Logistics and Transportation Review*, v. 175, pp. 103152, 2023. ISSN: 1366-5545. doi: <https://doi.org/10.1016/j.tre.2023.103152>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1366554523001400>.
- [14] SCHÖLLER, F. E., ENEVOLDSEN, T. T., BECKTOR, J. B., et al. “Trajectory Prediction for Marine Vessels using Historical AIS Heatmaps and Long Short-Term Memory Networks**This research is sponsored by the Danish Innovation Fund, The Danish Maritime Fund, Orients Fund and the Lauritzen Foundation through the Autonomy part of the ShippingLab project, Grant number 8090-00063B.” *IFAC-PapersOnLine*, v. 54, n. 16, pp. 83–89, 2021. ISSN: 2405-8963. doi: <https://doi.org/10.1016/j.ifacol.2021.10.077>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2405896321014798>. 13th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2021.
- [15] NGUYEN, D., FABLET, R. “A Transformer Network With Sparse Augmented Data Representation and Cross Entropy Loss for AIS-Based Vessel Trajectory Prediction”, *IEEE Access*, v. 12, pp. 21596–21609, 2024.

ISSN: 2169-3536. doi: 10.1109/access.2024.3349957. Disponível em:
<<http://dx.doi.org/10.1109/ACCESS.2024.3349957>>.

Apêndice A

Matriz de confusão dos modelos de classificação

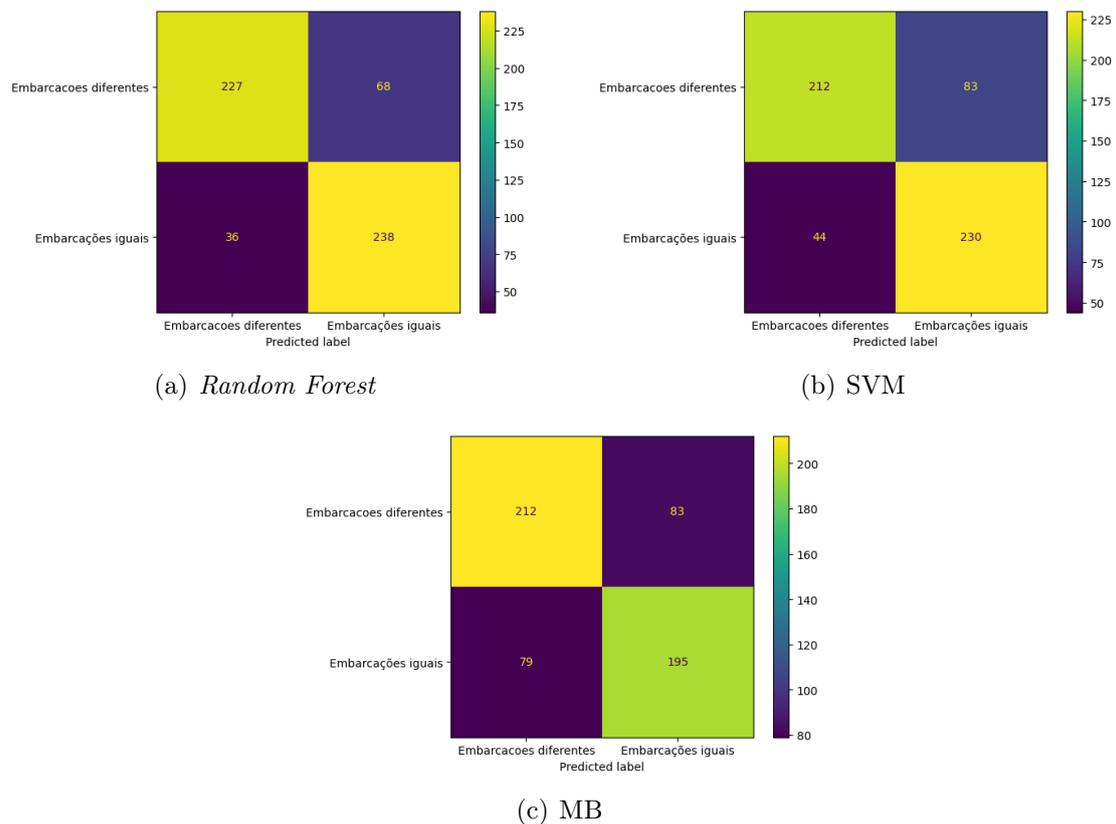


Figura A.1: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 1 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

