UNIVERSIDADE FEDERAL DO AMAZONAS - UFAM

INSTITUTO DE COMPUTAÇÃO- ICOMP

PROGRAMA PÓS-GRADUAÇÃO EM INFORMÁTICA - PPGI

Uso de Região de Interesse Para Tratamento de Desbalanceamento de Bases de Dados de Monitoramento de Tráfego de Redes de Acesso geradas por Adesão Voluntária

Juliana Castro da Silva

Manaus - AM

Novembro de 2022

Juliana Castro da Silva

Uso de Região de Interesse Para Tratamento de Desbalanceamento de Bases de Dados de Monitoramento de Tráfego de Redes de Acesso geradas por Adesão Voluntária

Dissertação submetida à avaliação, como requisito parcial, para a obtenção do título de Mestre em Informática no Programa de Pós-Graduação em Informática, Instituto de Computação.

Orientador(a)

César A. V. de Melo, Prof. Dr.

Universidade Federal do Amazonas - UFAM

Instituto de Computação- IComp

Manaus - AM

Novembro de 2022

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

Silva, Juliana Castro da
S586u Uso de região de interesse para tratamento de desbalanceamento de bases de dados de monitoramento de tráfego de redes de acesso geradas por adesão voluntária / Juliana Castro da Silva . 2022
77 f.: il. color; 31 cm.

Orientador: César Augusto Viana Melo
Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas.

1. desbalanceamento de bases de dados. 2. oversampling baseado em vizinhança. 3. oversampling baseado em região de interesse. 4. aprendizagem de máquina. 5. regressão linear. I. Melo, César Augusto Viana. II. Universidade Federal do Amazonas III. Título

FOLHA DE APROVAÇÃO

"Uso de Região de Interesse para Tratamento de Desbalanceamento de Bases de Dados de Monitoramento de Tráfego de Redes de Acesso geradas por Adesão Voluntária"

JULIANA CASTRO DA SILVA

Dissertação de Mestrado defendida e aprovada pela banca examinadora constituída pelos Professores:

Prof. Dr. César Augusto Viana Melo - PRESIDENTE

Prof. Dr. Gustavo Bittencourt Figueiredo - MEMBRO EXTERNO

Prof. Dr. Eduardo James Pereira Souto - MEMBRO INTERNO

Manaus, 30 de novembro de 2022

Dedico este trabalho aos meus pais, que fizeram todos os esforços possíveis para que alcançasse mais essa realização, a quem sou grata por todo o amor e dedicação.

AGRADECIMENTOS

Meus agradecimentos:

- ao Prof. César Melo, pela orientação, apoio e tempo dedicados;
- aos professores e demais colaboradores do ICOMP/UFAM pela dedicação e empenho em fazer desse instituto um local de desenvolvimento acadêmico de excelência;
- aos meus pais, Aldo e Ana e meus irmãos Neto e Arthur por sempre me apoiarem, encorajarem e acreditarem na minha capacidade de realizar esse trabalho;
- aos amigos e colegas com os quais compartilhei parte do tempo neste período;

"Todos os homens têm, por natureza, desejo de conhecer"

Aristóteles

Uso de Região de Interesse Para Tratamento de Desbalanceamento de Bases de Dados de Monitoramento de Tráfego de Redes de Acesso geradas por Adesão Voluntária

Autor: Juliana Castro da Silva

Orientador: César A. V. de Melo, Prof. Dr.

Resumo

Uma base de dados desbalanceada é caracterizada pela diferença entre a quantidade de amostras observadas entre os grupos de dados, o mais observado é chamado majoritário e o menos observado é chamado minoritário. Essa característica está presente em bases de diferentes domínios, como finanças, diagnóstico de doenças e clima. Bases de dados geradas por adesão voluntária também podem apresentar desbalanceamento, pois os dados coletados estão diretamente relacionado com o perfil social e econômico do voluntário. Em geral, a coleta desses dados é demorada e consume recursos financeiros significativos impossibilitando a extensão do período de coleta ou a repetição da coleta. Nesse contexto, a representatividade dos dados é uma questão fundamental a ser observada quando se usa essas bases de dados para treinamento de modelos de aprendizagem, por exemplo, para resolver problemas de predição e classificação com precisão significativa. Estratégias para resolver o problema de desbalanceamento têm sido propostas e avaliadas em diferentes domínios de aplicação. Essas estratégias abordam o problema tanto em nível algorítmico, em que modifica-se os modelos de aprendizagem, quanto em nível de dados, em que modifica-se a distribuição estatística dos dados. No nível de dados, tem-se o método de *oversampling*, que consiste em

modificar a distribuição dos dados gerando amostras pouco observadas do grupo de interesse. A geração das amostras utiliza o conceito de vizinhança que é estabelecida por medida de similaridade, por exemplo, uma medida de distância entre amostras. Essa abordagem é implementada pelo *SMOTE for Regression* (SMOTER) e tem sido bastante difundida devido a sua simplicidade. A maior crítica a essa abordagem é desconsiderar a região em que a amostra é gerada, o que pode produzir amostras com valores inadequados de atributos. Para superar as dificuldades identificadas nos métodos baseados em vizinhança, outra abordagem, que propõe a geração de amostras a partir da identificação da região de interesse, é implementada pelo método *Radial-Based Oversampling* (RBO). Esse método usa uma função de base radial para caracterizar as regiões de interesse de geração de novas amostras. A principal crítica a esse método é o alto custo computacional dessa operação, tornando o seu uso inviável em grandes conjuntos de dados. Este trabalho apresenta um método, extensão do método RBO, para tratar o desbalanceamento de bases, também baseado em região de interesse, que supera as limitações características do RBO. As avaliações realizadas usando as bases de dados do projeto Neubot coletadas por 06 anos, com mais de 12 milhões de registros de sensoriamento de sessões de streaming de vídeo, mostram a eficiência do método na geração das amostras. A qualidade das amostras geradas foi avaliada sob diferentes perspectivas, inclusive quando elas são utilizadas para treinar modelos de regressão.

Palavras-chave: desbalanceamento de bases de dados, *oversampling*, *oversampling* baseado em vizinhança, *oversampling* baseado em região de interesse, aprendizagem de máquina, regressão linear

Uso de Região de Interesse Para Tratamento de Desbalanceamento de Bases de Dados de Monitoramento de Tráfego de Redes de Acesso geradas por Adesão Voluntária

Autor: Juliana Castro da Silva

Orientador: César A. V. de Melo, Prof. Dr.

Abstract

An unbalanced dataset is characterized by a significant difference among groups of data. These groups have been named the majority group, i.e., it has a large majority number of samples, and the minority group, i.e., it has a small number of samples. This pattern has been observed in datasets from different domains, e.g., finance, weather, and medical diagnostics. More recently, datasets collected using crowdsourcing techniques were put in this basket due to the social and economic profile of gathered volunteers. In general, the process of collecting data is costly and time-consuming which imposes severe restrictions to extend the collecting period or repeat the process to acquire more data or to improve the quality of acquired data. Moreover, the most wanted learning characteristics are misrepresented in the minority group. In this context, data representativeness is a key issue in using those datasets for training Machine Learning models, for instance, to solve classification and prediction problems with significant accuracy. Therefore, strategies for solving the unbalanced dataset problem have been proposed by using an algorithmic approach, i.e., it changes the learning algorithm, or using a data-driven approach, i.e., it changes the data distribution probability. Oversampling

is a data-driven approach and works by changing the data distribution through sampling and patching the minority group. This sampling happens based on the concept of a neighborhood which is established by measuring the similarity among samples of the minority group, for instance, using Euclidean distance. The SMOTER, SMOTE for Regression, implements neighborhood-based oversampling and has been widely considered due to its simplicity and acceptable accuracy. The neighborhood-based approaches suffer from the inlay-regions problem, i.e., they ignore the existence of inlay minority regions, which induces the neighborhood-based algorithms to sample data with inappropriate values. For overcoming this problem, the concept of the region of interest is defined and used to guide the sampling. Radial-Based Oversampling - RBO is driven by this concept. It applies a Radial-based kernel function to characterize the regions of interest and induce the sampling. In this work, we present a novel method, named RBO-QS, for unbalanced datasets which overcomes the identified drawbacks of the RBO method. The numerical studies show that the proposed methods can do the sampling in an efficient and accurate way. The quality of data samples was evaluated under different criteria which includes the regression model training. The dataset used to carry out the experimental studies was collected during six years and has over 12 million sensing entries of video streaming sessions.

Keywords: imbalanced datasets, oversampling, neighborhood-based oversampling, region of interest, machine learning, linear regression

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1

INTRODUÇÃO

O problema de desbalanceamento se caracteriza pela desproporção entre os grupos de instâncias de um conjunto de dados (LI et al., 2018) (XIE et al., 2019). O grupo de dados mais representado é chamado majoritário, e chama-se minoritário o grupo menos representado. Bases de dados com essa característica são empregadas em tarefas como regressão e classificação e são encontradas em diversos contextos. Neste trabalho, a base de dados estudada está inserida no contexto do streaming adaptativo, uma tecnologia usada para transmissão de vídeo pela Internet. Os registros nessa base visam caracterizar como as aplicações percebem o estado das conexões usadas para transferência de seus dados. Esse esforço faz parte do projeto Neubot que monitorou as infraestruturas de redes quanto a observância do princípio da neutralidade usando diferentes aplicações. O sensoriamento acontece por meio de Bots, que simulam a dinâmica de aplicações reais, instalados em máquinas de usuários que voluntariamente aderiram ao projeto.

As vantagens desse tipo de iniciativa são diversas e se expressam principalmente na diversidade das medições, que capturam as influências do tempo, do espaço físico, das tecnologias utilizadas, e das matizes sociais dos voluntários. Esse tipo de sensoriamento, conhecido como crowdsourcing, tem sido realizado em diversos contexto, e.g., monitoramento do qualidade do ar(HUANG et al., 2019), de vias públicas(LUCIC et al., 2020), de riscos de desastre(KANKANAMGE et al., 2019), e infraestruturas de comunicação(BASSO et al., 2014). A sistematização do sensoriamento com participação voluntária é uma questão estudada nos diversos contextos(HOßFELD et al., 2014), visando a otimização dos recursos empregados frente aos resultados obtidos, i.e. as base

de dados coletadas. A base de dados usada neste trabalho foi coletada usando a técnica de *crowdsourcing* e, embora sua metodologia de medição tenha no tempo um fator central, outros fatores, em especial a matiz social e a tecnologia usada pelo voluntário, podem ter influenciado as coletas realizadas. Dessa forma, avalia-se a presença do desbalanceamento nas bases coletadas, considerando os aspectos temporais, tecnológicos e espaciais associados às medições.

O uso de tais bases para a criação de modelos de aprendizagem de máquina é um desafio que tem sido abordado de duas formas: a nível de dados e a nível de algoritmo. A última abordagem trabalha modificando os algoritmos empregados nas tarefas de aprendizado, enquanto a primeira trabalha o conjunto de dados, modificando sua distribuição, por meio da remoção de instâncias, o chamado *undersampling*, ou por meio da criação de instâncias, o chamado *oversampling*.

Neste trabalho, estudam-se duas soluções a nível de dados que implementam o *oversampling*. A primeira solução é chamada SMOTER (TORGO et al., 2013), uma versão do SMOTE (CHAWLA et al., 2002), que baseia-se no conceito de vizinhança para gerar novas amostras. A segunda solução é chamada RBO - *Radial-Based Oversampling* (KOZIARSKI; KRAWCZYK; WOŹNIAK, 2019), que implementa o conceito de região de interesse para gerar novas amostras que caracterizam-se por permanecerem nas regiões caracterizadas.

1.1 Problema

A base de dados Neubot apresenta características que tornam as duas abordagens mencionadas anteriormente inadequadas para o tratamento do desbalanceamento. Seja porque os dados gerados no processo são caracterizados como ruídos, impactando negativamente o treinamento dos modelos; seja porque o volume de dados da base torna o método computacionalmente ineficiente. O método SMOTER (TORGO et al., 2013) baseia-se no conceito de vizinhança para gerar novas amostras, que tem como principal limitação a geração de amostras do grupo minoritário em região do grupo majoritário, o que implica em amostras com características inéditas, e pouco relacionadas com aquelas

do grupo alvo.

