Prevenção de Colisões baseada em Aprendizado de Máquina e Comunicação para Redes Veiculares

Andreia A. Felix[1], **Joahannes B. D. da Costa**[2], **Helder M. N. da S. Oliveira**[3]

[1]Universidade Federal do ABC (UFABC), Santo André, Brasil
[2]Universidade Federal de São Paulo (UNIFESP), São José dos Campos, Brasil
[3]Universidade de São Paulo (USP), São Paulo, Brasil

andreia.alexandre@ufabc.edu.br, joahannes.costa@unifesp.br,
helderoliveira@ime.usp.br

Abstract. *Road safety can be enhanced through the integration of V2X communication and Machine Learning techniques. This work proposes a system for predicting and classifying vehicle collision risk based on mobility data obtained from realistic simulations in the urban scenario of Cologne. Traditional models (KNN, Decision Tree, and Random Forest) were evaluated for both regression and classification tasks, considering prediction horizons of up to 5 seconds. The results indicate high predictive performance, with accuracy exceeding 97% in risk classification and high R^2 values for short-term predictions, demonstrating the models' ability to capture relevant patterns in vehicle dynamics. Random Forest showed greater robustness across different horizons, while KNN achieved strong performance in immediate predictions. Additionally, the V2X communication analysis revealed low latency and high message delivery rates, supporting the feasibility of the system in a simulated environment. It is concluded that low-complexity models can achieve high performance in collision anticipation, contributing to the development of efficient and scalable vehicle safety solutions.*

Resumo. *A segurança viária pode ser aprimorada com a integração de comunicação V2X e técnicas de Aprendizado de Máquina. Este trabalho propõe um sistema para previsão e classificação de risco de colisões veiculares com base em dados de mobilidade obtidos de simulações realistas no cenário urbano de Cologne. Foram avaliados modelos tradicionais (KNN, Decision Tree e Random Forest) em tarefas de regressão e classificação, considerando horizontes de previsão de até 5 segundos. Os resultados indicam elevado desempenho preditivo, com acurácia superior a 97% na classificação de risco e altos valores de R^2 nas previsões de curto prazo, demonstrando a capacidade dos modelos em capturar padrões da dinâmica veicular. O Random Forest apresentou maior robustez entre os horizontes analisados, enquanto o KNN se destacou em previsões imediatas. Adicionalmente, a análise da comunicação V2X revelou baixa latência e alta taxa de entrega de mensagens, reforçando a viabilidade do sistema em ambiente simulado. Conclui-se que modelos de menor complexidade podem alcançar desempenho elevado na antecipação de colisões, contribuindo para o desenvolvimento de soluções eficientes e escaláveis para segurança veicular.*

1. Introdução

A segurança viária é uma preocupação global, especialmente diante do aumento do tráfego e da complexidade das dinâmicas viárias [Comi et al. 2024]. Em geral, acidentes

ocorrem devido à falhas humanas, tempos de resposta insuficientes e condições adversas da via, tornando essencial o desenvolvimento de soluções para mitigar tais riscos. O avanço tecnológico tem permitido a aplicação de estratégias mais eficazes, como a comunicação entre veículos e infraestrutura associada a algoritmos de Aprendizado de Máquina, ou *Machine Learning (ML)* em inglês [Souza et al. 2024]. Essas abordagens possibilitam a troca de informações em tempo real, permitindo a antecipação de situações de risco, a redução de colisões e aumento da eficiência no fluxo do tráfego veicular [Kumar et al. 2025].

Entre as inovações mais promissoras está a comunicação veículo-para-tudo, do inglês *Vehicle-to-Everything (V2X)*, que viabiliza a troca de dados entre veículos, infraestruturas de comunicação veicular e outros elementos comunicáveis presentes no trânsito, tais como sensores e semáforos inteligentes [Xiang et al. 2024]. Tal conectividade desempenha um papel fundamental na implementação de sistemas de transporte inteligentes e na evolução da direção autônoma. Por exemplo, veículos conectados e autônomos, do inglês *Connected and Autonomous Vehicles (CAVs)*, utilizam essa tecnologia para aprimorar a segurança e a eficiência viária em seu entorno, reduzindo congestionamentos e minimizando riscos tanto para passageiros quanto para pedestres [Pipicelli et al. 2024, Parvini et al. 2024].

Além disso, a utilização de algoritmos de ML está cada vez mais presente no cenário rodoviário, principalmente no contexto da prevenção de acidentes [Zeng et al. 2024]. Levando em consideração a massiva quantidade de dados que se pode extrair de veículos, tais como sensores, câmeras, dentre outros, comportamentos de riscos podem ser detectados e ativamente mitigados, tanto pelo próprio veículo quanto pelo motorista. Algumas aplicações nesse contexto incluem detecção de mudanças bruscas de faixa, detecção de freadas inesperadas e manutenção de distância segura entre veículos.

Diferentemente de abordagens existentes, este trabalho investiga o uso de modelos de ML para prever colisões com antecedência de até 5 segundos, avaliando não apenas a acurácia dos modelos, mas também seu comportamento em diferentes horizontes temporais e seu impacto em um ambiente de comunicação V2X. Especificamente, busca-se responder: (i) qual o desempenho de modelos tradicionais em diferentes janelas de previsão; (ii) como esses modelos se comportam em cenários urbanos realistas; e (iii) quais são as limitações desses métodos frente à natureza temporal dos dados veiculares. Para isso, foram conduzidas 1000 simulações no cenário de Cologne, permitindo uma análise abrangente dos eventos de colisão e risco.

O restante deste trabalho está organizado da seguinte forma: A Seção 2 apresenta uma revisão dos principais estudos que relacionam ML e comunicação V2X. A Seção 3 descreve a metodologia empregada para investigação dos algoritmos de ML no contexto de prevenção de acidentes veiculares. A Seção 4 analisa e discute o desempenho dos modelos nos cenários propostos. Por fim, a Seção 5 apresenta a conclusão e discute as limitações do estudo, além de direcionar para pesquisas futuras.