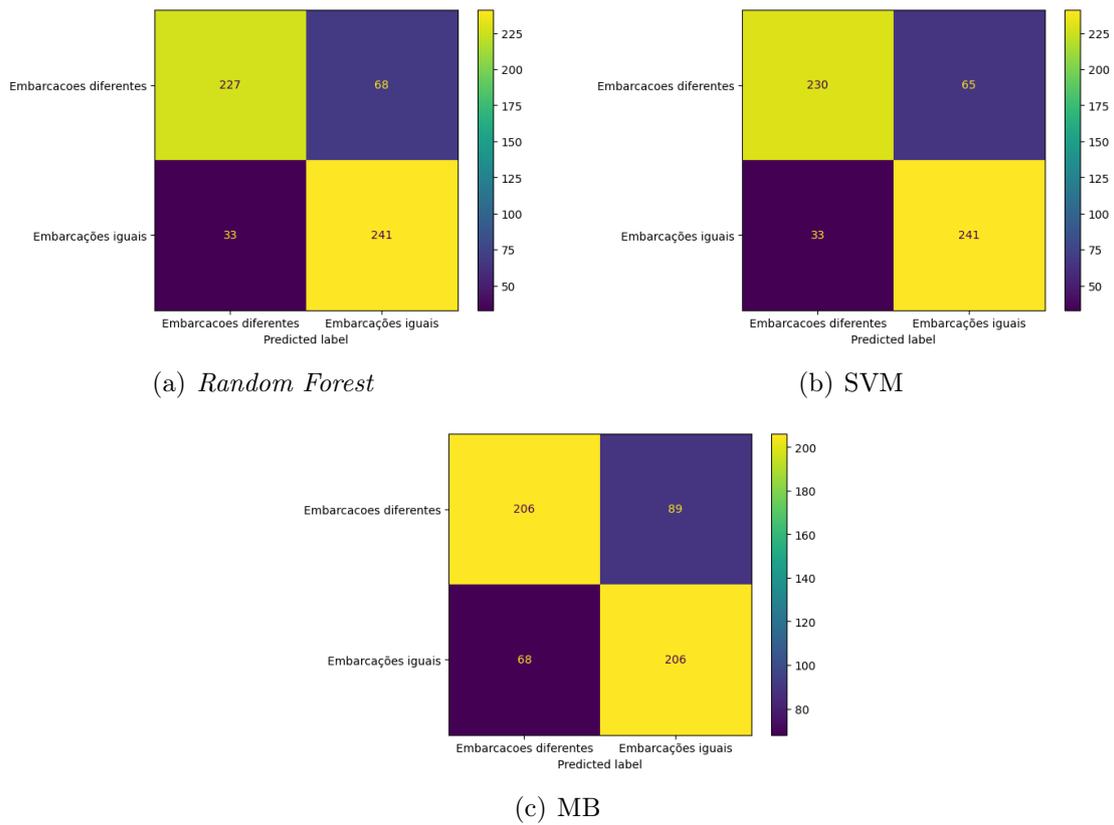


Figura A.2: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 1 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

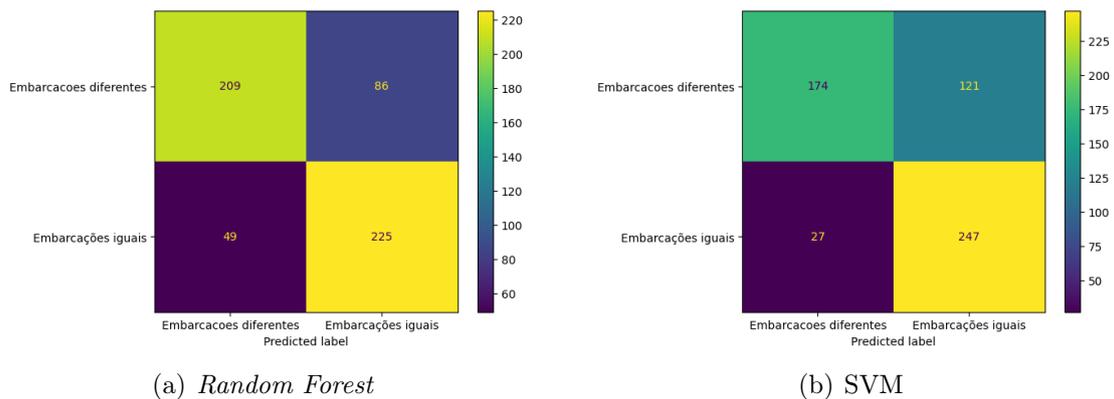


Figura A.3: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 1 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

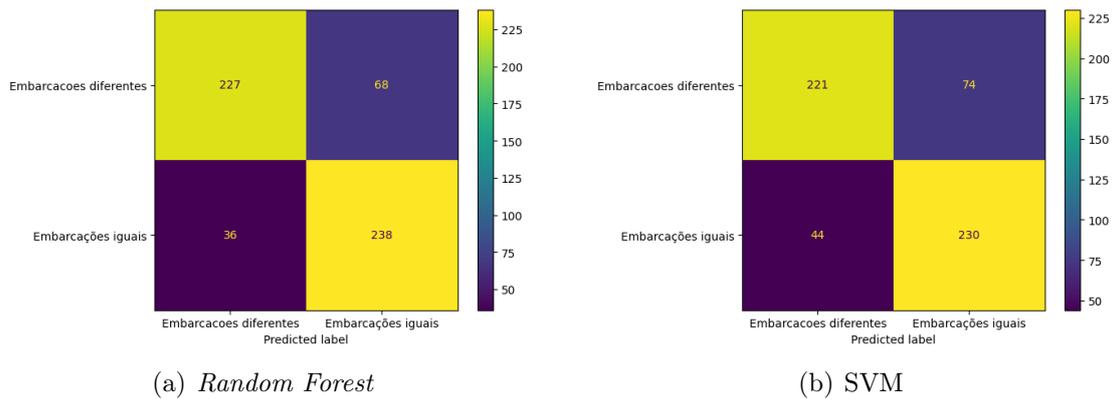


Figura A.4: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 1 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