O RBO (KOZIARSKI; KRAWCZYK; WOŹNIAK, 2019) lida com a limitação da abordagem baseada em vizinhança por meio do conceito de região de interesse, definida por uma função kernel que caracteriza as regiões pela semelhanças das amostras, evitando o uso de amostras com pouca relação no processo de geração. Contudo, o exploração do espaço de busca, visando a caracterização das regiões, tem um alto custo computacional, o que torna a sua utilização inviável para grandes bases de dados.

Para superar essas limitações, neste trabalho propõe-se o emprego do RBO a partir de duas melhorias algorítmicas. A primeira melhoria consiste na estratificação da base de dados, que gera subconjuntos menores sobre os quais o método será aplicado. A segunda melhoria altera a forma como a região de interesse é explorada na geração das amostras, usando o conceito de quadrante. O uso do quadrante caracteriza uma abordagem gulosa no processo de exploração das vizinhanças, que nos resultados se mostrou bastante eficiente. O potencial de geração do métodos SMOTER, RBO e a versão proposta, chamada de RBO-QS *Radial-based Oversampling using Quadrant Search and Stratification*, são analisados considerando o impacto das amostras geradas no desbalanceamento. Adicionalmente, avalia-se a similaridade das amostras geradas com as amostras do conjunto minoritário, em diferentes escala de tempo,i.e., dia da semana, horário do dia e faixas de horário. Por fim, três métodos de aprendizado para regressão são usados para avaliar o impacto das amostras geradas pelo RBO-QS no aprendizado sobre as bases usadas.

1.2 Objetivo

Este trabalho tem como objetivo propor e avaliar uma solução para o desbalanceamento de bases de dados lidando com as limitação de algoritmos que usam funções de base radial para caracterização de regiões de interesse, descrita no trabalho seminal que propôs o RBO. Para alcançar esse objetivo, foram definidos os seguintes objetivos específicos:

- Caracterizar o desbalanceamento presente em base de dados de monitoramento

de redes de acesso, gerada por adesão voluntária, projeto Neubot;

- Evidenciar o impacto do desbalanceamento no treinamento de um modelo de aprendizado profundo;
- Evidenciar a ineficácia dos métodos baseado em vizinhança no tratamento do desbalanceamento de bases de dados de monitoramento de redes de acesso, gerada por adesão voluntária;
- Implementar um método para desbalanceamento de base de dados que usa região de interesse baseado em quadrantes e estratificação;

1.3 Contribuições

A partir dos objetivos definidos neste trabalho, as contribuições realizadas são:

- Caracterização do desbalanceamento presente em base de dados de monitoramento de redes de acesso, gerada por adesão voluntária, projeto Neubot;
- Evidenciação do impacto do desbalanceamento no treinamento de um modelo de aprendizado profundo;
- Evidenciação da ineficácia dos métodos baseado em vizinhança no tratamento do desbalanceamento de bases de dados de monitoramento de redes de acesso, gerada por adesão voluntária;
- Um novo método para desbalanceamento de base de dados que usa região de interesse baseado em quadrantes e estratificação;

1.4 Organização do Trabalho

O restante deste trabalho está organizado da seguinte forma: no Capítulo 2 são apresentados conceitos necessários para a compreensão do trabalho: como é caracterizado o problema do desbalanceamento, as diferentes técnicas empregadas para tratar esse

problema, principalmente a nível de dados por meio da geração de amostras e também o conceito de *streaming* adaptativo, contexto em que está inserido a base de dados usada neste trabalho.

No Capítulo 3 é caracterizada a base de dados empregada neste trabalho, como ela está organizada e quais informações estão presentes; são apresentadas ainda análises a respeito do desbalanceamento presente em bases coletadas em diferentes anos; no Capítulo 4 apresenta-se um estudo sobre a adequação do método SMOTER para tratar desbalanceamento, analisando a relação entre as amostras usadas para gerar novas amostras e as amostras geradas e o efeito do método na base de dados considerando características específicas.

O Capítulo 5 apresenta o método proposto e o método base, com análise do efeito de ambos em relação a características específicas, e avaliação do impacto do método proposto no aprendizado de modelos de regressão e no Capítulo 6 as conclusões, retomando o que foi apresentado neste trabalho, os resultados obtidos e as possibilidades de trabalhos futuros.

2

FUNDAMENTOS

Neste capítulo são apresentados os termos e conceitos relevantes para a compreensão do tema abordado.

O problema do desbalanceamento, descrito na Seção 2.1, caracteriza-se pela diferença na quantidade de representações entre grupos de dados, tendo um grupo com maior representação em relação a outro. A Seção 2.2 apresenta abordagens que buscam lidar com esse problema, principalmente aquelas que modificam a distribuição dos dados por meio de geração de amostras.

Esse problema pode estar presente em bases de dados de diferentes domínios, na Seção 2.3 apresenta-se o conceito de *streaming* adaptativo, uma forma de transmissão de vídeo que se adapta de acordo com aspectos definidos, como largura de banda, contexto no qual o conjunto de dados desse estudo está inserido.

2.1 Desbalanceamento de bases de dados

Bases de dados desbalanceadas apresentam como característica principal a diferença entre a quantidade de instâncias observadas pertencentes a diferentes grupos de domínio. Neste tipo de conjunto de dados, o grupo que possui maior número de instâncias é chamado classe majoritária ou negativa, enquanto o de menor quantidade é chamada classe minoritária ou positiva (LEEVY et al., 2018).

O grau do desbalanceamento entre esses grupos pode variar de leve a severo e pode de ser definido pela proporção entre as classes majoritária e minoritária, entre

$100 : 1$ e $10000 : 1$ considera-se um alto grau de desbalanceamento. No contexto de problemas que utilizam esse tipo de bases de dados, a classe minoritária normalmente é a classe de interesse já que tem menor representação, o que dificulta o aprendizado sobre ela, ao contrário da majoritária que é mais comumente observada.

Esse tipo de base de dados é encontrado em diferentes domínios, um exemplo é (JAIN, 2020). Esta é uma base usada para pesquisa em segurança cibernética, que apresenta dados de tráfego de rede normais e sob ataque, com $7.049.989$ observações, deste total $93,93\%$ representam o tráfego normal, ou seja, classe majoritária e $6,07\%$ o tráfego sob ataque, a classe minoritária. O desbalanceamento também pode ser encontrado em tarefas de regressão, nas quais estima-se valores reais para uma variável. Neste caso o desbalanceamento caracteriza-se pela pouca representação de valores nos quais se tem mais interesse, chamados valores raros (MONIZ; BRANCO; TORGO, 2017), é equivalente à classe minoritária. Os valores mais observados são chamado valores normais, equivalente à classe majoritária.

Tanto na classificação quanto na regressão, o desbalanceamento afeta o desempenho dos modelos empregados. Classificadores acabam sofrendo com o viés em relação à classe majoritária, classificando instâncias como pertencentes a classe majoritária, mesmo as que não pertencem, os chamados falso negativos que podem ser mais custosos para o domínio do problema do que um falso positivo. De forma similar, na regressão, os modelos também sofrem com o viés das instâncias mais observadas, se distanciando dos valores de interesse.

2.2 Técnicas para tratamento do desbalanceamento

Modelos de aprendizado de classificação e regressão implicitamente assumem que o conjunto de dados sobre o qual são aplicados apresentam uma distribuição balanceada dos dados, o que não ocorre em muitas aplicações reais, ocorrendo o conhecido problema do desbalanceamento, em que há desproporção entre a quantidade de exemplos observados entre os diferentes tipos de dado do domínio do problema (Cao et al., 2013). Isso acaba degradando o desempenho dos modelos, que podem sofrer com o viés de

exemplos mais recorrentes sobre os menos recorrentes.

Para se tratar esse desbalanceamento, há dois grupos de abordagens: a nível de dado ou de algoritmo (LEEVY et al., 2018). A primeira abordagem trabalha sobre o conjunto de dados utilizado, pode ser sub agrupada em métodos de *data sampling*, que alteram a distribuição dos dados, e de *feature selection*, que ajudam a selecionar os atributos mais relevantes para a diferenciação entre as instâncias. Métodos de *data sampling* alteram a distribuição dos dados modificando o número de amostras, podem ser divididos em dois subgrupos: os métodos de *oversampling* e os de *undersampling*.

Métodos de *oversampling* alteram a distribuição dos dados adicionando instâncias normalmente do tipo menos observado ao conjunto de dados, replicando-as de forma aleatória ou de acordo com um algoritmo enquanto os métodos de *undersampling* removem instâncias do tipo mais observado, também de forma aleatória ou de acordo com algum algoritmo. A abordagem a nível de algoritmo trabalha modificando modelos e alterando parâmetros para reduzir os efeitos causados pelo desbalanceamento, e pode ser utilizada em combinação com métodos a nível de dados.

2.2.1 Oversampling

Métodos a nível de dados têm sido bastante explorados para tratar o desbalanceamento de bases de dados, principalmente na tarefa de classificação. A principal vantagem desse tipo de abordagem é a independência dos modelos utilizados, pois modifica-se o conjunto de dados, podendo ser aplicado diferentes modelos de aprendizado. Há métodos com diferentes estratégias para gerar as amostras desejadas, como considerar o contexto de vizinhos próximos, usar funções que descrevam os dados ou modelos generativos, por exemplo. Uma das desvantagens é que as amostras adicionadas podem adicionar um nível de ruído que acabe prejudicando o aprendizado sobre o conjunto de dados.

Mais recentemente, modelos generativos de aprendizado profundo como as GANs (*generative adversarial networks*) ganharam muito destaque como soluções que apresentam ótimo desempenho em diversas aplicações, e passaram a ser aplicadas

também na geração de dados sintéticos, sobretudo quando se trata de imagens. Normalmente apresentam uma estrutura mais complexa e requerem uma grande quantidade de dados a partir dos quais possam extrair padrões que caracterizem os dados sobre os quais aprendem.

Uma solução que faz uso de aprendizado profundo para tratar o problema de desbalanceamento é a proposta por (YI; SUN; HE, 2018), que utiliza CGANs para expandir a base de dados de imagens para reconhecimento facial de emoções. De forma semelhante, (LI et al., 2018) utilizam GANs para sintetizar imagens de tempo a partir de um conjunto de dados desbalanceado em uma tarefa de classificação, gerando amostras sintéticas da classe minoritária.

Quanto a trabalhos que buscam aprender a distribuição dos dados para gerar amostras, (XIE et al., 2019) propõem um solução baseada em distribuições gaussianas, aprendendo a distribuição geral do conjunto de dados com base em parâmetros como média e variância. Essa abordagem, no entanto, não é tão explorada quanto a baseada em vizinhança, como descrito a seguir.

2.2.2 Oversampling baseado em similaridade de vizinhança

Métodos baseados em vizinhança consideram a informação local, ou seja, encontram e utilizam informação de amostras vizinhas para gerar novas amostras. Para definir os vizinhos mais próximos são usadas métricas de distância, como a Distância Euclidiana. Dos métodos que fazem *oversampling*, um dos trabalhos mais relevantes, o SMOTE (CHAWLA et al., 2002), considerado um método simples, é um dos primeiros deste tipo. Este método realiza uma interpolação entre uma amostra e um vizinho aleatório escolhido entre os k mais próximos para gerar uma nova amostra da classe minoritária . Também serviu de ponto de partida para diversos outros métodos que fazem adaptação dele, muito mais explorados no contexto de classificação do que regressão.

Um dos métodos derivados do SMOTE é o CGMOS (ZHANG et al., 2016), que cria novas instâncias de acordo com a mudança de certeza que a adição de cada nova instância gera no conjunto de dados como um todo. Primeiro calcula a certeza do

conjunto de dados considerando ambas as classes, para cada instância atribui um peso calculado como a mudança de certeza gerada se uma instância da classe minoritária for adicionada na mesma posição da instância avaliada. Esse peso determina a probabilidade de essa instância ser selecionada como semente, quanto maior o peso, maior a probabilidade de ser selecionada. As novas instâncias são adicionadas onde podem melhorar a certeza sobre o conjunto de dados. Outro exemplo é o ADASYN (HE et al., 2008), que gera instâncias de acordo com a dificuldade de aprender de cada instância.

Além dos métodos voltados para classificação, embora muito menos explorado, o SMOTE também serviu de base para tratar desbalanceamento de bases de dados usadas em tarefas de regressão, é o caso do SMOTER (TORGO et al., 2013). Este método é uma adaptação do SMOTE para gerar valores reais para uma variável alvo, utilizando a abordagem de vizinhança empregada pelo SMOTE para gerar novas amostras.