2. Trabalhos Relacionados

Esta seção revisa pesquisas recentes sobre previsão e prevenção de colisões veiculares, destacando o uso de ML e comunicação V2X. Diversos estudos exploram modelos preditivos para melhorar a segurança no trânsito, analisando dados de sensores e infraestrutura inteligente com Inteligência Artificial (IA) para antecipar riscos. A combinação de simulações detalhadas e algoritmos de aprendizado supervisionado tem se mostrado eficaz

na previsão de eventos críticos, como frenagens bruscas, mudanças de faixa e proximidade excessiva entre veículos.

Por exemplo, [Alagarsamy et al. 2023] apresentam um estudo onde analisam as principais causas de acidentes em rodovias urbanas por meio da utilização de algoritmos de ML, mais especificamente a Regressão Linear (RL) e *Random Forest (RF)*, para identificar fatores associados a colisões. Essa abordagem busca compreender padrões de acidentes e classificá-los com maior precisão. No entanto, os autores não consideram cenários dinâmicos e não lineares, o que torna a aplicabilidade do estudo limitada.

Na mesma direção, [Veluchamy et al. 2023] apresentaram um sistema para tomada de decisão de frenagem em sistemas avançados de assistência ao motorista, do inglês *Advanced Driver-Assistance System (ADAS)*. O estudo combina Redes Generativas Adversariais (GANs) e Redes Neurais Convolucionais Profundas (DeepCNNs) para processar vídeos capturados por câmeras, extraindo informações relevantes para a tomada de decisão automatizada. No entanto, o treinamento dos modelos acontece de forma centralizada, o que exige uma alta utilização da capacidade de comunicação dos veículos.

Em [Ribeiro et al. 2023], um modelo baseado em comunicação V2X e redes neurais recorrentes, mais especificamente as do tipo *Long Short-Term Memory (LSTM)*, foi desenvolvido para prever colisões envolvendo usuários vulneráveis das vias, como motociclistas. O estudo demonstrou que a comunicação V2X pode aprimorar a detecção de colisões, permitindo previsões mais precisas em situações onde sensores convencionais apresentam limitações, como bloqueios de linha de visão. No entanto, a aplicabilidade do modelo é prejudicada em contextos específicos e no impacto de variáveis não capturadas pelas simulações, como as características do tráfego urbano em cidades complexas como da cidade de Cologne, usada no presente estudo. Ou seja, embora as simulações apresentem bons resultados em cenários controlados, fatores como a interação entre diferentes tipos de veículos, padrões de mobilidade dinâmicos e condições ambientais específicas de uma cidade podem não ser totalmente representados.

Além disso, [Radi et al. 2024] propõem um sistema descentralizado de comunicação Veículo-para-Veículo (V2V) para redes veiculares. O objetivo é melhorar a segurança, eficiência e escalabilidade do tráfego, eliminando a necessidade de um servidor central. A arquitetura do sistema inclui três componentes principais: Unidades de Bordo (OBUs), Unidades de Beira de Estrada (RSUs) e um servidor em nuvem. As OBUs processam dados dos sensores e realizam comunicação direta via DSRC (IEEE 802.11p), enquanto as RSUs coletam informações sobre o tráfego e as retransmitem.

Apesar dos avanços apresentados, observa-se que grande parte dos trabalhos utiliza modelos mais sofisticados, como redes neurais profundas ou abordagens híbridas com visão computacional, explorando explicitamente a natureza temporal dos dados veiculares. Em contraste, este trabalho adota modelos tradicionais com menor custo computacional, o que favorece aplicações em tempo real, porém pode limitar a capacidade de capturar dependências temporais de longo prazo. Diferentemente de [Ribeiro et al. 2023], que utiliza LSTM para modelar sequências temporais, este estudo avalia o impacto de janelas de previsão crescentes em modelos não sequenciais, evidenciando sua degradação de desempenho em horizontes maiores. Assim, este trabalho complementa a literatura ao analisar o trade-off entre desempenho, simplicidade e custo computacional em ambientes V2X.

3. Prevenção de Colisões Veiculares

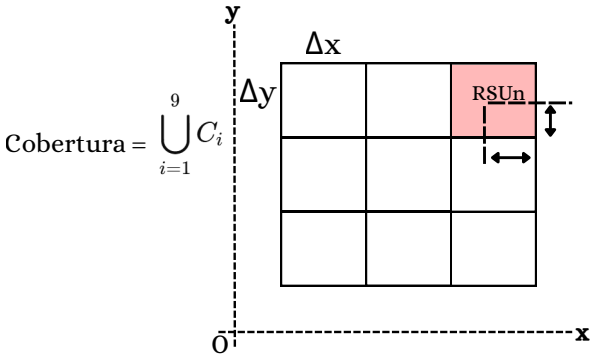
Esta seção descreve a metodologia utilizada para avaliação das técnicas de ML no contexto de prevenção de colisões veiculares. Considerou-se simulações de mobilidade veicular que incorporam um *trace* de mobilidade realística e um cenário de comunicação V2X, com o objetivo de aprofundar a compreensão sobre o comportamento dos veículos em diferentes contextos urbanos.

3.1. Dinâmica Veicular

Para considerar uma dinâmica veicular mais realista, foi utilizado o *trace* de mobilidade do projeto TAPASCologne[1], que reproduz o tráfego de veículos da cidade de Cologne, Alemanha. Porém, apenas um recorte central de 114 km^2 foi considerado, conforme mostra a Figura 1a. Em termos de comunicação veicular, considerou-se um cenário composto por um conjunto N de veículos, onde cada veículo u_i possui uma identificação individual ($i \in [1, x]$) e é equipado com uma Unidade de Bordo, do inglês *On-Board Unit (OBU)*, que possibilita a comunicação V2X. A Figura 1b apresenta a representação matemática da área de cobertura de comunicação da infraestrutura implantada.

Nessa expressão, C_i representa a região de cobertura da i-ésima RSU, enquanto o operador de união $\bigcup$ indica que a área total de cobertura é obtida pela combinação das regiões individuais de cobertura de cada unidade de infraestrutura. Como nove RSUs foram implantadas, o índice i varia de 1 a 9. O espaço de simulação é dividido em uma grade bidimensional definida pelas dimensões espaciais Δx e Δy. Cada célula da grade representa uma subárea do cenário, e as RSUs são posicionadas estrategicamente dentro dessa estrutura. Cada unidade possui um raio de comunicação r, representado na figura por setas horizontais e verticais ao redor da RSU destacada.. Os veículos enviam mensagens de dados a cada um segundo na rede (frequência de 1 Hz), de modo que a Roadside Unit (RSU) coleta essas informações em tempo real e constrói o conhecimento necessário para sua tomada de decisão futura conforme exemplificado na Figura 1b.