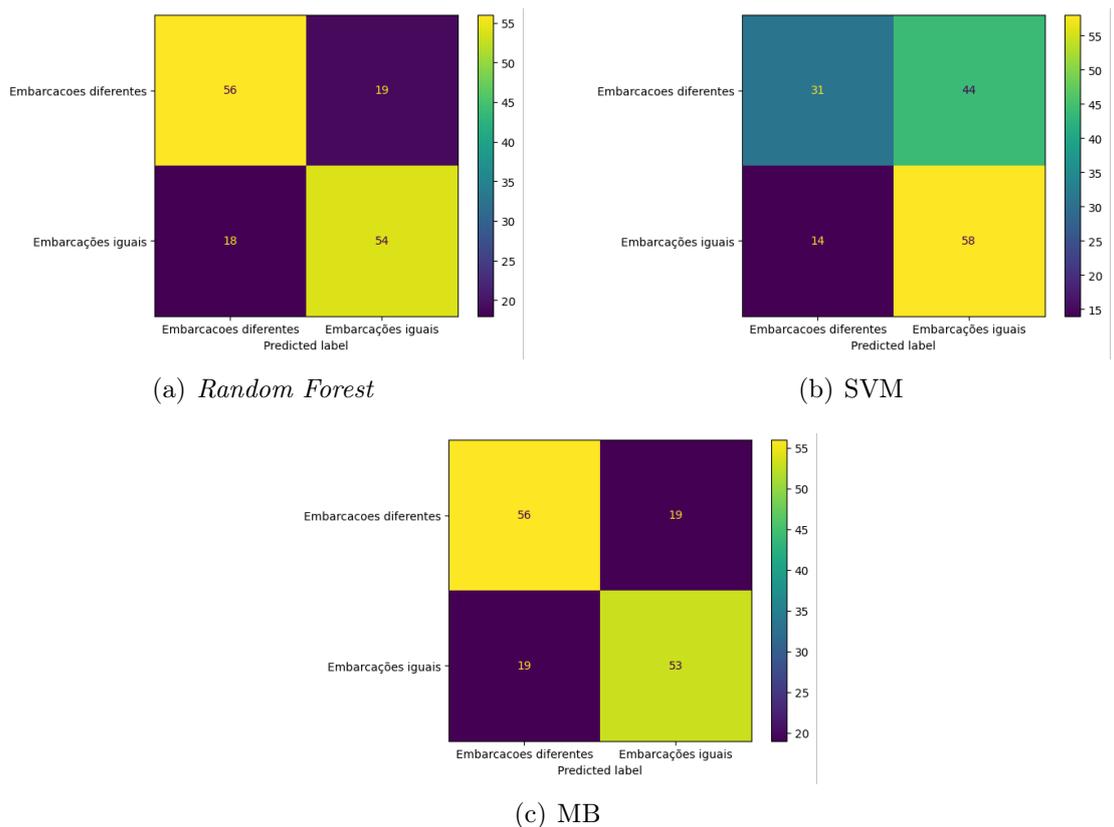


Figura A.5: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 2 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

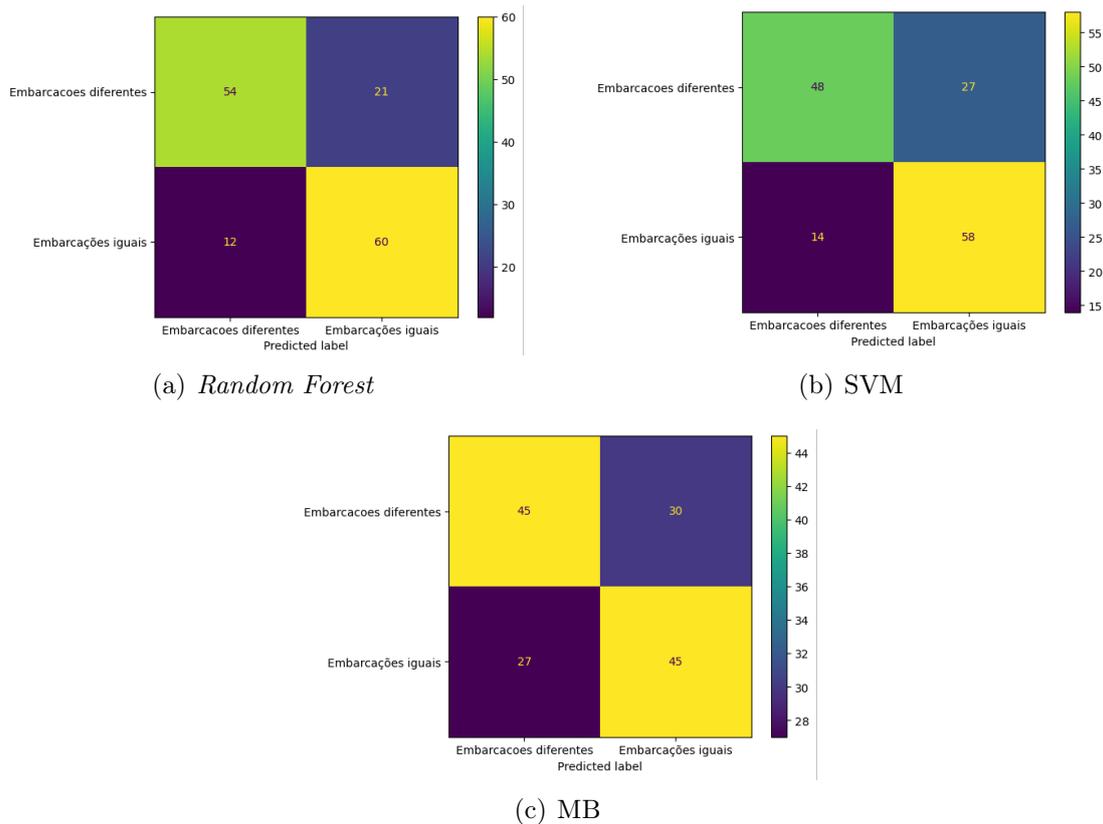


Figura A.6: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 2 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