2.3 Streaming Adaptativo

O aumento da demanda por serviços de vídeos influencia na forma como esse tipo de conteúdo é transmitido para sua audiência, tendo-se experimentado diferentes abordagens. As principais tecnologias usadas para a distribuição de vídeos são: *streaming* tradicional, *download* progressivo, *streaming segmentado* e *streaming* adaptativo.

No *streaming* tradicional, a mídia é dividida em partes, eliminando a necessidade de *download* completo e reproduzindo as partes da mídia individualmente no cliente assim que são recuperados. O *download* progressivo assemelha-se ao *streaming* tradicional, com armazenamento do arquivo no dispositivo do cliente.

O *streaming* segmentado utiliza o conceito de segmentos, dividindo a mídia em segmentos de mesma duração que são requisitados ao servidor para serem reproduzidos pelo cliente. Esse processo pode ser afetado pela variação de vazão que desalinha as taxas de transferência e reprodução, gerando interrupções na reprodução que prejudicam a percepção do usuário. Para contornar problemas como esse, surgiu *streaming* adaptativo que tem como uma das suas principais abordagens o HAS (*HTTP Adaptive Streaming*).

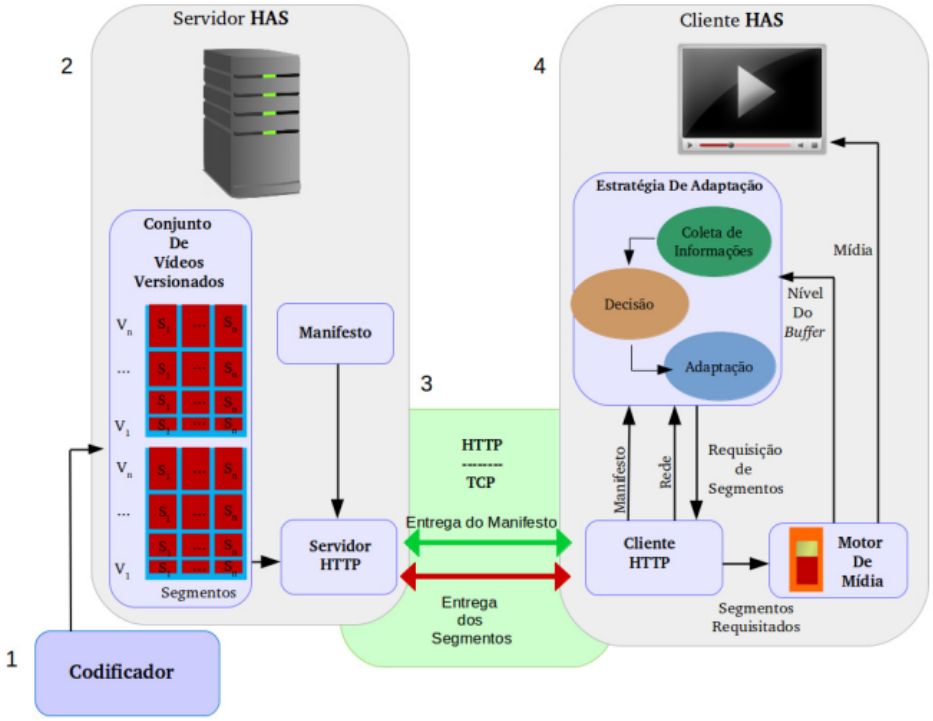


Figura 2.1 – Esquema geral de um sistema HAS. Fonte: (COELHO, 2018)

Como ilustra-se na Figura 2.1, o servidor HAS (2) mantém um conjunto de versionamento para cada vídeo, onde múltiplas versões com diferentes qualidades (fps, resolução e taxa de bits) são divididas em uma quantidade fixa de segmentos de mesma duração e alinhados no tempo. Um arquivo de descrição contém as informações sobre a qualidade e a localização de cada versão, o manifesto.

Acessando esse arquivo pelo servidor HAS, o cliente HAS (4) escolhe a qualidade mais adequada de acordo com fatores considerados, como largura de banda disponível, capacidade do dispositivo e ocupação do *buffer* de reprodução. O cliente então seleciona o segmento da qualidade mais adequada, essa adaptação dinâmica contribui para uma melhor experiência do usuário.

Para o *streaming* adaptativo, existem dois tipos de serviços para distribuição de vídeo: ao vivo e sob demanda. A principal diferença entre eles está no tempo entre produção e apresentação do *streaming* ao usuário. No serviço de vídeo sob demanda, o conteúdo é previamente codificado e então vai para plataforma de distribuição para ser publicado, podendo ser de maneira distribuída. Enquanto no serviço ao vivo essas etapas são precedidas pela gravação do conteúdo. Após o início da gravação, o fluxo é codificado em segmentos e transcodificados em diferentes versões que serão publicadas

em seguida e disponibilizadas por um único servidor. Além disso, por se tratar de um evento em tempo real, requer menor latência para que o usuário esteja alinhado ao evento.

2.4 Considerações Finais

Neste capítulo viu-se que bases desbalanceadas podem ser encontradas em diversos domínios e que o desbalanceamento, ou seja, a significativa diferença de quantidades observadas entre os grupos de problema tem impacto sobre os modelos de aprendizados, como os usados para regressão e classificação. Por isso, diversas maneiras para minimizar o impacto dessa característica sobre os modelos têm sido exploradas, com o objetivo de tornar esses conjuntos de dados mais adequados e representativos.

Uma das principais estratégias utilizadas para buscar solucionar o problema do desbalanceamento é o *oversampling*, uma abordagem a nível de dados que altera a distribuição dos dados gerando amostras baseado em vizinhança, distribuição dos dados ou usando modelos generativos. O SMOTE é um método baseado em vizinhança que serviu de ponto de partida para outros métodos, como o SMOTER, uma versão para tratar o desbalanceamento em bases de dados usadas em regressão.

Além disso, foi apresentado o conceito de *streaming* adaptativo, uma forma de transmissão de vídeo, contexto em que está inserido o Projeto Neubot para monitoramento de dados de tráfego de rede, apresentado no Capítulo 3.

3

UMA BASE DE DADOS DE MEDIÇÃO GERADA POR ADESÃO: O DATASET DO PROJETO NEUBOT

Neste capítulo apresenta-se o projeto Neubot, responsável pela coleta das bases de dados estudadas neste trabalho. O projeto Neubot está inserido no contexto do *streaming* adaptativo à medida que sua coleta de dados, realizada por adesão de voluntários, se dá sobre medições de informações de tráfego de rede durante sessões de transmissão de vídeo.

Apresenta-se também características das medições coletadas em diferentes anos, e realiza-se um estudo de caracterização geral dessas medições em relação a componentes como planos de dados, dias da semana e horário das medições realizadas, de forma a compreender melhor o comportamento das medições observadas ao longo dos anos de acordo com esses critérios.

3.1 Projeto Neubot: medindo a vazão a partir da borda

O projeto Neubot monitora os provedores de acesso à Internet para testar a neutralidade do serviço prestado (BASSO et al., 2014). Usam-se os bots instalados em máquinas de participantes voluntários, que simulam o comportamento de aplicações de Vídeo *Streaming*, P2P (do inglês, *peer-to-peer*) e *Web Browser*. Esses bots enviam requisições

para servidores instalados no M-Lab(LAB, 2020) e registram os dados e as dinâmicas observadas em cada sessão de acesso.

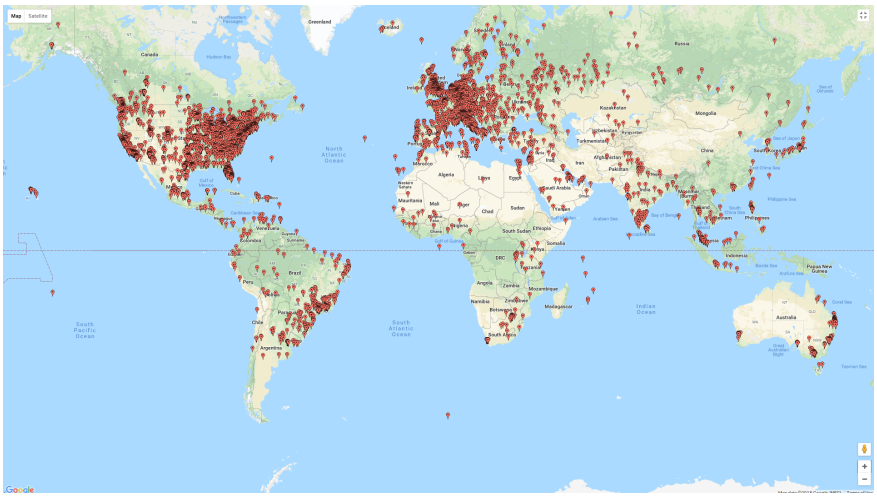


Figura 3.1 – Distribuição Geográfica dos Clientes Neubot.

A Figura 3.1 mostra como os clientes do projeto estão distribuídos pelo planeta, tendo como maior destaque na América do Sul a presença de voluntários brasileiros. O Neubot realiza aproximadamente 10.000 testes por dia envolvendo mais de 1.000 endereços IP em cerca de 100 países e 1.000 sistemas autônomos (grupo de de redes IP, sob o controle de uma gerência técnica e uma mesma política de roteamento). Os bots vídeo *streaming* estão programados para realizar dois testes, diariamente, como um serviço executando em segundo plano e a cada 30 minutos realiza uma análise, mas tal agendamento depende diretamente do padrão de conexão dos voluntários. Os voluntários que permanecem mais tempo conectado permitem um monitoramento maior das suas infraestruturas. As informações associadas a cada sessão são enviadas para um repositório público.

As instâncias do bot vídeo *streaming*, em cada sessão de acesso, geram 15 requisições para o servidor. Cada segmento requisitado contém 2 segundos de vídeo e é disponibilizado em uma das seguintes taxas de bits (BASSO et al., 2014): 100, 150, 200, 250, 300, 400, 500, 700, 900, 1.200, 1.500, 2.000, 2.500, 3.000, 4.000, 5.000, 6.000, 7.000, 10.000, 20.000 kbps. O algoritmo de adaptação da taxa de bit, implementado pelo bot vídeo *streaming* usa a vazão medida do último segmento requisitado, para definir a taxa de bits do próximo segmento. O primeiro segmento é acessado sempre na menor taxa de bits (100 kbps). O algoritmo é definido como mostrado no Algoritmo 3.1 (BASSO et

al., 2014):

if $EDT > PLAY\_TIME$ **then**
  $REL\_ERR = 1 - EDT/PLAY\_TIME$;
  $EAB = EAB + REL\_ERR * EAB$;
  $EAB = max(min\_rep\_bitrate, EAB)$;
endif
else
  $EAB = size\_of\_segment/EDT$;
  $EAB = max(all\_rep\_bitrate < EAB)$;
endif

Algorithm 3.1: Adapatação da taxa de bit.

onde EAB é a vazão estimada, o EDT é o tempo de download, o PLAY_TIME é a duração de reprodução do segmento, o *size_of_segment* é o tamanho do segmento em kbit e os *min_rep_bitrate* e *all_rep_bitrate* representam a taxa de bits mínima e todas as taxas de bits disponíveis no arquivo de manifesto (arquivo que descreve a mídia), sucessivamente.

As requisições são realizadas usando uma conexão HTTP persistente, com o servidor configurado para receber um número ilimitado de requisições, por um tempo indeterminado. Cada requisição de teste é descrita com propriedades que caracterizam o cliente Neubot, o servidor, a conexão e cada um dos segmentos acessados. Após cada teste, o cliente envia os resultados, em formato JSON, ao servidor que os armazena e disponibiliza ao público. Aos dados coletados em cada requisição foram acrescidas as seguintes informações: localização geográfica e propriedade de rede, especificamente o nome País, Cidade e Sistema Autônomo (do inglês, *Autonomous System*). Essas informações foram obtidas usando os serviços disponíveis em maxmind.com (MAXMIND, 2019).

Para a realização deste estudo, utilizou-se a base de dados disponibilizada pelo projeto Neubot, que contém medições em diversos países, e também uma versão filtrada por país, contendo somente as medições referentes ao Brasil. Neste trabalho, esta base será chamada de Neubot-BR. As singularidades de cada país, no tocante aos seus serviços de provimento de conexão à Internet, estão além do escopo deste trabalho.