(a) Cenário de Cologne no *framework* Veins.



(b) Dinâmica da troca de mensagens.

Figure 1. Cenário considerado nos experimentos.

Cada RSU é capaz de gerar um registro das mensagens trocadas com os veículos. Esse registro contém informações cruciais para a análise do desempenho, como a

[1] `http://kolntrace.project.citi-lab.fr/`

identificação do veículo emissor, a identificação única da mensagem recebida, o *timestamp* de quando a mensagem foi recebida, a identificação da resposta gerada pela RSU e o *timestamp* de envio dessa resposta. Dada a possibilidade de comunicação V2X, conforme ilustrado na Figura 1b, é possível ter dados fundamentais para avaliar a latência, a eficiência da troca de informações e a qualidade da comunicação entre os veículos e as RSUs ao longo do tempo.

Em relação aos diferentes contextos de mobilidade no trânsito, foram configuradas três situações distintas que ilustram diferentes níveis de risco de colisão entre os veículos. A primeira situação, considerada a mais segura, onde os veículos mantêm uma distância de segurança adequada entre si, permitindo que possa haver reação em tempo hábil para evitar colisão. Na segunda situação, os veículos se encontram em uma situação de risco de colisão. Nesse caso, a distância entre eles é inferior a 2 segundos. A simulação da colisão permitiu avaliar e mapear o comportamento veicular que antecede o evento de colisão.

Por fim, o Algoritmo 1 descreve o funcionamento da RSU na troca de mensagens com veículos em sua área de cobertura. Ela recebe mensagens dos veículos e verifica se algum deles está em situação de risco ou colisão. Caso isso ocorra, a RSU transmite uma mensagem de alerta para todos os veículos próximos (*broadcast*).

Algoritmo 1: Comunicação e Resposta da RSU

Entrada: Mensagens M_i recebidas dos veículos $i \in N$
Saída: Mensagens de alerta transmitidas aos veículos na área de cobertura

Função `Resposta()`:
  Inicializar lista de mensagens recebidas $L_M \leftarrow \emptyset$
  Inicializar lista de alertas enviados $L_A \leftarrow \emptyset$
  `// Etapa 1: Recepção das mensagens`
  para cada *mensagem* M_i *recebida do veículo* $i \in N$ **faça**
    Armazenar M_i em L_M
  `// Etapa 2: Processamento das mensagens`
  para cada *mensagem* $M_i \in L_M$ **faça**
    Extrair id_i, pos_i, vel_i, $situation_i$
    `// Verificação de criticidade`
    se $situation_i \in \{RISCO, COLISAO\}$ **então**
      se $id_i \notin L_A$ **então**
        Criar mensagem de alerta A_i
        Definir conteúdo de A_i: (id_i, pos_i, tipo de evento)
        `// Tamanho fixo da mensagem`
        Definir tamanho de $A_i \leftarrow 43$ bytes
        Adicionar id_i em L_A
        `// Etapa 3: Disseminação`
        Transmitir A_i em *broadcast* para todos os veículos na área de cobertura
  `// Etapa 4: Atualização contínua`
  Limpar L_M após processamento

3.2. Algoritmos de *Machine Learning (ML)*

Após a coleta dos dados, realizada por meio da comunicação em tempo real entre veículos e RSUs, obtém-se um conjunto de dados $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, no qual $\mathbf{x}_i \in \mathbb{R}^d$ representa o vetor de características associado às informações de mobilidade (como posição, velocidade e direção), enquanto y_i corresponde à variável alvo, podendo assumir valores contínuos (regressão) ou discretos (classificação). Com base nesse conjunto, foram empregados algoritmos clássicos de *Machine Learning*, tanto para regressão quanto para

classificação, a saber: *K-Nearest Neighbors* (KNN), *Random Forest* (RF) e *Decision Tree* (DT). As versões de regressão são denotadas por KNN-r, RF-r e DT-r, enquanto as versões de classificação são indicadas por KNN-c, RF-c e DT-c.

No algoritmo KNN, a estimativa é obtida a partir dos k elementos mais próximos no espaço $\mathbb{R}^d$, segundo uma métrica de distância. Neste estudo, foi fixado $k = 5$ ($n\_neighbors = 5$) para ambas as abordagens (regressão e classificação), mantendo consistência na definição da vizinhança local. Para o modelo *Decision Tree*, foi adotado $random\_state = 42$ em ambas as versões, garantindo reprodutibilidade no particionamento recursivo do espaço de atributos. No caso do *Random Forest*, que consiste na agregação de múltiplas árvores de decisão construídas a partir de subconjuntos amostrados dos dados, também foi utilizado $random\_state = 42$.

Adicionalmente, na versão de classificação, foi estabelecido um limite superior para a profundidade das árvores, com $max\_depth = 100$, de modo a restringir a complexidade do modelo. A definição desses hiperparâmetros implica uma configuração controlada do espaço de busca dos modelos, favorecendo estabilidade e capacidade de generalização, conforme evidenciado em [Gnoatto and Franzen 2023].

Os dados utilizados para treinar e testar os modelos foram extraídos de simulações de mobilidade veicular através do simulador SUMO (*Simulation of Urban MObility*), com a execução de 1000 simulações com sementes aleatórias. As variáveis de entrada selecionadas que representam características essenciais para a previsão da dinâmica dos veículos são: Posição X, Posição Y, Velocidade, Aceleração, Direção e Distância. Essas informações permitiram capturar o comportamento do tráfego em diferentes condições e fornecer uma base robusta para a previsão de colisões. Ainda, para garantir a qualidade dos dados, foi realizado um pré-processamento que incluiu a normalização das variáveis contínuas, eliminando discrepâncias de escala entre os atributos. Além disso, registros inconsistentes ou incompletos foram removidos para evitar viés nos modelos.