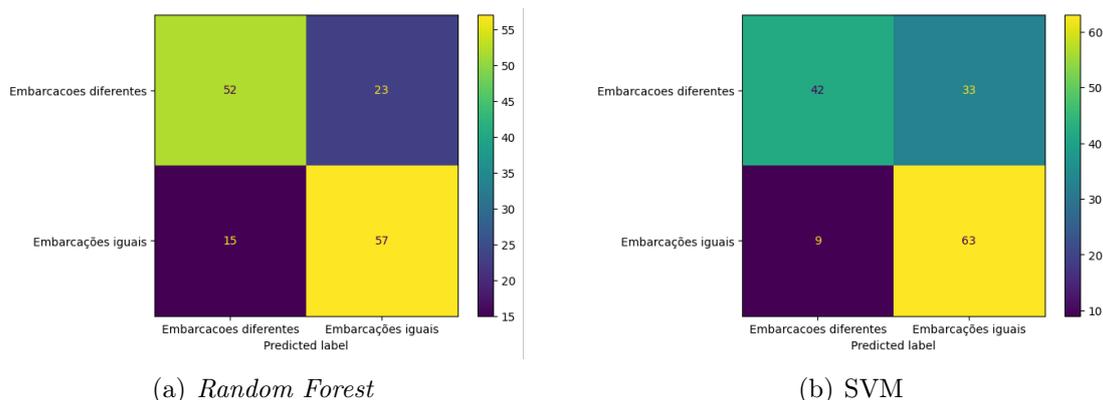
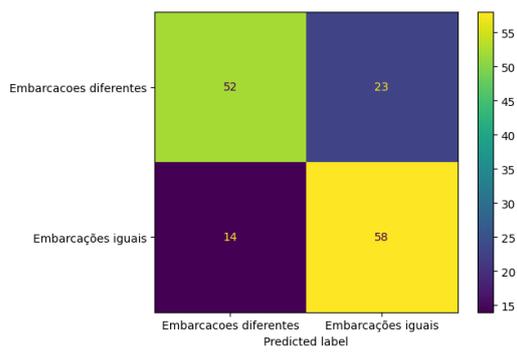
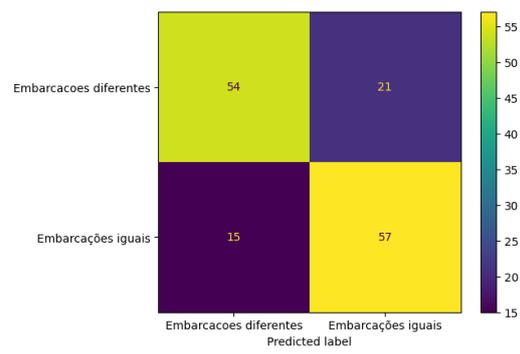


Figura A.7: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 2 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

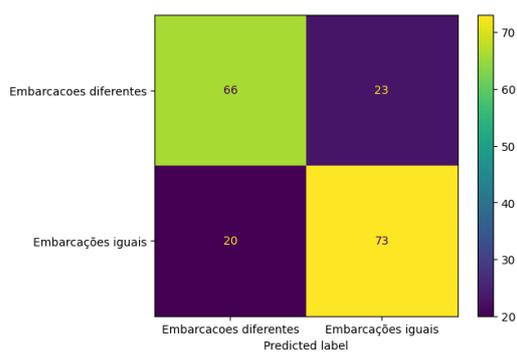


(a) *Random Forest*

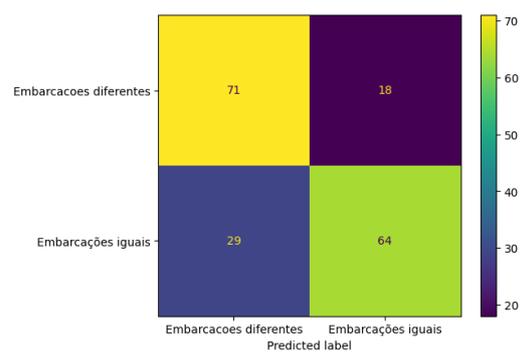


(b) *SVM*

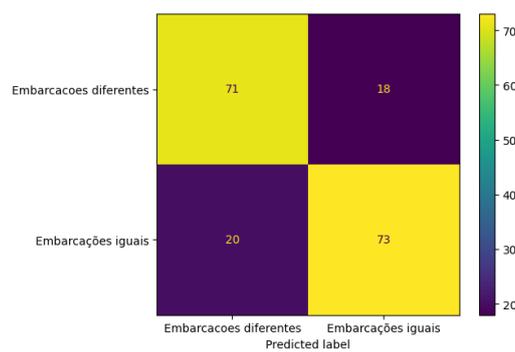
Figura A.8: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 2 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