Como já discutido anteriormente, uma característica dessa base é que os dados são resultado de adesão voluntária à tarefa de medir os recursos de transmissão provisionados pelo provedor do serviço de conexão à Internet. Dessa forma, é legítimo

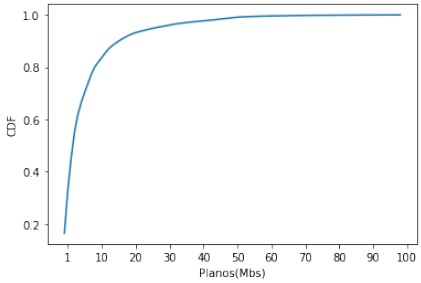
supor que a representatividade das medições é diretamente afetada pela abrangência da adesão, i.e., é preciso um número significativo de voluntários para que os diversos aspectos do fenômeno que se está medindo sejam capturados. (PAZ; MELO, 2020) descreve os resultados de um estudo realizado com a base Neubot-BR caracterizando diversos aspectos dos dados coletados, como por exemplo, frequência e ocorrência de medições em diversos períodos do dia. A conclusão é que em diversos períodos do dia não há medições ou elas são em um número pequeno, o que torna pouco confiável as conclusões que se possa derivar do comportamento observado. A inexistência de recompensas materiais nesse tipo de iniciativa torna a tarefa de incentivo a adesão um grande desafio (HIRTH et al., 2015), em especial para se obter informações específicas ou ainda não capturadas do fenômeno em observação.

3.2 Caracterização da base

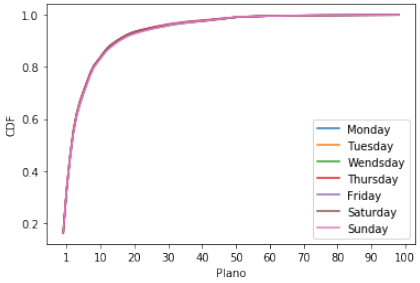
Esta seção descreve uma caracterização da base de dados do projeto Neubot, apresentando um estudo sobre a distribuição das medições a partir do plano de dados identificado e associado aos bots. A Figura 3.2 apresenta os resultados da avaliação conduzida sobre os dados coletados no ano de 2018. Os bots que coletam os dados estão conectados por diferentes planos de dados, que foram neste estudo caracterizados por taxas de vazão máxima alcançada durante uma sessão. Ao todo, foram identificadas 11 classes de plano de dados, caracterizados na Tabela 3.1.

A Figura 3.2a mostra uma visão geral do comportamento das sessões de acordo com os planos de dados estabelecidos, indicando a probabilidade de uma sessão estar dentro da faixa de um plano de dado. Observa-se que para os planos de dados mais baixos, as probabilidade de ocorrência de uma sessão é maior que comparada com os planos de dados maiores.

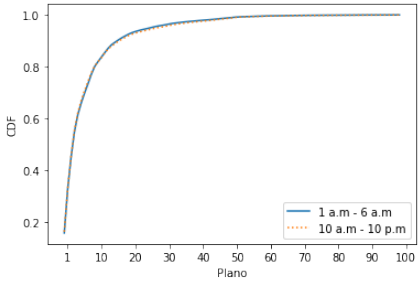
A Figura 3.2b apresenta a probabilidade de ocorrência de sessões para cada um dos planos de dados em cada um dos dias da semana, considerando o dia da semana em que a sessão iniciou. Em todos os dias é observado um comportamento similar: maior probabilidade de ocorrência de sessões de planos de dados menores, para planos



(a) Ocorrência de sessões



(b) Ocorrência de sessões por dias da semana



(c) Intervalos de horários

Figura 3.2 – CDF das sessões de planos de dados na base de 2018.

maiores a curva se mantém quase estável devido suas baixas frequências.

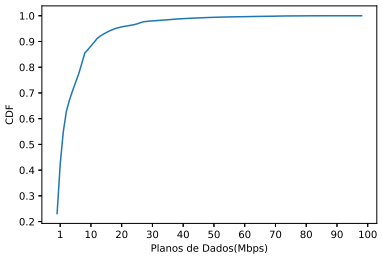
A Figura 3.2c apresenta o comportamento das sessões observados em dois intervalos distintos de horários: 1h e 6h da manhã e 10h da manhã e 10h da noite, considerando o horário em que foram iniciadas. Embora para cada um dos intervalos de horários haja uma grande diferença no número de sessões realizadas, suas curvas tem comportamento muito similar. Assim como é observado na Figura 3.2b, isso ocorre porque apesar da diferença na quantidade de observações, as sessões classificadas como de planos de dados mais baixos são maioria em relação aos demais, mantendo a proporção.

Para todos os casos apresentados, foram considerados plano até 500Mbps, mas limitou-se a 100Mbps para melhor visualização e porque para planos maiores que 100Mbps a curva permanece estável, sem apresentar grande variação.

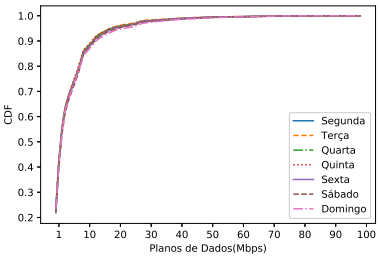
O comportamento verificado nessa base de dados apresenta similaridade com as demais bases? Para responder a essa pergunta, o mesmo estudo foi realizado com as bases coletadas nos demais anos, 2015 a 2017, 2019 e 2020. As Figuras 3.3,3.4 e 3.5 apresentam os resultados dessas avaliações. A conclusão geral é que em todos os anos analisados tem-se o mesmo padrão, independente da escala de tempo considerada, i.e., se mais longas(anual), curtas(dias da semana) e curtíssimas(períodos do dia). O padrão

	Sessões por ano			
Plano de Dados (Mbps)	2015	2016	2017	2018
[1, 10]	102643	139703	132078	279044
(10, 20]	11439	13527	19957	41707
(20, 30]	3268	7656	6253	11749
(30, 40]	1142	1909	4139	6519
(40, 50]	689	1459	5192	4643
(50, 60]	361	415	1714	2498
(60, 70]	230	55	592	807
(70, 80]	179	41	212	372
(80, 90]	44	11	147	278
(90, 100]	5	14	78	132
(100, 500]	4	23	104	210
Total	120004	164813	170466	347959

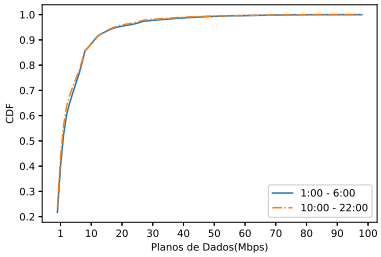
Tabela 3.1 – Planos de dados e suas sessões nos quatro anos avaliados.



(a) Ocorrência de sessões.



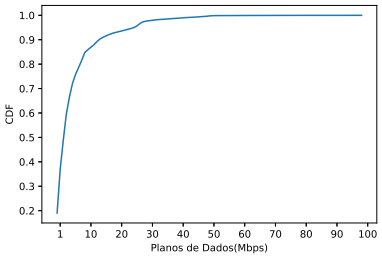
(b) Ocorrência de sessões por Dias da semana.



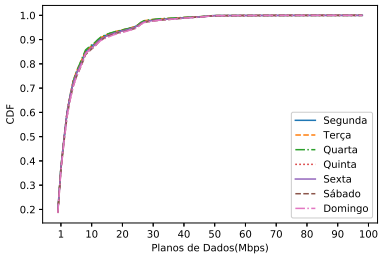
(c) Ocorrência de Sessões por Intervalos de horários.

Figura 3.3 – CDF das sessões dos planos de dados na base de 2015.

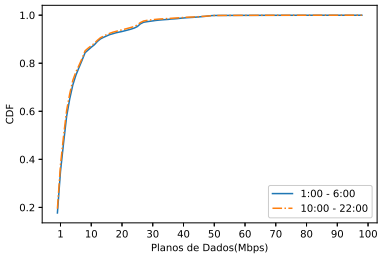
observado é a concentração de medições em um subconjunto de planos, que tem vazão sempre menor ou igual a 10MBps. Em todas as escalas de tempo observada esses planos de dados sempre representaram 80% de todas as entradas da base.



(a) Ocorrência de sessões.

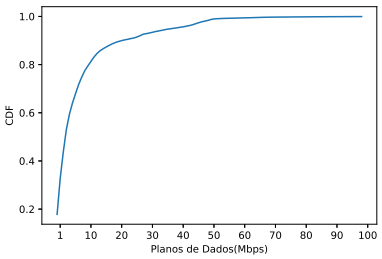


(b) Ocorrência de sessões por Dias da semana.

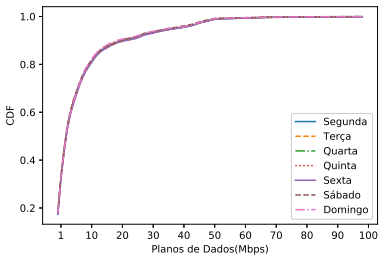


(c) Ocorrência de Sessões por Intervalos de horários.

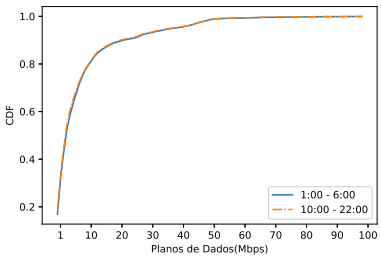
Figura 3.4 – CDF das sessões dos planos de dados na base de 2016.



(a) Ocorrência de sessões.



(b) Ocorrência de sessões por Dias da semana.

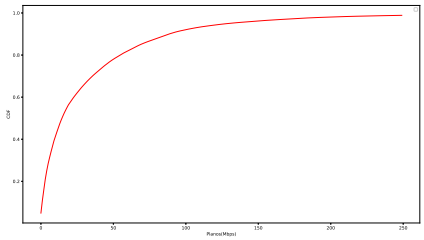


(c) Ocorrência de Sessões por Intervalos de horários.

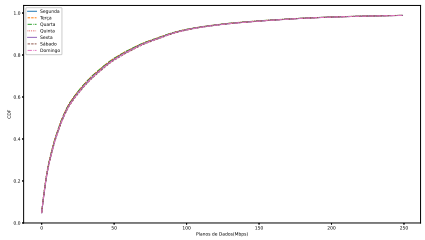
Figura 3.5 – CDF das sessões dos planos de dados na base de 2017.

3.3 Visão da base a partir do Provedor de Acesso

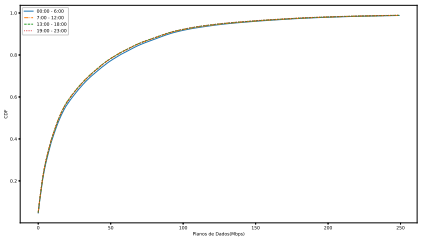
Outra pergunta formulada foi se o padrão de distribuição observado na base possuía alguma relação com os provedores de acesso. Em outras palavras, o padrão até então



(a) Ocorrência de sessões.

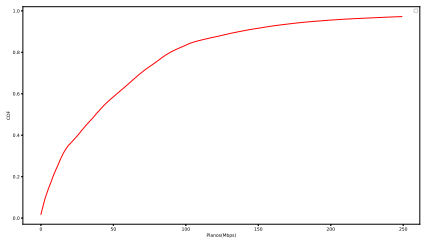


(b) Ocorrência de sessões por Dias da semana.

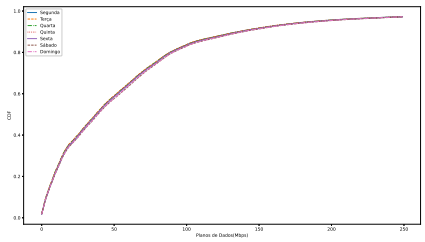


(c) Ocorrência de Sessões por Intervalos de horários.

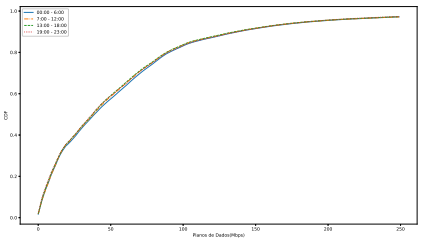
Figura 3.6 – CDF das sessões dos planos de dados na base de 2019.



(a) Ocorrência de sessões.



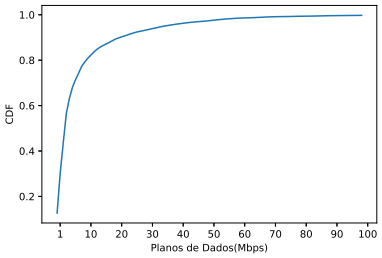
(b) Ocorrência de sessões por Dias da semana.