O Algoritmo 2 descreve o sistema de predição e comunicação de risco que é executado em cada veículo no cenário. Inicialmente, ele recebe dados de mobilidade dos veículos. Com esses dados, o sistema treina modelos para prever o risco de colisões. Ele funciona em duas etapas principais, a primeira de predição e a segunda de comunicação, denominadas `Predicao` e `Comunicacao`, respectivamente. Primeiro, continuamente os veículos realizam o processo de predição localmente (Linha 3). Na etapa de predição, modelos regressivos são utilizados para realizar a predição do comportamento do veículos em diferentes janelas de observação (w) (Linhas 6 a 9). Após essa predição, modelos de classificação são aplicados para identificar as situações (seguro, risco ou colisão) que o veículo se encontrará em cada w futuro (Linhas 10 a 13). A situação do veículo se dá por meio da combinação desses resultados, podendo ser *seguro*, *risco* ou *colisão* (Linha 14). Por fim, o veículo envia uma mensagem de dados para a RSU mais próxima (Linha 4), podendo ser uma mensagem de alerta em caso de situação de *risco* ou *colisão* ou uma mensagem de atualização se sua situação for de estado *seguro* (Linhas 16 a 20).

O pipeline proposto combina modelos de regressão e classificação de forma sequencial. Inicialmente, os modelos regressivos são utilizados para prever variáveis futuras do estado do veículo em diferentes horizontes temporais. Em seguida, essas previsões são utilizadas como entrada para os modelos de classificação, que determinam a situação do veículo (seguro, risco ou colisão). A definição das classes baseia-se na regra dos dois segundos, onde distâncias inferiores ao deslocamento equivalente a esse intervalo são classificadas como risco, e valores críticos associados à proximidade extrema indicam colisão iminente. A decisão final do sistema é obtida pela combinação das previsões ao

Algoritmo 2: Predição de Risco no Veículo

Entrada: Dados de entrada pré-treinados
Saída: Situação $situacao$ do veículo $i \in N$

$w \leftarrow [1, 2, 3, 4, 5]$; `// Janelas de predição`
para cada *veículo* $i \in N$ **faça**
    `Predição();`
    `Comunicação();`
Função `Predição()`**:**
    `/* Modelos de Regressão */`
    para cada *modelo* $r \in R$ *que prediz o comportamento do veículo* **faça**
        Executa os modelos com os dados de entrada em cada w;
        Calcula métricas de desempenho;
    Seleciona $r \in R$ com as melhores métricas;
    `/* Modelos de Classificação */`
    para cada *modelo* $c \in C$ *que classifica a situação do veículo* **faça**
        Executa os modelos com os dados de entrada em cada w;
        Calcula métricas de desempenho;
    Seleciona $c \in C$ com as melhores métricas;
    $situacao \leftarrow$ Combina resultados de R e C para cada w;
    retorna $situacao$ em cada w;
Função `Comunicação()`**:**
    se $situacao$ == *RISCO* **ou** $situacao$ == *COLISAO* **então**
        Envia mensagem de *alerta* para RSU; `// 83 bytes`
    senão
        Envia mensagem de *atualização* para RSU; `// 83 bytes`

longo das diferentes janelas w, priorizando estados mais críticos.

3.3. Métricas de Avaliação

Levando em consideração que modelos de regressão e de classificação estão presentes na avaliação, as seguintes métricas foram consideradas. Em relação aos modelos de regressão, considerou-se as métricas: *(1) Mean Absolute Percentage Error (MAPE)*; *(2) Root Mean-Square Error (RMSE)*; e *(3) Coefficient of Determination (R2)*, denotado por R^2. Além disso, foi considerado o tempo de predição para cada um dos modelos.

Inicialmente, o R^2 é usado como principal métrica para avaliar o melhor modelo regressivo. Enquanto as métricas de erro possuem valores que podem variar de zero ao infinito, dificultando a interpretação isolada dos resultados, o R^2 fornece uma escala padronizada entre $-\infty$ e 1, permitindo uma análise mais clara da qualidade do modelo. Além disso, essa métrica expressa a proporção da variabilidade dos dados explicada pelo modelo, sendo sensível ao desempenho global da regressão. Ou seja, o R^2 elevado mostra que a maioria dos valores reais são corretamente previstos, evitando distorções causadas por variações de escala ou pela presença de *outliers* [Chicco et al. 2021].

Posteriormente, usa-se o MAPE que é uma métrica amplamente utilizada para avaliar a precisão de modelos de regressão, expressando o erro médio percentual entre os valores previstos e os valores reais. Finalmente, é usado o RMSE, que mede a magnitude média dos erros de previsão, atribuindo maior peso a erros elevados devido ao uso do quadrado das diferenças entre valores previstos e reais. Essa característica torna o RMSE particularmente útil em cenários onde grandes erros precisam ser fortemente

penalizados [Chicco et al. 2021].

Já para os modelos de classificação, foram consideradas as seguintes métricas: *(1) Acurácia*; *(2) Precisão*; *(3) Sensibilidade*; *(4) F1-Score*; *(5) Especificidade*; e *(6) Curva ROC*. A Acurácia indica a proporção de previsões corretas, sendo importante para avaliar o desempenho geral do modelo. A Precisão avalia a relevância das previsões positivas, enquanto a Sensibilidade mede a capacidade do modelo em identificar corretamente os casos positivos. O *F1-Score*, sendo a média harmônica entre precisão e sensibilidade, busca equilibrar essas duas métricas. A Especificidade mede a capacidade de identificar corretamente os negativos. A Curva ROC é crucial para avaliar o desempenho, analisando a relação entre verdadeiros positivos e falsos positivos em diferentes limiares, ajudando a definir o melhor ponto de equilíbrio entre sensibilidade e especificidade. Os modelos K-Nearest Neighbors (KNN), Decision Tree (DT) e RF foram escolhidos por sua facilidade de treinamento e menores requisitos computacionais, o que facilita a análise em diversos cenários [Alnuaimi and Albaldawi 2024].