(a) *Random Forest*



(b) *SVM*



(c) *MB*

Figura A.9: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 3 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

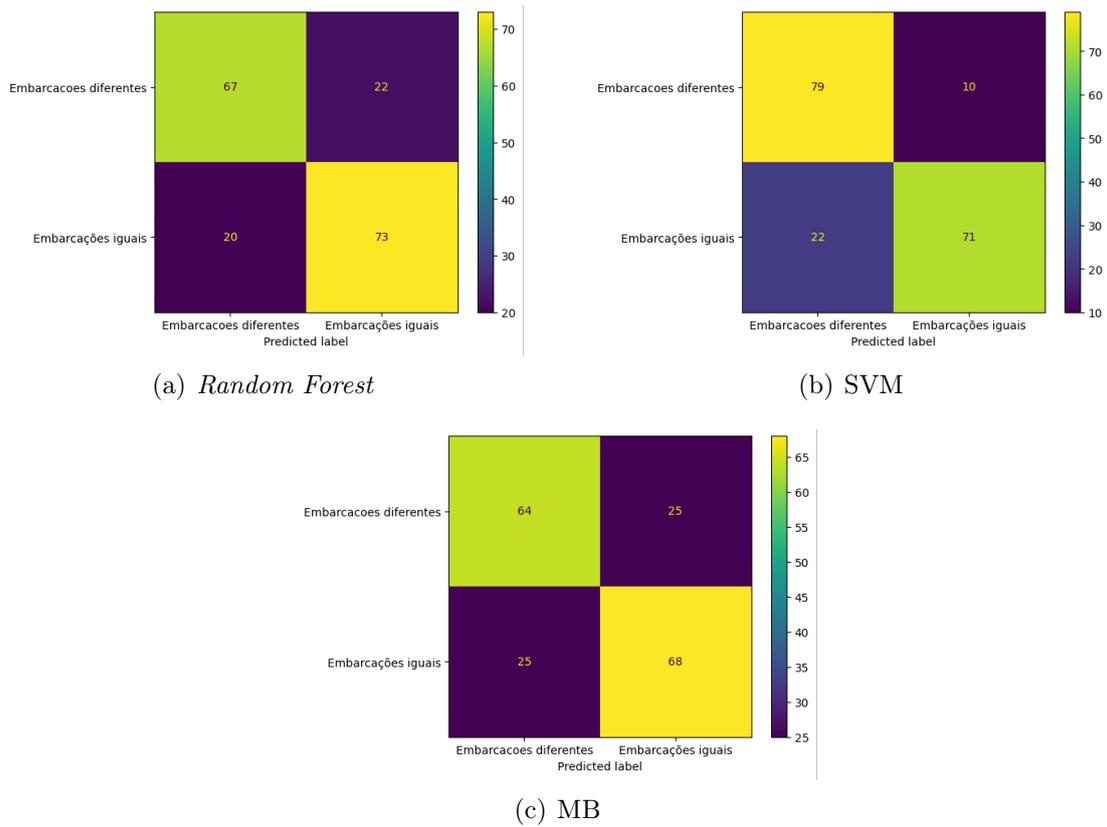


Figura A.10: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 3 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

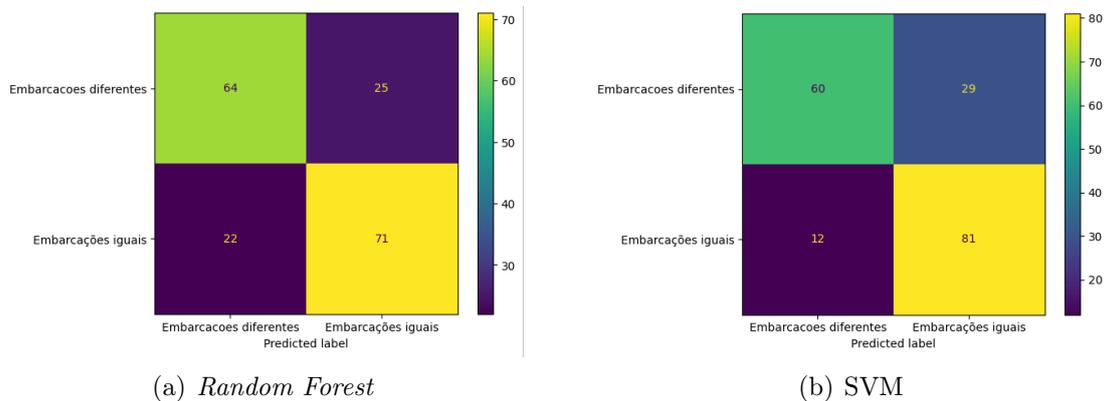


Figura A.11: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 3 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