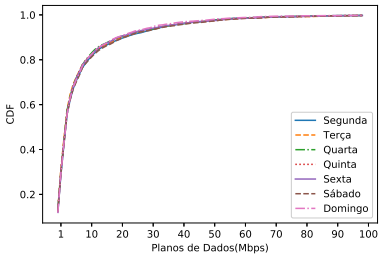
(c) Ocorrência de Sessões por Intervalos de horários.

Figura 3.7 – CDF das sessões dos planos de dados na base de 2020.

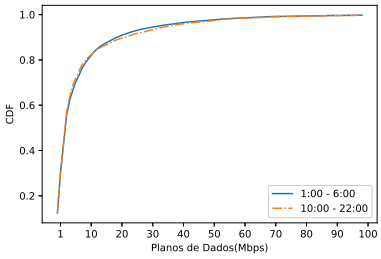
identificado, i.e., concentração de entradas em uma pequena faixa de planos de dados, permanece ocorrendo quando se observa a coleta de dados realizada em um provedor específico? Para responder a essa questão, foram extraídas as sessões correspondentes aos dois provedores de acesso com maior presença na base, que são a Telefônica Brasil



(a) Ocorrência de sessões.

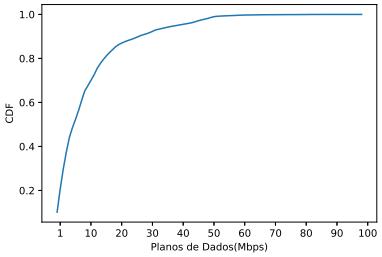


(b) Ocorrência de sessões por Dias da semana.

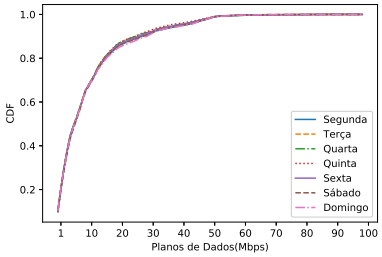


(c) Intervalos de horários.

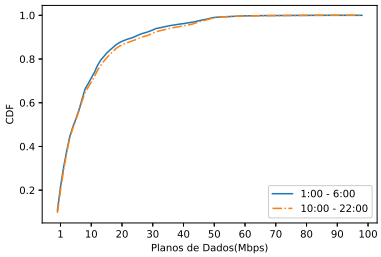
Figura 3.8 – Sessões de medição do provedor Telefônica Brasil S.A em 2018.



(a) Ocorrência de sessões.



(b) Ocorrência de sessões por Dias da semana.



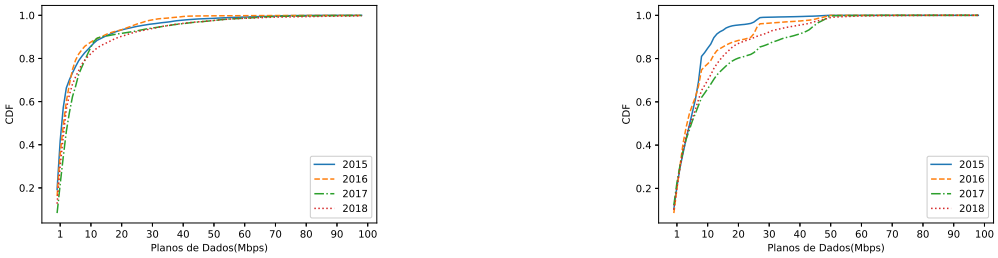
(c) Intervalos de horários.

Figura 3.9 – Sessões de medição do provedor Claro S.A em 2018.

S.A. e a Claro S.A.

A Figura 3.10 mostra a distribuição das entradas nos quatro anos analisados, naqueles dois provedores.

A conclusão geral é muito similar àquela verificada quando se estudou toda a base. Embora apresentem alguma variação ao longo dos anos, as sessões de planos de



(a) Telefônica S.A. (b) Claro S.A.

Figura 3.10 – CDF das sessões dos planos de dados com ISPs com maior número de entradas na base.

dados menores são prevalentes em relação as entradas geradas por planos de dados de maior taxa.

3.4 Engenharia de Atributos

A engenharia de atributos tem como objetivo realizar um conjunto de operações sobre o conjunto de dados de forma a torná-lo mais adequado, removendo ruídos e tornando atributos mais descritivos. A engenharia aplicada na base Neubot e Neubot-BR é apresentada a seguir, mostrando os atributos derivados a partir dos dados coletados, bem como o tratamento aplicado aos dados coletados e registrados na base. Cada instância do *bot* vídeo *streaming* requisita 15 segmentos e registra individualmente os dados associadas a cada requisição. Parte dos dados registrados varia ao longo da sessão como, por exemplo, tamanho dos segmentos e duração da transmissão do segmento, outra parte é fixa como, por exemplo, endereço IP das máquinas envolvidas na transmissão, e identificador do coletor.

O conceito de sessão foi utilizado permitindo que se agrupasse dados fixos, mantendo os dados que variam ao longo da sessão e descartando dados que representavam sessões incompletas. A Tabela 3.2 lista os atributos variáveis e não variáveis de uma sessão.

Os atributos dia da semana, dia, mês, ano, hora, minuto e segundo foram derivados a partir do registro de tempo (*timestamp*) que cada requisição apresenta. Além desses atributos, os dados coletados por sessão permitem que se derivem a taxa de

Não Variáveis	Variáveis
real_address	delta_user_time
AS_name	elapsed_target
remote_address	received
platform	connect_time
city	iteration
internal_address	elapsed
weekday	rate
	delta_sys_time
	request_ticks
	day
	month
	year
	hour
	minute
	second
	download_rate
	max_download_rate

Tabela 3.2 – Atributos registrados durante uma sessão *streaming* de vídeo.

download apresentada durante a transmissão de cada segmento, e a taxa máxima de *download* observada durante a sessão.

A base Neubot-BR possui dimensão definida por 25 atributos, entretanto alguns desses atributos apresentam escalas grandes. Por exemplo, o endereço IP apresenta 2^{32} entradas possíveis, a taxa de *download* varia de poucos kbytes a centenas de Mbytes. Essas características precisam ser consideradas na hora de se escolher a técnica que tratará o desbalanceamento da base. A criação de modelos guiados por dado, nesse cenário, demandará tratamentos para essa característica em função da dificuldade de se treinar modelos nesse cenário.

3.5 O Desbalanceamento da Base Neubot

Esta seção apresenta o resultado de estudo realizado para observar o impacto do desbalanceamento em um modelo de predição que foi concebido a partir da base Neubot. O modelo considerado utiliza técnicas de aprendizado de máquina profundo.

Para a realização desse estudo, os dados de entrada do modelo de predição

foram ordenados cronologicamente. Após a fase de treinamento, o aprendizado foi avaliado de duas formas: i) utilizando o conjunto de teste sem distinção de classe e ii) utilizando o conjunto de teste com distinção de classe majoritária e minoritária. Na classe majoritária estão as sessões de medição cuja vazão máxima foi menor igual a 30Mbps, e na classe minoritária estão os planos cuja vazão é maior que 30Mbps, conforme análise apresentada nas seções anteriores.

O modelo utilizado, chamado de STNet (*Short-term Time-series Network*), é descrito em detalhes em (SILVA, 2020). A seguir, faremos uma breve explicação da modelagem empregada pelo autor. Seja Z uma série temporal multivariada definida da seguinte forma: $Z = \{z_1, z_2, ..., z_T\}$, onde $z_t \in \mathbb{R}^n$ e n é a dimensão da variável. As dimensões de Z são definidas por um conjunto de fatores que estão correlacionados a variável dependente, a vazão da conexão, e são registrados na base Neubot-BR.

A modelagem feita considera a predição da variável dependente *vazão*, no instante de tempo $\mathbf{z}_{T+k}$, onde k é o tamanho do horizonte de previsão. A previsão da *vazão* no instante de tempo $\mathbf{z}_{T+k}$ é denotada por $\hat{\mathbf{z}_{T+k}}$. Utilizou-se um horizonte de 30 segundos em razão da característica dos conjuntos de sessões do fluxo de adaptação que envolvem 15 segmentos com duração de 2 segundos cada um.

O modelo STNet utiliza uma *Long Short-Term Memory* (LSTM) (HOCHREITER; SCHMIDHUBER, 1997) com integração de quatro componentes: camada de convolução, camada recorrente, camada de atenção temporal e um componente auto-regressivo.

A camada de convolução tem a caraterística de extrair padrões de curto prazo e padrões de dependência entre variáveis locais. A camada recorrente é uma BiLSTM. A intuição por trás de tal rede consiste em dividir os neurônios de estado de uma rede neural recorrente regular em uma parte responsável pela direção positiva do tempo (*forward states*) e uma parte pela direção negativa do tempo (*backward states*)(Schuster; Paliwal, 1997). O papel do componente de atenção temporal consiste em aliviar questões como a duração da dinâmica do período de séries de tempo não sazonais, isto é, tal mecanismo de atenção analisa as informações em cada etapa de tempo anterior e seleciona as informações relevantes para ajudar a gerar os resultados para previsões multivariadas. Já o componente auto-regressivo é uma parte linear, o qual é decomposto da saída

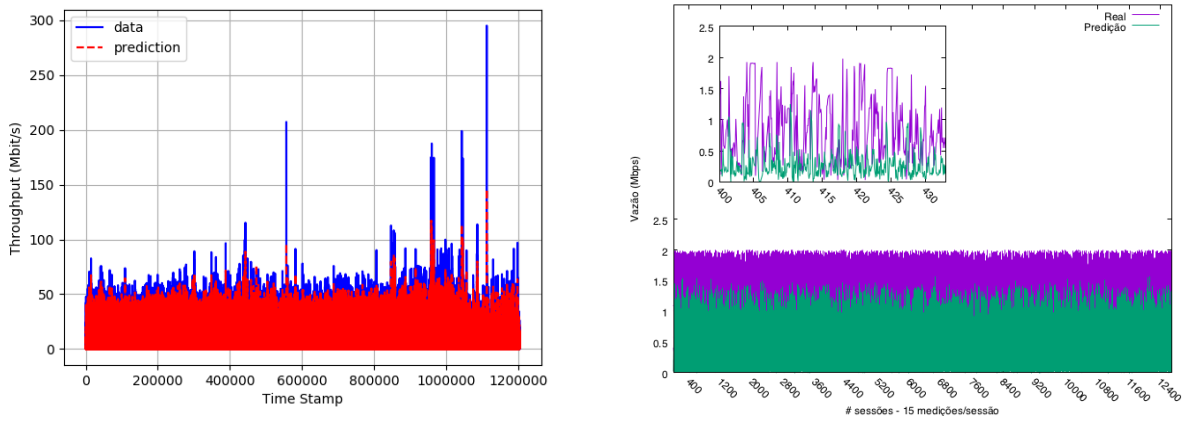
não-linear da rede neural, em razão da natureza não-linear da rede convolucional e componentes recorrentes.

Para avaliação do modelo, a base Neubot foi dividida em conjunto de treino (80%), validação (10%) e teste (10%) na ordem cronológica. Utilizou-se como medida de desempenho o *Mean Absolute Error* (MAE) de acordo com a seguinte fórmula:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|Z_{it} - \hat{Z}_{it}\right| \tag{3.1}$$

onde Z e $\hat{Z}$ são os sinais originais e sinais da previsão, respectivamente.

Na Figura 3.11 apresenta-se o resultado obtido quando as sessões de medição não são discriminadas entre a classe majoritária e minoritária. A base de teste possui 80.324 sessões e cada sessão possui 15 medições.



(a) Conjunto de Teste. (b) Conjunto Teste - Sessões com até 2.0 Mbps.

Figura 3.11 – Desempenho do modelo de predição.