4. Avaliação de Desempenho

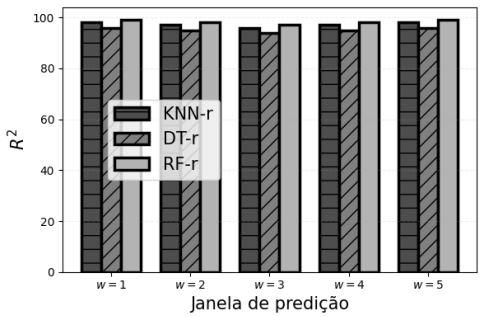
Esta seção apresenta detalhes da avaliação de desempenho realizada com os modelos de ML no contexto de detecção e prevenção de acidentes veiculares. A análise avalia a previsão de comportamentos veiculares, a identificação de padrões de tráfego e a eficiência da comunicação. Os resultados estão segmentados entre os modelos de regressão (Seção 4.1), os modelos de classificação (Seção 4.2) e resultados sobre a comunicação veicular (Seção 4.3).

4.1. Resultados: Modelos de Regressão

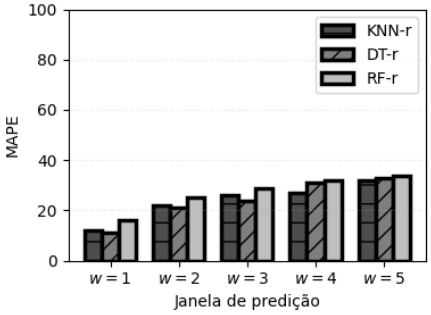
A Figura 2 compara os modelos KNN-r, DT- e RF-r em termos de R^2, MAPE e RMSE para diferentes horizontes de previsão $w = \{1, 2, 3, 4, 5\}$. Observa-se que o KNN-r é o modelo mais eficiente para previsões de curto prazo ($w = 1, 2, 3$), pois apresenta os maiores valores de R^2, indicando um melhor ajuste aos dados, além dos menores valores de MAPE e RMSE, evidenciando uma menor margem de erro nas previsões. No entanto, à medida que o horizonte de previsão aumenta ($w \geq 4$), o desempenho do KNN-r se deteriora, resultando em uma redução significativa no R^2 e um aumento nos erros de MAPE e RMSE. Isso mostra que o KNN-r é mais eficiente na captura de padrões locais e na extrapolação de tendências de curto prazo, mas mostra dificuldade em manter a precisão quando a previsão se estende para janelas maiores.

Adicionalmente, o RF-r se destaca para previsões de médio e longo prazo, apresentando menor erro e melhor ajuste a partir de $w = 4$. Isso se deve ao aprendizado do RF-r, que combina múltiplas árvores de decisão para reduzir a variabilidade e melhorar a generalização dos dados. Assim, o RF-r se mostra mais robusto para previsões de longo prazo, ao contrário do KNN-r, que depende fortemente da proximidade dos pontos de treino e sofre queda no desempenho com o aumento do horizonte de previsão.

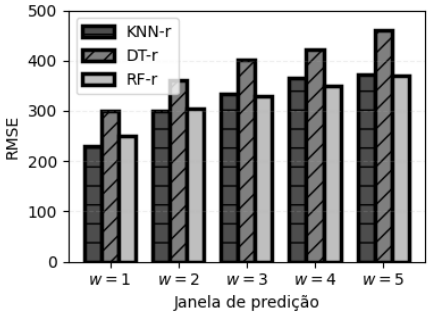
Por fim, o DT-r apresenta bom desempenho em todas as métricas analisadas. Ele mantém o comportamento estável ao longo dos diferentes horizontes de previsão, porém não se destaca como o melhor modelo. Seu desempenho fica abaixo do KNN-r para previsões de curto prazo e inferior ao RF-r para previsões mais longas, mostrando a limitação que a sua técnica de divisão dos dados por meio de uma árvore de decisão tem em comparação as outras técnicas abordadas.



(a) R² geral dos modelos



(b) MAPE geral dos modelos



(c) RMSE geral dos modelos

Figure 2. Desempenho dos modelos por janela de tempo no cenário de Cologne

4.2. Resultados: Modelos de Classificação

Nesta seção, serão apresentados os resultados obtidos pelos modelos de classificação nos diferentes cenários analisados. De modo geral, foi possível observar comportamento estável nos modelos, mesmo diante da complexidade dos dados, caracterizados por variabilidade na movimentação dos veículos e pela presença de múltiplos fatores que influenciam as trajetórias.

A Tabela 1 apresenta os resultados dos modelos de classificação, considerando diferentes valores de w. Os resultados mostram que, de forma geral, o modelo RF-c obteve os melhores desempenhos em todas as métricas, com acurácia variando entre 98,60% e 97,31%, seguido pelo KNN-c, que apresentou acurácia entre 98,45% e 97,05%. O modelo DT-c teve os menores valores de acurácia, variando de 97,89% a 96,20%. A tendência observada nos três modelos é uma leve redução do desempenho conforme w aumenta. Esse comportamento indica que, à medida que a janela de previsão se torna maior, a capacidade do modelo de manter previsões precisas pode diminuir.

Além disso, a Especificidade permaneceu alta para todos os modelos, com valores superiores a 99,95% na maioria dos casos, demonstrando que os modelos têm uma capacidade de identificar corretamente as instâncias negativas. Já a Sensibilidade, apresentou pouca queda conforme w aumentou, indicando que o modelo pode ter dificuldade em prever corretamente as classes positivas em janelas maiores. E por fim, o *F1-Score*, também apresentou redução à medida que w aumentou, mostrando uma redução da capacidade dos modelos de manter um boa classificação.

Como ilustrado na Figura 3, a Curva ROC dos modelos apresentou pequenas

Table 1. Resultados dos modelos de classificação.