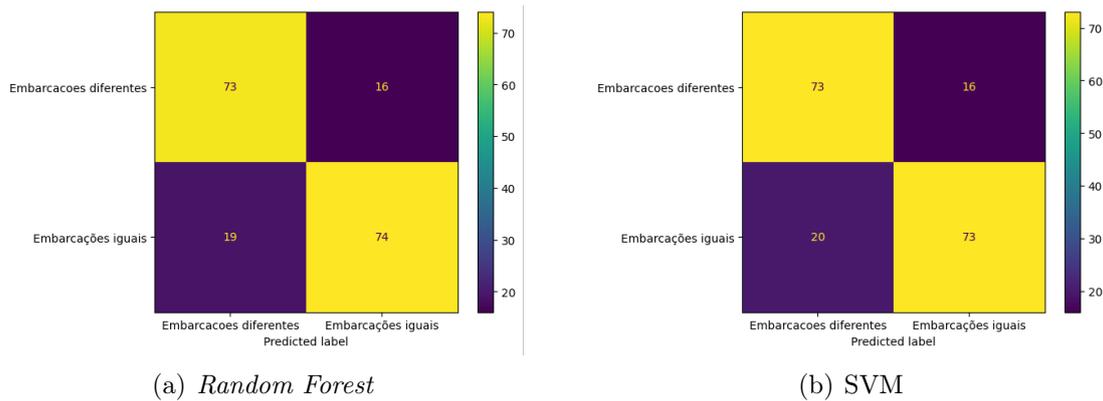


Figura A.12: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 3 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos

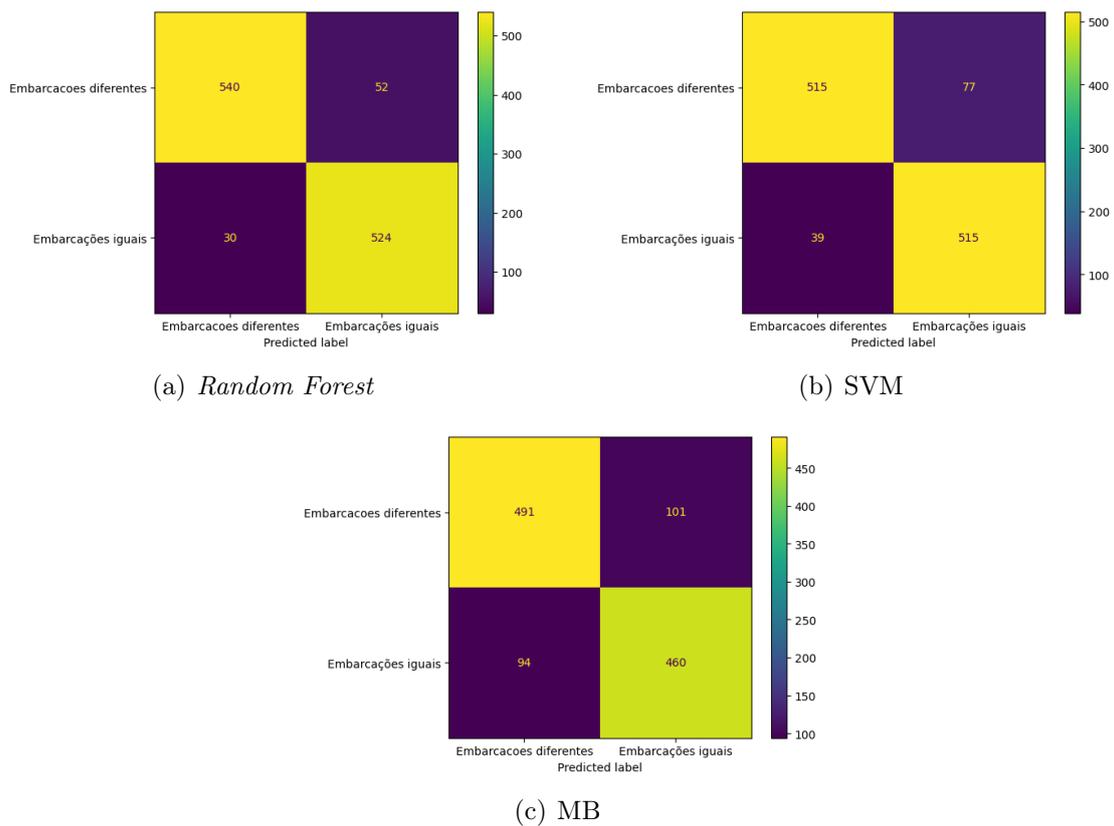
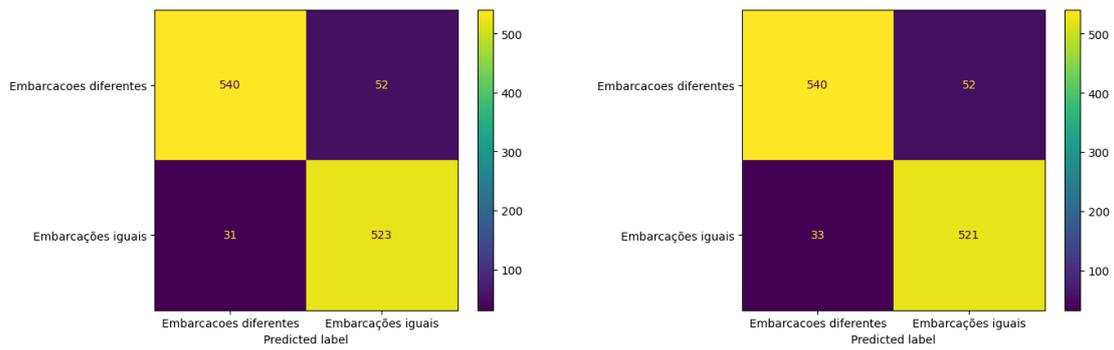
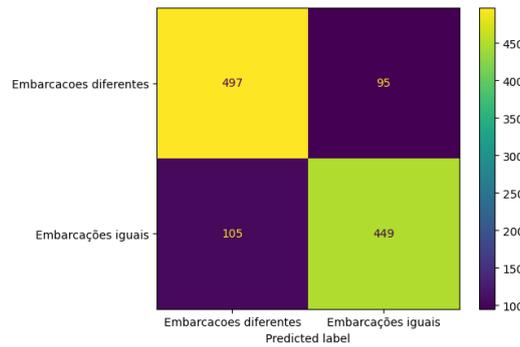