Conjunto	MAE
Teste	2.3362
Treino	0.0105

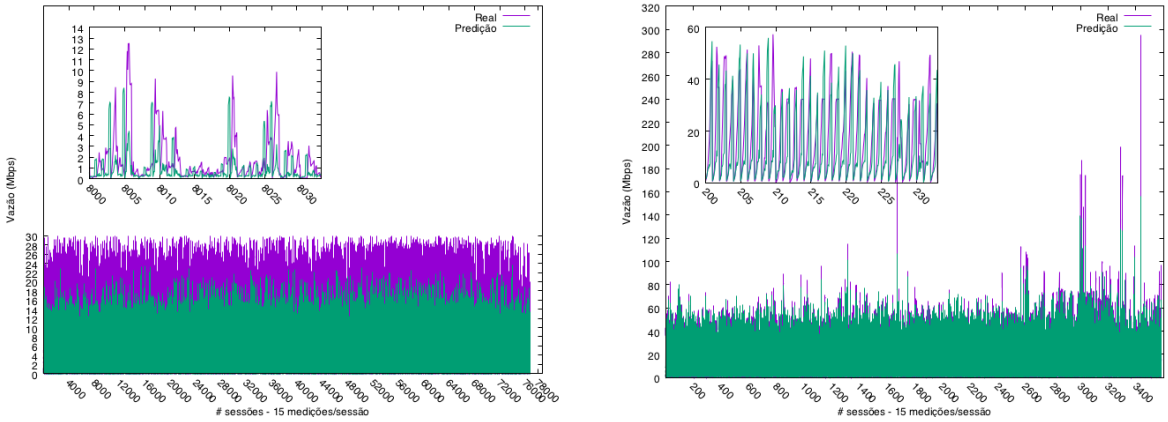
Tabela 3.3 – MAE para conjuntos de treino e teste.

A Tabela 3.3 apresenta os valores das métricas obtidos para o conjunto de treino e teste.

A conclusão geral é que o modelo consegue realizar predição da vazão com menor taxa de erro até os planos com velocidade de até 2 Mbit/s (Figura 3.11b), que

respondem por 70% da base, aproximadamente. Porém, não consegue realizar uma predição precisa ou acompanhar a flutuação da vazão para planos com velocidade acima desse valor na maioria dos instantes de tempo.

A Figura 3.12 apresenta o resultado obtido quando as avaliações são realizadas com a separação do conjunto de teste, cada um contendo amostras de somente uma das classes. A Figura 3.12b apresenta o resultado obtido para o conjunto de teste contendo apenas amostras da classe minoritária com 3585 sessões e a Figura 3.12a o resultado para o conjunto de teste da classe majoritária com 76739 sessões. A Tabela 3.4 apresenta os valores obtidos em cada um desses conjuntos para a MAE.



(a) Conjunto de Teste da Classe Majoritária. (b) Conjunto de Teste da Classe Minoritária.

Figura 3.12 – Predições do modelo para conjunto de teste separado por classe.

Conjunto de Teste	MAE
Majoritária	2.5527
Minoritária	9.8399

Tabela 3.4 – MAE para o conjunto de teste separado entre classes (majoritária e minoritária).

Observa-se que para o conjunto de teste contendo apenas amostras da classe majoritária, o modelo apresentou predições com maior precisão, um erro absoluto que é 3,5 vezes menor que aquele verificado nas avaliações realizadas com a classe minoritária. Esse resultado sugere que a predição de vazão em planos de dados de maior velocidade será afetada severamente, o conjunto de dados não é suficiente para que o modelo consiga aprender de forma precisa.

No contexto de *streaming* adaptativo, esse desbalanceamento pode ter diferentes efeitos. Em relação ao provedor, pode levar a dificuldade de oferecer um conjunto de versionamento adequado ao perfil de usuários de uma determinada localização, por exemplo, em decorrência de pouca representação de certos perfis de usuários. Ou ainda, durante a sessão de *streaming* de vídeo, a dificuldade de fazer uma adaptação satisfatória, considerando as características de rede do usuário, podendo não utilizar os recursos disponíveis de forma apropriada, causando problemas como interrupções e travamentos, o que afeta a qualidade da experiência da audiência.

3.6 Consideraçãoes Finais

Neste capítulo foi apresentado o projeto Neubot, descrevendo o processo de coleta de medições registradas nas bases de dados usadas neste trabalho e suas características. Estas estão inseridas no contexto de *streaming* adaptativo, em que informações de rede durante sessões de reprodução de vídeo por adesão de voluntários.

Também foram apresentados os resultados de avaliações iniciais sobre os dados da base Neubot-BR e base geral. O objetivo foi verificar se esta base de dados apresenta algum nível de desbalanceamento e, se sim, como ele está presente. A base foi analisada sob diferentes escalas de tempo, com medições i) durante um ano todo; ii) diferentes dias da semana e iii) diferentes períodos do dia. A base também foi avaliada quanto a medições feitas em provedores de acesso específicos considerando escala de tempo anual.

Para essas avaliações adotou-se a separação de dados em grupos de planos de dados de acordo com a vazão máxima alcançada durante uma sessão. Observou-se um padrão que se repete nas análises feitas nas diferentes escalas de tempo: a predominância de ocorrências de sessões de grupos de planos de dados menores até 10 Mbit/s, que representam a maior parte das medições registradas. Isso foi observado na análise dos anos de 2015 a 2018 e também considerando as medições de provedores de acesso específicos, e também nas bases de 2019 e 2020, o que sugere que mesmo que as medições fossem realizadas por muito mais tempo, o comportamento delas

não mudaria de forma significativa já que observa-se um padrão no comportamento e características dos coletores, permanecendo assim o desbalanceamento.

4

OVERSAMPLING BASEADO EM VIZINHANÇA

Este capítulo apresenta o estudo realizado para investigar a adequação do *oversampling* baseado em vizinhança para tratar o desbalanceamento nas bases Neubot. Nessa abordagem, a hipótese formulada é que se amostras estabelecidas na vizinhança de uma amostra base forem envolvidas na geração das novas amostras então amplia-se a probabilidade de similaridade entre a amostra base e as amostras geradas. O desafio nessa abordagem é encontrar métricas que capturem com precisão a similaridade entre as amostras.

Considerou-se o uso de métrica de similaridade baseada em distância, que foi originalmente proposta na implementação do SMOTE(CHAWLA et al., 2002) e posteriormente ajustada para séries temporais em uma versão do SMOTE, chamada de SMOTER(TORGO et al., 2013).

A Seção 4.1 apresenta o SMOTER e suas premissas. A Seção 4.2 apresenta o protocolo utilizado para avaliar a aderência dos fatores temporais da sessão gerada, em relação a sessão base. A Seção 4.3 mostra o potencial de impacto que a escolha aleatória de vizinhos, entre aqueles identificados pelo KNN, pode produzir na geração das novas sessões. Na Seção 4.4 mostra-se a função de distribuição das amostras, após o método ser aplicado, e avalia-se a qualidade dessa mudança.

4.1 SMOTER: Vizinhança e Aleatoriedade

Esta seção aborda o problema do desbalanceamento usando o *oversampling* baseado no conceito de vizinhança. Nessa abordagem, novas amostras são geradas a partir de amostras da classe minoritária, que combinam-se nesse processo. Como discutido no Capítulo 2, existem vários algoritmos que tratam o problema do desbalanceamento a partir do conceito de vizinhança. O SMOTER, versão do SMOTE(CHAWLA et al., 2002), aborda o desbalanceamento em séries temporais usando o *oversampling*. O SMOTER foi proposto e avaliado em (TORGO et al., 2013).

A Figura 4.1 apresenta a dinâmica de geração de novas amostras a partir do conceito de vizinhança. No primeiro estágio, identificam-se os vizinhos com maior similaridade em relação a amostra base s_b, do conjunto minoritário. A métrica de similaridade utilizada pelo SMOTER é a distância euclidiana (d_i), implementada pelo algoritmo KNN. No segundo estágio, esses vizinhos são selecionados para participar da geração das novas amostras s_n.

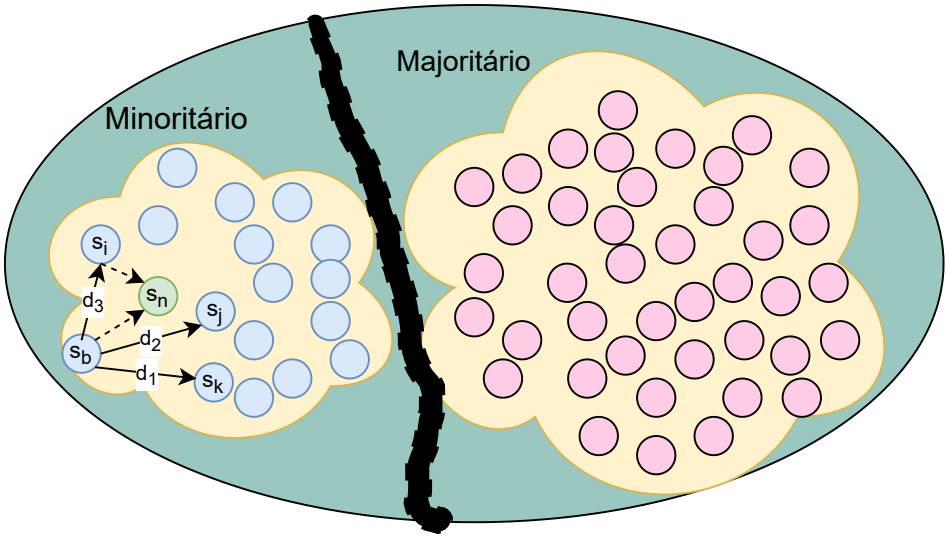


Figura 4.1 – Gerando Novas Amostras com a Vizinhança. Fonte: de autoria própria

Nesse estágio, a racionalidade é a geração de novas amostras com um certo grau de similaridade, mas não exatamente igual as amostras envolvidas no processo. Pequenas variações nas novas amostras geram diversidade para o conjunto minoritário. No caso do SMOTER, um valor aleatório entre [0,1) indica quanto da distância d_i, medida entre s_b e s_i, será considerada na geração de s_n.

O Algoritmo 4.1 apresenta o SMOTER. Nesse algoritmo, gera-se uma quantidade de amostras para cada amostra do conjunto minoritário, linhas 4 e 6, que é definida pela taxa de *oversampling* o (linha 3). Em cada amostra gerada, os valores dos atributos são calculados com base nos valores da amostra base e de um vizinho escolhido aleatoriamente entre k vizinhos mais próximos, (linha 7). No caso de atributos numéricos, usa-se interpolação para a geração das amostras (linha 11), e em caso de atributos categóricas, escolhe-se aleatoriamente entre os valores da amostra base e do vizinho escolhido, (linha 13). A variável alvo é calculada a partir da sua similaridade com a amostra base e a amostra vizinha (linhas 16, 17 e 18).

Data: Coleção de objetos da classe Minoritária D
Result: Coleção de objetos sintético minoritário S
1 Parameter: taxa de oversampling o, número de vizinhos k;
2 inicializa S = {} ;
3 ng = $o/100$;
4 **for** $case \in D$ **do**
5 $nns \leftarrow knn(k, case, D - case)$);
6 **for** $i \leftarrow 1$ **to** ng **do**
7 $x \leftarrow$ *escolha aleatória entre os vizinhos de* nns;
8 **for** $a \in attributes$ **do**
9 **if** $isNumeric(a)$ **then**
10 $diff \leftarrow case[a] - x[a]$;
11 $new[a] \leftarrow case[a] + random(0,1) * diff$;
12 **else**
13 $new[a] \leftarrow$ *escolha aleatória entre* $case[a]$ *e* $x[a]$;
14 **endif**
15 **end**
16 $d_1 \leftarrow dist(new, case)$;
17 $d_2 \leftarrow dist(new, x)$;
18 $new[target] \leftarrow \frac{d_2 * case[target] + d_1 * x[target]}{d_1 + d_2}$;
19 *incluir* new *em* S;
20 **end**
21 **end**

Algorithm 4.1: SMOTER - Oversampling baseado em vizinhança.

O estudo conduzido avaliou como os vizinhos escolhidos (linhas 5 e 7), se correlacionam com a amostra base quanto a atributos temporais, por exemplo, dia da semana e horário, e como isso impacta nas amostras geradas.

4.2 O Protocolo Empregado nas Avaliações

Esta seção apresenta o protocolo utilizado no estudo conduzido para avaliar os impactos que a aleatoriedade das escolhas, implementada no SMOTER, e a métrica de similaridade, implementada pelo KNN, produzem nas amostras geradas.

A primeira investigação conduzida considera como a escolha aleatória dos vizinhos, selecionados pelo KNN, pode influenciar na aderência aos fatores temporais da amostra base. Em outras palavras, ao escolher uma sessão, entre as k identificadas pelo KNN, para produzir a nova sessão, qual é o impacto que essa escolha pode causar naqueles fatores.