Modelo	Métrica	$w = 1$	$w = 2$	$w = 3$	$w = 4$	$w = 5$
KNN-c	Acurácia	98.45%	97.93%	97.56%	97.27%	97.05%
	Precisão	98.41%	97.84%	97.43%	97.09%	96.83%
	Sensibilidade	98.45%	97.93%	97.56%	97.27%	97.05%
	F1-Score	98.43%	97.87%	97.48%	97.16%	96.91%
	Especificidade	99.97%	99.97%	99.96%	99.95%	99.95%
DT-c	Acurácia	97.89%	97.25%	96.83%	96.46%	96.20%
	Precisão	97.90%	97.26%	96.84%	96.48%	96.23%
	Sensibilidade	97.89%	97.25%	96.83%	96.46%	96.20%
	F1-Score	97.90%	97.25%	96.84%	96.47%	96.21%
	Especificidade	99.97%	99.97%	99.96%	99.96%	99.96%
RF-c	Acurácia	98.60%	98.11%	97.77%	97.50%	97.31%
	Precisão	98.57%	98.02%	97.64%	97.33%	97.10%
	Sensibilidade	98.60%	98.11%	97.77%	97.50%	97.31%
	F1-Score	98.58%	98.04%	97.68%	97.38%	97.16%
	Especificidade	99.97%	99.97%	99.96%	99.96%	99.95%

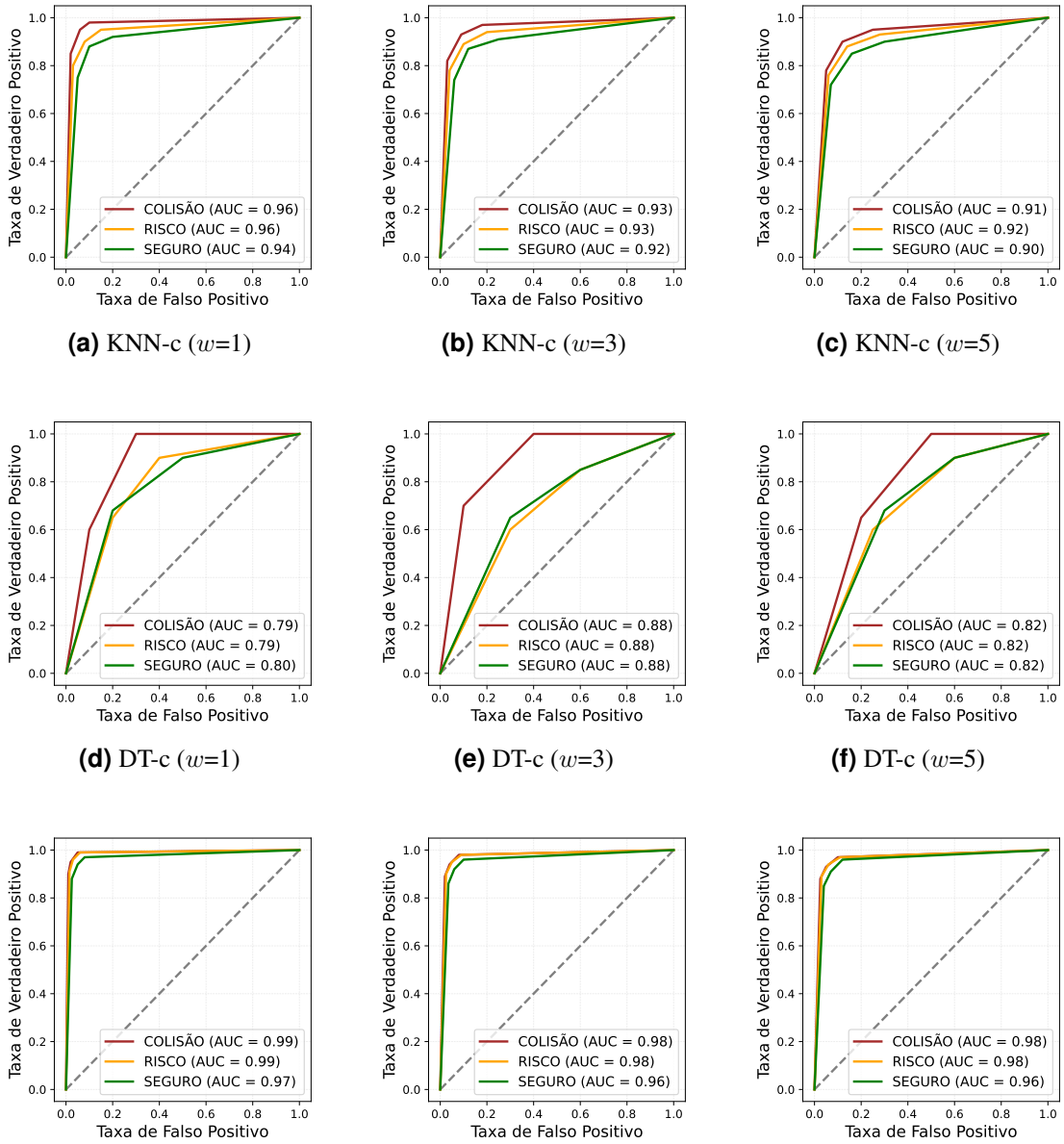
variações, mas manteve-se consistente, indicando que, apesar da queda, a capacidade discriminativa dos modelos permaneceu elevada. Foi apresentada apenas a análise para $w = 1, 3$ e 5 devido à baixa variabilidade nos resultados para outras janelas de tempo. A tendência observada sugere uma leve queda no desempenho dos modelos à medida que w aumenta, o que pode estar relacionado à maior dispersão das informações ao longo do tempo. No entanto, essa variação é pequena, indicando que os modelos são relativamente estáveis para diferentes valores de w.

A degradação observada no desempenho para janelas maiores está diretamente relacionada à incapacidade dos modelos tradicionais em capturar dependências temporais de longo prazo. Modelos como KNN dependem fortemente da proximidade local dos dados, enquanto árvores de decisão possuem limitações na modelagem de dinâmicas sequenciais complexas. Isso explica a maior variabilidade observada no RF para w = 5. Além disso, observou-se que variáveis como velocidade e distância têm maior impacto na previsão de risco, enquanto posição absoluta apresentou menor relevância isolada, indicando que relações relativas são mais importantes para detecção de colisões.

4.3. Resultados: Comunicação Veicular

A Figura 4 apresenta o tempo de predição dos modelos nos cenários de regressão e classificação, considerando diferentes valores de $w = 1, 2, 3, 4, 5$. A análise desses tempos permitiu avaliar o custo computacional associado a cada abordagem, destacando as diferenças de desempenho entre os modelos testados.