Figura A.13: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 4 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



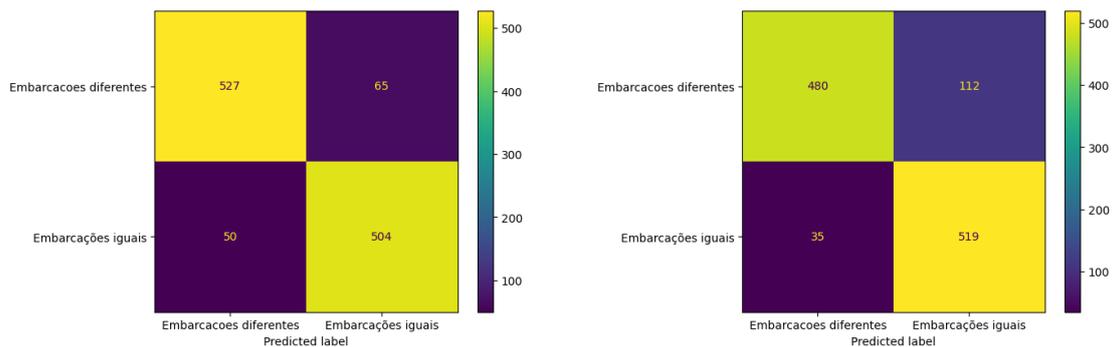
(a) *Random Forest*

(b) *SVM*



(c) *MB*

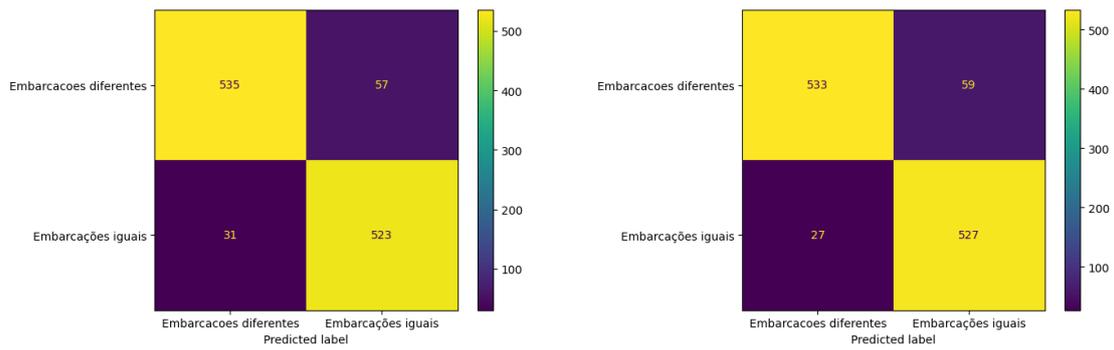
Figura A.14: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 4 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



(a) *Random Forest*

(b) *SVM*

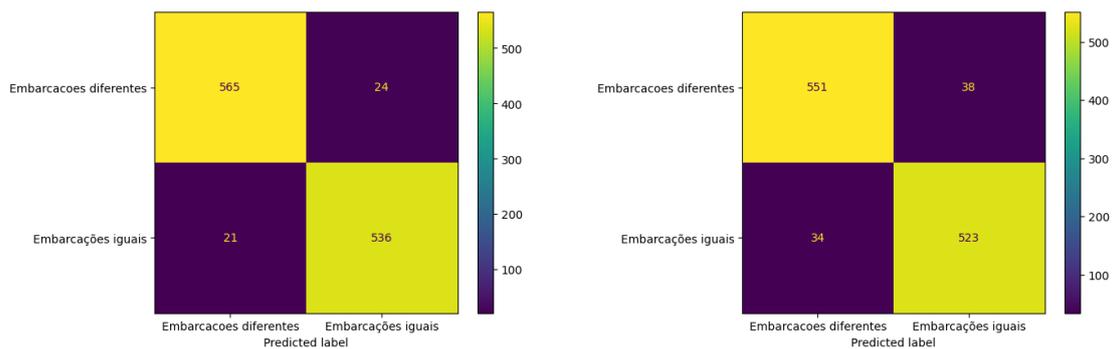
Figura A.15: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 4 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



(a) *Random Forest*

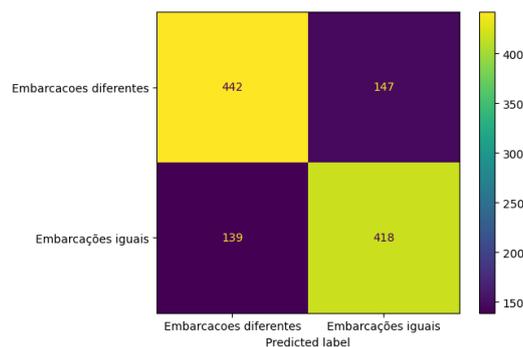
(b) *SVM*

Figura A.16: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 4 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