A segunda investigação analisa a similaridade entre a sessão base (S_b) e as sessões escolhidas pelo KNN, *K-Nearest Neighborhood*. Essa investigação avalia os fatores temporais das amostras identificadas pelo KNN, que são: horário e dia da semana.

A terceira investigação, considerando as faixas de horários e as ofertas do KNN para as sessões similares, verifica a aderência dessas sessões aqueles horários. Desta forma, deseja-se saber se na oferta do KNN a correlação com os horários foi mantida. Ou seja, dado que a sessão base ocorre em uma certa faixa de horário do dia, portanto espera-se que as sessões similares ocorram dentro dessa mesma faixa de horário.

Essas investigações são motivadas pelo fato do fenômeno monitorado ter uma forte componente temporal. Dessa forma, foram consideradas duas escalas de tempo: dia da semana e horário do dia, que estão definidas como segue:

- dia da semana: segunda, terça, quarta, quinta, sexta, sábado e domingo;
- faixas de horário: 0-6, 7-12, 13-18, 19-23

4.3 Aderência do Vizinho Escolhido aos Atributos Temporais.

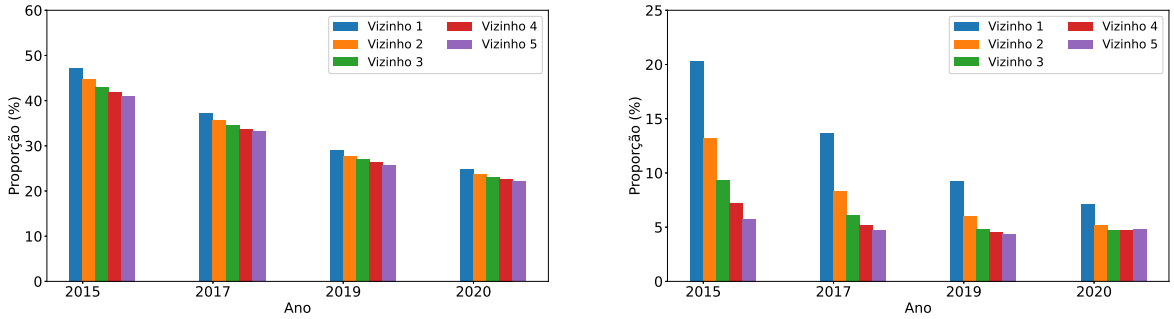
A escolha do vizinho oferecido pelo KNN, linha 7 do Algoritmo 4.1, introduz um fator de aleatoriedade no processo de geração das amostras. Essa aleatoriedade tem o papel de geração de variações em relação a amostra base. Entretanto, em bases com atributos

	Dia da semana Vizinhos(%)					Hora do dia Vizinhos(%)				
Base	1	2	3	4	5	1	2	3	4	5
2015	47,29	44,70	43,06	41,91	41,08	20,33	13,25	9,33	7,25	5,74
2017	37,30	35,60	34,60	33,80	33,18	13,66	8,31	6,09	5,17	4,73
2019	28,99	27,81	26,98	26,37	25,81	9,25	5,99	4,85	4,56	4,35
2020	24,94	23,73	23,02	22,59	22,08	7,13	5,15	4,70	4,69	4,78

Tabela 4.1 – Aderência dos Vizinhos aos Atributos Temporais.

temporais, essas alterações podem significar mudanças de escala temporal, que por sua vez pode introduzir ruídos na base.

A descrição do SMOTER deixa em aberto a forma como aquela escolha pode ser realizada. Dessa forma, para avaliar a estratégia de escolha mais adequada, caracterizou-se a aderência dos vizinhos oferecidos pelo KNN aos atributos temporais da amostra base, sendo k=5. Foram considerados os atributos dia da semana e horário, e os resultados estão na Figura 4.2 e Tabela 4.1.



(a) Dia da Semana. (b) Horário.

Figura 4.2 – Aderência dos Vizinhos aos Atributos Temporais.

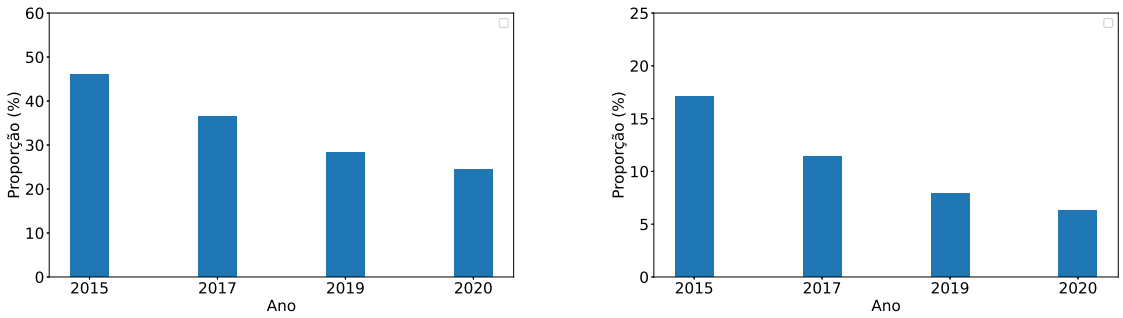
Para o atributo dia da semana, a conclusão geral é que a aderência dos vizinhos apresenta uma distribuição uniforme. Em outras palavras, entre os vizinhos oferecidos pelo KNN, em cada posição, existe uma aderência ao dia da semana da amostra base que é uniforme, mas a proporção que isso ocorre é numericamente mais alta na base de 2015, e é mais baixa na base de 2020.

Para o atributo horário, a conclusão geral é que existe uma maior aderência nos primeiros vizinhos, e na medida que a vizinhança se expande essa aderência é reduzida, e menos uniforme que aquela identificada anteriormente. Ao mesmo tempo, verifica-se que numericamente as proporções medidas são bem menores que aquelas do estudo

anterior.

Na segunda etapa desse estudo, um padrão de escolha dos vizinhos oferecidos pelo KNN foi definido e avaliado. Considerou-se um padrão com viés voltado para as posições com maior similaridade à mostra base. Dessa forma, a distribuição gerada produziu um padrão de escolha em que o primeiro vizinho foi escolhido 64% da vezes, o segundo vizinho foi escolhido 23%, o terceiro vizinho, 9%, o quarto vizinho 3%, e o quinto vizinho 1%.

As Figuras 4.3a e 4.3b mostram a aderência aos fatores temporais das amostras escolhidas em cada base. Observa-se que as proporções estão muito próximas dos valores definidos pela combinação dos dois primeiros vizinhos oferecidos pelo KNN.



(a) Dia da semana. (b) Horário.

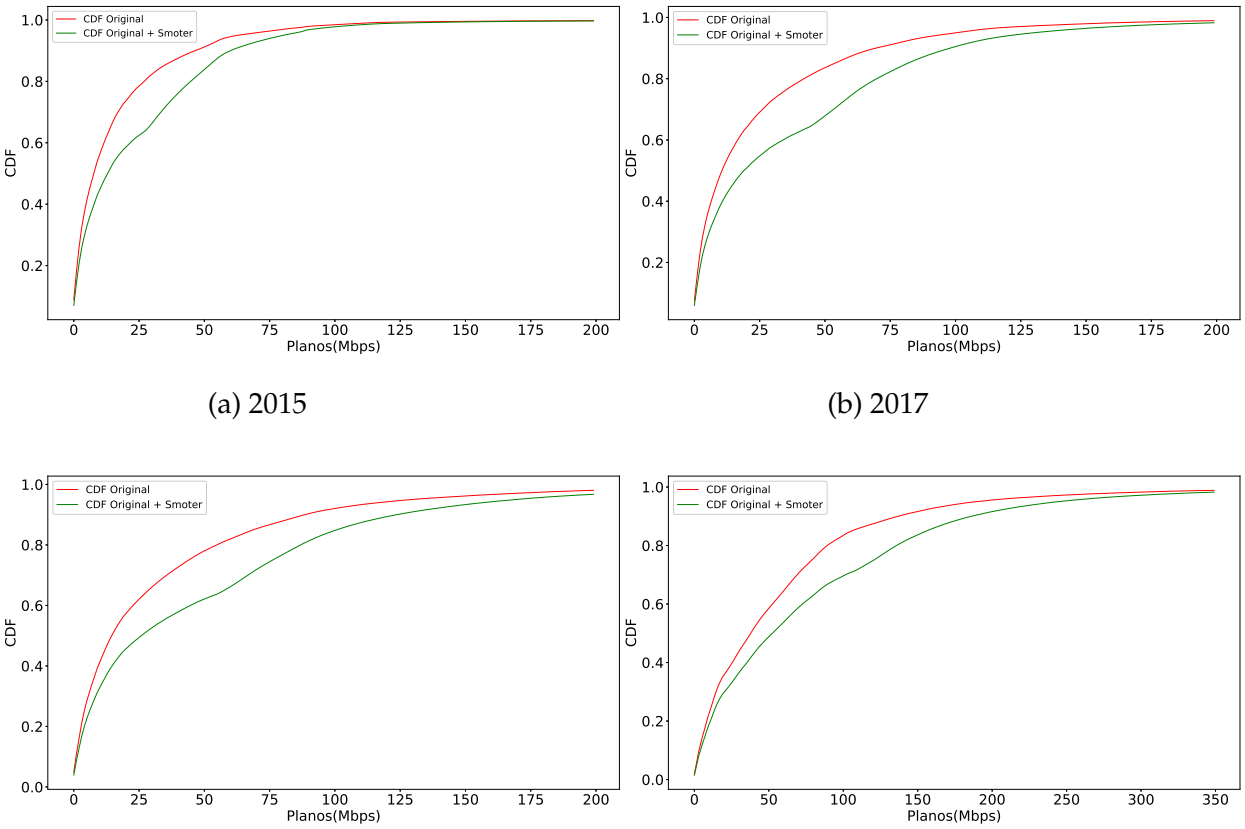
Figura 4.3 – Vizinhos escolhidos com atributos temporais da amostra base.

4.4 Resultados numéricos

Esta seção apresenta as análises feitas sobre as bases de dados Neubot, coletadas em quatro anos, após a aplicação do oversampling, por meio do método SMOTER, para tratar o desbalanceamento da base. As mudanças introduzidas nas séries e a qualidade dessas mudanças são avaliadas, especificamente a aderência aos atributos temporais da amostra base, nas suas diferentes escalas de tempo.

O primeiro estudo investigou o comportamento da Função Distribuição Acumulada - CDF (*Cumulative Distribution Function*) para os planos de dados identificados na base. A CDF foi obtida em dois momentos: antes e depois da aplicação do método SMOTER (Figura 4.4), com a geração de duas amostras para cada amostra presente no

grupo minoritário. Para todas as quatro bases avaliadas, dos anos de 2015, 2017, 2019 e 2020, a conclusão geral é que ocorre mudança na distribuição dos planos, com aumento do número de planos com maior vazão, portanto, o objetivo inicial do *oversampling* foi realizado.



(a) 2015 (b) 2017

(c) 2019 (d) 2020

Figura 4.4 – CDFs antes e após aplicação do SMOTER.

A mudança mais significativa ocorreu na distribuição dos planos da base Neubot de 2019. Os planos de maior vazão, gerados pelo SMOTER, aparecem com maior frequência, gerando uma modificação expressiva a curva. A racionalidade desse evento é que o método conseguiu gerar amostras com atributos de vazão muito próximo daquele identificado com sendo da classe minoritária. Ou seja, a escolha dos vizinhos pelo KNN apresentou significativa similaridade e a computação implementada foi suficientemente eficiente para determinar os valor da variável alvo.

Por outro lado, a menor influência foi identificada na base Neubot de 2015. Nesse sentido, a racionalidade anterior parece não ter aplicação no contexto da base

de Neubot 2015, ressaltando-se que a quantidade de sessões da base de 2015 é a menor entre todas as bases avaliadas. Além disso, o grau de desbalanceamento da base é inferior ao identificado nas outras bases.