Observa-se que os modelos de KNN-r e KNN-c mostram comportamentos distintos. Conforme ilustrado na Figura 4a, há um aumento significativo no tempo de predição para o modelo de classificação à medida que o valor de w cresce, sugerindo um maior custo computacional para encontrar os vizinhos mais próximos em um espaço de alta dimensionalidade. Em contrapartida, o modelo regressivo se mantém mais eficiente, com um tempo de predição estável. Já, os modelos de DT-r e DT-c, conforme evidenciado na Figura 4b, apresenta variações menores entre os dois cenários. O modelo regressivo mantém um tempo de predição mais estável, enquanto o modelo de classificação exibe



(a) KNN-c (w=1) **(b)** KNN-c (w=3) **(c)** KNN-c (w=5)

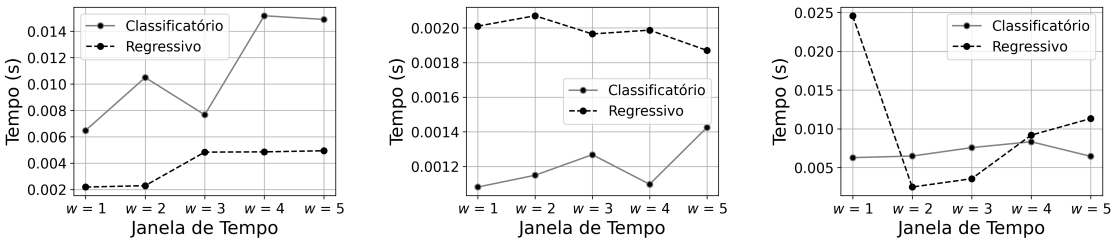
(d) DT-c (w=1) **(e)** DT-c (w=3) **(f)** DT-c (w=5)

(g) RF-c (w=1) **(h)** RF-c (w=3) **(i)** RF-c (w=5)

Figure 3. Curvas ROC dos modelos KNN, Árvore de Decisão e Random Forest para diferentes janelas de predição (w).

flutuações sutis, mas sem um impacto expressivo na escalabilidade. Isso mostra que a estrutura hierárquica da DT consegue lidar de forma equilibrada com ambas as abordagens, sem grandes penalizações no tempo de inferência.

Em contraste, o RF conforme mostrado na Figura 4c, o tempo de predição inicial para o modelo RF-r é relativamente alto, mas estabiliza conforme w aumenta. Já o modelo RF-c apresenta um desempenho mais uniforme ao longo dos diferentes valores de w, mostrando que a estrutura do RF lida de maneira eficiente com a classificação, mantendo um tempo de predição previsível. Os resultados indicam que os modelos regressivos tendem a ser mais rápidos em termos de tempo de predição, enquanto os modelos de classificação apresentam um custo computacional maior.



(a) Tempo de predição K-nearest neighbors

(b) Árvore de decisão

(c) Random Forest

Figure 4. Tempo de predição por modelo

Adicionalmente, foi realizada uma análise detalhada da troca de mensagens no cenário veicular, considerando a comunicação entre os veículos e as RSUs. A comunicação V2X de veículo para infraestrutura não apresentou perdas de pacotes. Foram transmitidas 10.660 mensagens pelos veículos, todas corretamente recebidas por pelo menos uma RSU, conforme ilustrado na Figura 5. Por outro lado, na comunicação de infraestrutura para veículo, foi registrada uma perda de 418 mensagens, ou seja, esses pacotes não foram recebidos por nenhum veículo. Essa perda pode ser atribuída principalmente a dois fatores: colisão de pacotes e interferências no canal de comunicação.

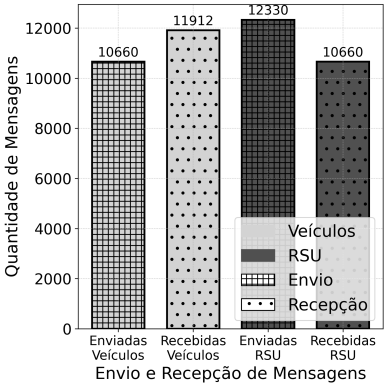
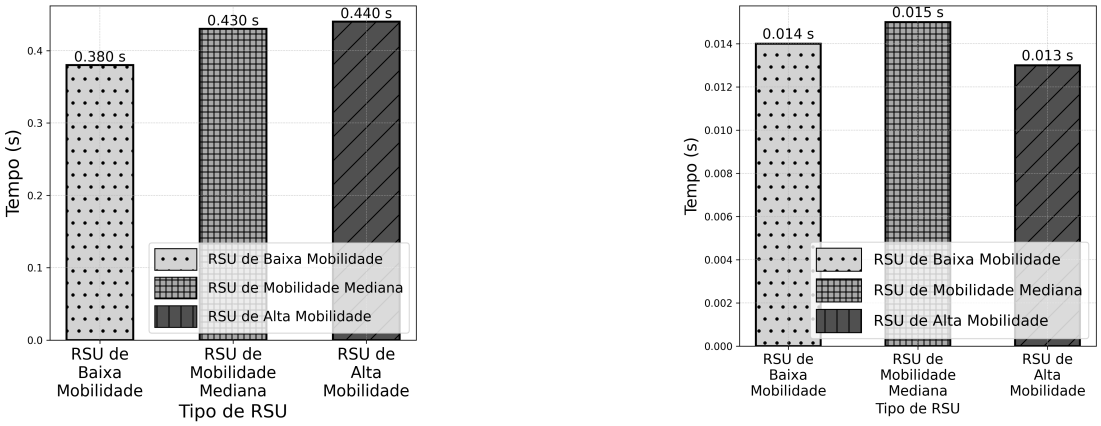


Figure 5. Mensagens enviadas e recebidas na simulação

Além da análise da taxa de entrega de mensagens, também foi avaliado o tempo necessário para a comunicação entre os veículos e as RSUs dentro do ambiente simulado, conforme ilustrado na Figura 6. A latência na comunicação desempenha um papel crítico na eficiência dos sistemas V2X, uma vez que tempos de resposta elevados podem comprometer a disseminação de informações sensíveis ao tempo, como alertas de segurança e atualizações sobre o tráfego. Na Figura 6a, observa-se que as RSUs que atendem a um número maior de veículos apresentam tempos de resposta mais elevados. Esse aumento na latência está diretamente relacionado à sobrecarga no processamento de pacotes por essas unidades, que precisam gerenciar um maior volume de transmissões simultâneas dentro da sua área de cobertura. Por outro lado, a Figura 6b mostra o tempo de comunicação das RSUs para os veículos. Os resultados indicam uma latência menor e constante, e esse comportamento é explicado pelo menor volume de mensagens que os veículos precisam processar em comparação com as RSUs.