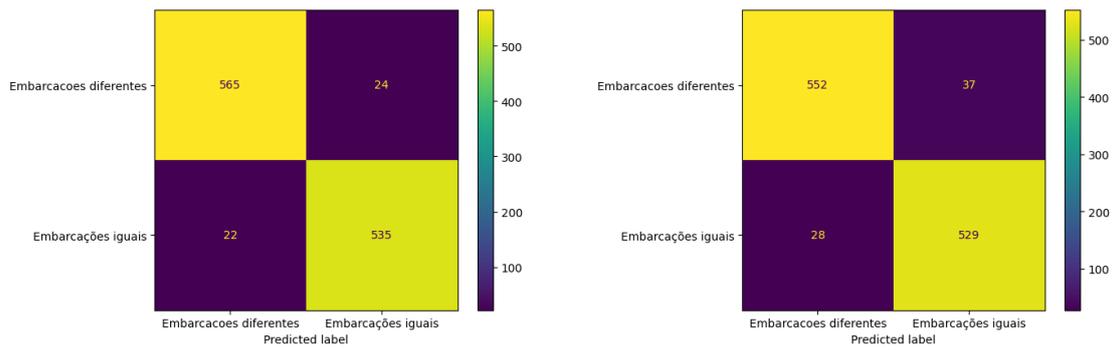
(a) *Random Forest*

(b) *SVM*



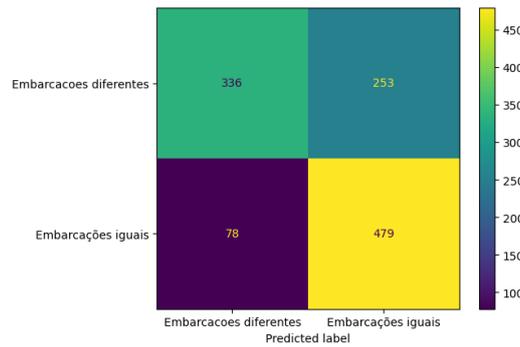
(c) *MB*

Figura A.17: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 5 com dados completos(sem pré-processamento). O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



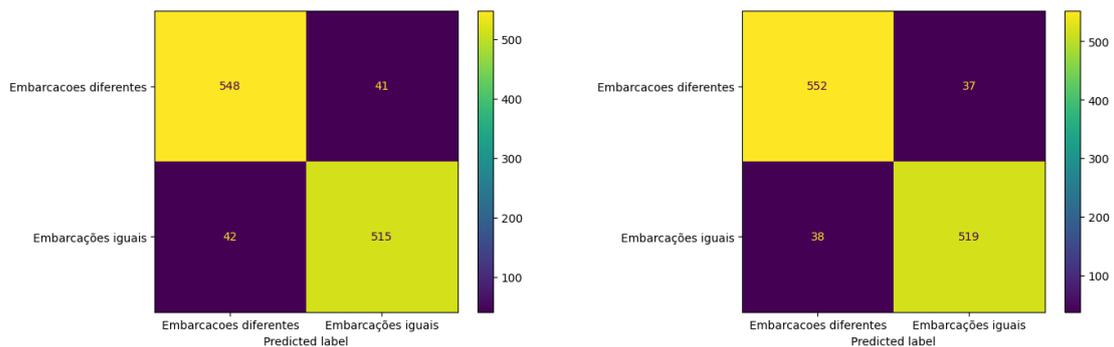
(a) *Random Forest*

(b) *SVM*



(c) *MB*

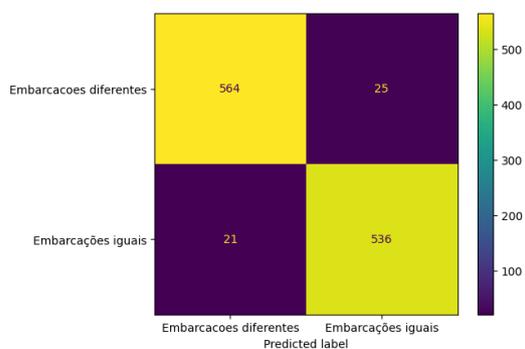
Figura A.18: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 5 com dados pré-processados com normalização logarítmica. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



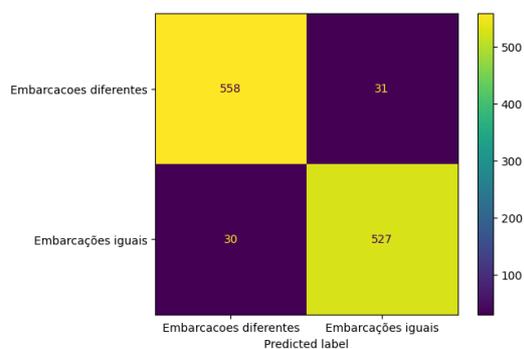
(a) *Random Forest*

(b) *SVM*

Figura A.19: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 5 utilizando os atributos estatísticos extraídos das trajetórias. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos



(a) *Random Forest*



(b) SVM

Figura A.20: Matrizes de confusão com os resultados obtidos pelos classificadores para o conjunto de dados 5 com normalização logarítmica e redução para utilização de 10 atributos utilizando o algoritmo SFS. O eixo y representa as classificações verdadeiras e o eixo x, as classificações feitas pelos algoritmos