(a) Comunicação Veículo para RSU

(b) Comunicação RSU para Veículo

Figure 6. Resultados de tempo de comunicação

4.4. Resultados: Análise Geral

Os melhores modelos estão apresentados na Tabela 2, onde são comparadas suas respectivas métricas de desempenho para diferentes janelas de previsão. A análise dos resultados evidencia que o modelo KNN-r apresenta a melhor eficiência para previsões de curto prazo ($w \leq 4$). Esse comportamento pode ser explicado pelo fato de que o KNN-r se baseia na similaridade com exemplos previamente observados, o que lhe permite capturar padrões locais com alta precisão. Para pequenos horizontes de previsão, essa abordagem é vantajosa, pois reduz a influência de variações aleatórias e mantém a acurácia elevada.

Table 2. Melhores modelos

	$w = 1$	$w = 2$	$w = 3$	$w = 4$	$w = 5$
Modelo	KNN-r RF-c	KNN-r RF-c	KNN-r RF-c	RF-r RF-c	RF-r RF-c
Tempo de predição	0,0084 s	0,0087 s	0,0124 s	0,0175 s	0,0177 s

Por outro lado, para janelas de previsão mais longas, o modelo RF-r se destaca. Sua estrutura baseada em árvores permitiu modelar melhor relações não lineares que emergem à medida que a trajetória do veículo se torna mais incerta. O RF-r apresenta uma robustez maior na captura de padrões complexos, sendo capaz de ajustar-se a variações mais amplas nos dados sem comprometer significativamente a precisão.

Já nos modelos de classificação, o RF-c demonstrou desempenho superior em todas as janelas de previsão analisadas. Sua capacidade de capturar padrões complexos que se deve à combinação de múltiplas árvores, aumentam sua capacidade de generalização e reduzem o risco de superajuste. Isso refletiu diretamente na precisão da classificação das situações de tráfego, permitindo definir com maior confiabilidade o estado do modelo entre seguro, risco de colisão ou colisão. Além da precisão dos modelos, outro aspecto fundamental foi a análise dos tempos de predição apresentados. Observa-se que, mesmo com o aumento da janela de predição w, a soma da variação no tempo de processamento dos melhores modelo de regressão e classificação foi relativamente baixa. Esse crescimento moderado mostra que os modelos são escaláveis e mantêm um tempo de resposta adequado, nas janelas de previsão estudadas.

5. Conclusão

Este estudo demonstrou a eficácia de modelos de ML na previsão de colisões veiculares. Tendo o RF-c apresentado o melhor desempenho em todas as janelas de tempo, enquanto o KNN-r foi superior nos três primeiros passos de previsão, sendo superado pelo RF-r nos dois últimos. A comunicação veicular provou ser eficiente, embora desafios como perda de pacotes ainda precisem ser mitigados. Para trabalhos futuros, pretende-se incorporar o Aprendizado Federado para melhorar a escalabilidade, adaptação dos modelos e privacidade dos usuários. Além disso, pretende-se implementar outras técnicas de ML para identificar padrões incomuns no tráfego e prevenir possíveis falhas na previsão. Essa detecção de anomalias tem como objetivo aumentar a acurácia do modelo ao identificar padrões inesperados no tráfego.

References

Alagarsamy, S., Nagaraj, P., Srikanth, B., Krishna, C. V., Bharath, G., and Kalyan, S. S. (2023). A novel machine learning technique for predicting road accidents. In *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, pages 1547–1551. IEEE.

Alnuaimi, A. F. and Albaldawi, T. H. (2024). An overview of machine learning classification techniques. In *BIO Web of Conferences*, volume 97, page 00133. EDP Sciences.

Chicco, D., Warrens, M. J., and Jurman, G. (2021). The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation. *Peerj computer science*, 7:e623.

Comi, A., Hriekova, O., and Nigro, M. (2024). Exploring road safety in the era of micro-mobility: evidence from rome. *Transportation research procedia*, 78:55–62.

Gnoatto, R. and Franzen, E. (2023). Análise do desempenho de hiperparâmetros de aprendizagem de máquina aplicados na previsão da taxa de rotatividade de clientes. *Revista Destaques Acadêmicos*, 15(4).

Kumar, V. P., Chenchireddy, K., and Manohar, V. (2025). Smart road safety and vehicle accident prevention system for mountain roads. *CVR Journal of Science and Technology*, 27(1):80–84.

Parvini, M., Schulz, P., and Fettweis, G. (2024). Resource allocation in v2x networks: From classical optimization to machine learning-based solutions. *IEEE Open Journal of the Communications Society*.

Pipicelli, M., Gimelli, A., Sessa, B., De Nola, F., Toscano, G., and Di Blasio, G. (2024). Architecture and potential of connected and autonomous vehicles. *Vehicles*, 6(1):275–304.

Radi, W., Samir, R., El-Badawy, H., and Serag, E. (2024). Decentralized vehicle-to-vehicle (v2v) intelligent and sustainable communications for improving traffic safety. *Journal of Communication Sciences and Information Technology*, 3(1):9–18.

Ribeiro, B., Nicolau, M. J., and Santos, A. (2023). Using machine learning on v2x communications data for vru collision prediction. *Sensors*, 23(3):1260.

Souza, J. R. F., Oliveira, S. Z. L. N., and Oliveira, H. (2024). The impact of federated learning on urban computing. *Journal of Internet Services and Applications*, 15(1):380–409.

Veluchamy, S., Mahesh, K. M., Sheeba, P. T., et al. (2023). Deepdrive: A braking decision making approach using optimized gan and deep cnn for advanced driver assistance systems. *Engineering Applications of Artificial Intelligence*, 123:106111.

Xiang, H., Zheng, Z., Xia, X., Xu, R., Gao, L., Zhou, Z., Han, X., Ji, X., Li, M., Meng, Z., et al. (2024). V2x-real: a largs-scale dataset for vehicle-to-everything cooperative perception. In *European Conference on Computer Vision*, pages 455–470. Springer.

Zeng, T., Ferdowsi, A., Semiari, O., Saad, W., and Hong, C. S. (2024). Convergence of communications, control, and machine learning for secure and autonomous vehicle navigation. *IEEE Wireless Communications*, 31(4):132–138.