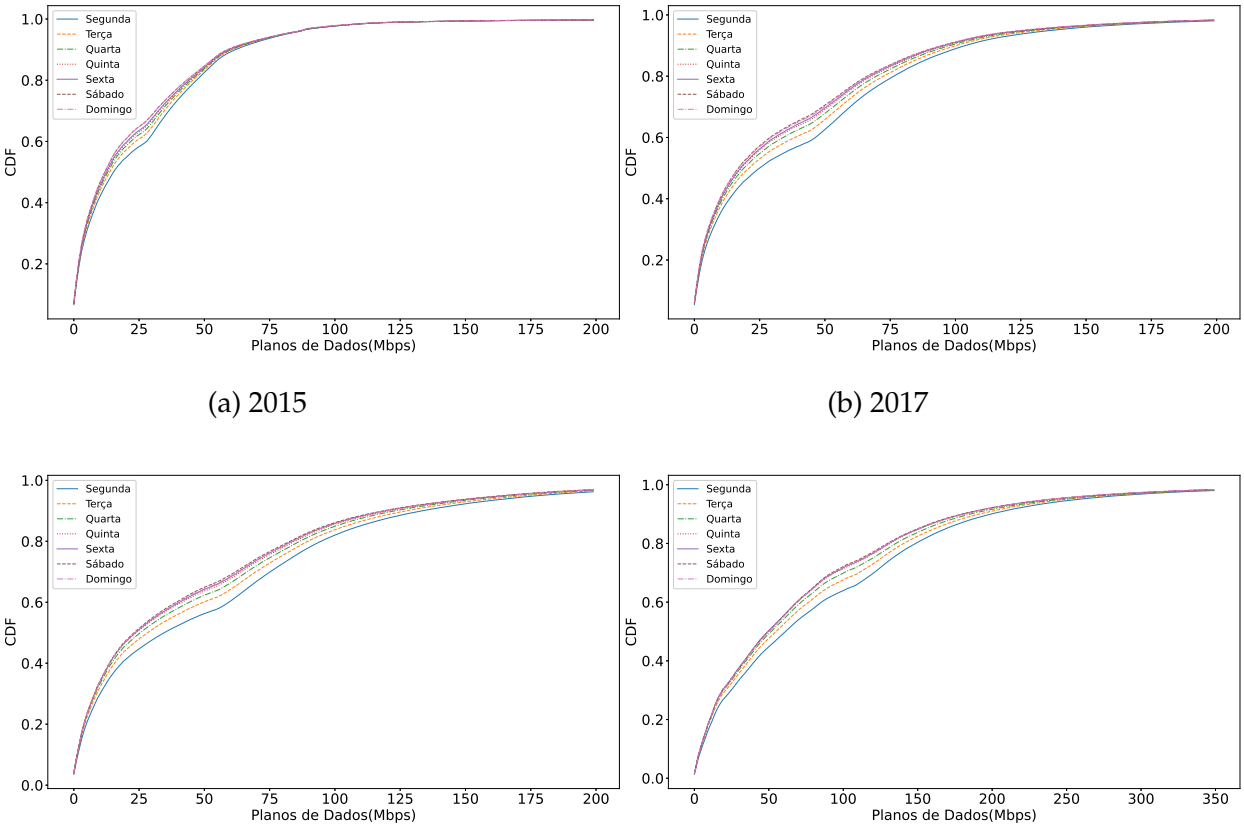
A conclusão geral que pode ser derivada desse estudo é que o SMOTER foi capaz de alterar a função de distribuição da base, ao introduzir o volume significativo de novas amostras que supostamente apresentam características similares àquelas identificadas na classe minoritária. O número maior de sessões com vazão similar àquelas identificadas anteriormente é destacado nos resultados apresentados. A questão que se impõe para esses resultados é se houve, nessas novas amostras, aderência aos fatores temporais da amostra base, isto é, a temporalidade associada as sessões base está representada nas amostras geradas?

O segundo estudo investigou a efetividade do método SMOTER em gerar amostras com características temporais adequadas. Em outras palavras, verificou se as escalas temporais presentes na série original foram respeitas no processo de oversampling, i.e., se as alterações naturalmente produzidas pelo oversampling foram nas distribuições esperadas. Para a realização desse estudo, as sessões foram agrupadas por dia da semana e desse agrupamento gerou-se a CDF dos planos de dados (ver a Figura 4.5).

A conclusão geral é que a alteração identificada anteriormente não pode ser atribuída a um único dia da semana, ou seja, as amostras geradas foram distribuídas por diversos dias da semana. A CDF calculada segue o mesmo padrão após a adição das novas amostras, com maior presença de amostras na curva de segunda-feira. Entretanto, essa presença não é discrepante com relação aos demais dias da semana.

O terceiro estudo avaliou os impactos das escolhas aleatórias do SMOTER na segunda escala temporal, i.e. faixas de horário de um dia. As sessões de acesso foram agrupadas conforme a hora de ocorrência, nas faixas horárias 0-6, 7-12, 13-18, e 19-23. Desse agrupamento calculou-se a CDF de cada agrupamento, para os quatro anos da base Neubot. A Figura 4.6 mostra o resultado dessa avaliação.

As conclusão geral é que a faixa horária de 0-6h é a que apresentou a maior alteração, quando comparado com a distribuição da base original. Esse comportamento é destacado na base do ano de 2020. A explicação para esse comportamento está na



(a) 2015 (b) 2017

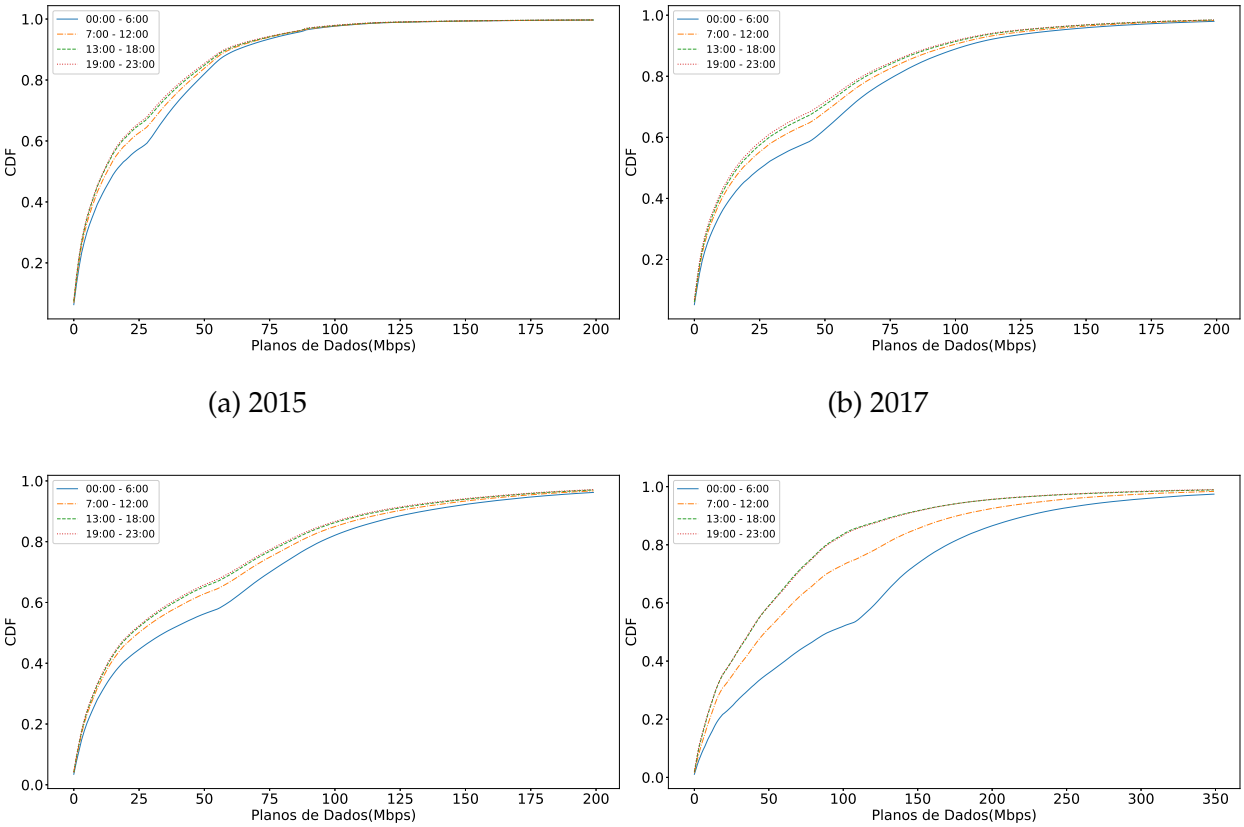
(c) 2019 (d) 2020

Figura 4.5 – CDFs de dias da semana após aplicação do método SMOTER

combinação de dois aspectos: baixo número de medições na faixa de horário de 0-6h, em todos os anos da base Neubot, e a quantidade de amostras sintéticas geradas nessa faixa. Em outras palavras, embora tenha gerado amostras em todas as faixas de horário, o *oversampling* gerou muito mais amostras na faixa horária de 0-6h, produzindo um afastamento da CDF calculada nesse horário em relação ao padrão global em que essa faixa pouco contribuiu para estabelecer.

Para corroborar com a conclusão anterior, realizou-se levantamento da distribuição das sessões de medição no conjunto minoritário antes e depois da aplicação do SMOTER para a base de 2020. A escolha dessa base é justificada pelo maior distanciamento entre CDFs, conforme mostrado anteriormente. As sessões foram agrupadas em duas escalas de tempo: dia da semana e horário, ver Figura 4.7.

É possível observar que os dados do conjunto minoritário apresentam uma proporção uniforme entre cada dia da semana e horas do dia, mas o mesmo não se



(a) 2015

(b) 2017

(c) 2019

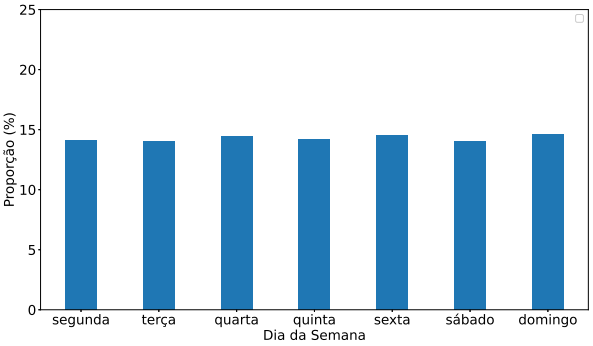
(d) 2020

Figura 4.6 – CDFs de horários após aplicação do método SMOTER.

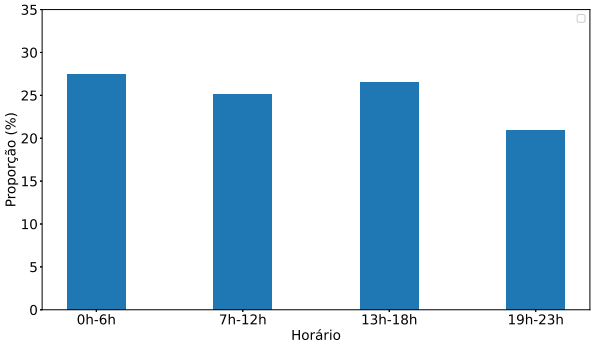
observa no conjunto de amostras geradas. Nesse conjunto, a proporção de amostras criadas é maior na segunda feira e nas horas que forma a faixa de 0h a 6h. Enquanto a segunda feira tinha apenas 15% das amostras da classe minoritária, após o oversampling esse valor chegou a 23%. Outra mudança significativa é identificada na faixa de horário de 0-6h que passou a responder por 39% das amostras após o oversampling, quando originalmente respondia por apenas 27%. Esse resultado resume uma mudança de padrão identificada na geração das amostras, que modifica significativamente a função de distribuição da classe minoritária.

O resultado anterior demandou avaliação sobre a origem dessa tendência. Para isso, foi avaliada a similaridade entre as amostras base e gerada, em especial os atributos que definem a escala temporal, i.e., dia da semana, faixas de horário e horas individuais.

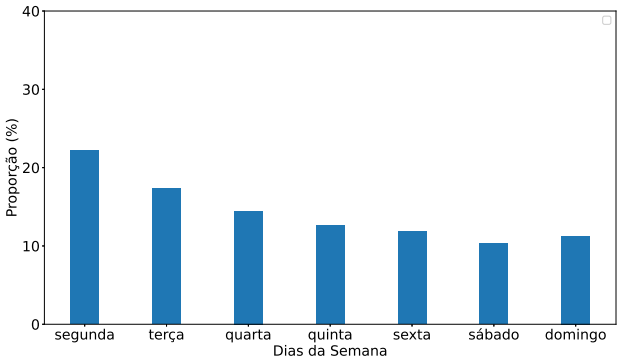
A Figura 4.8 apresenta as proporções observadas nas amostras geradas que estão dentro e fora da faixa de horário da amostra base. Uma observação a ser destacada é



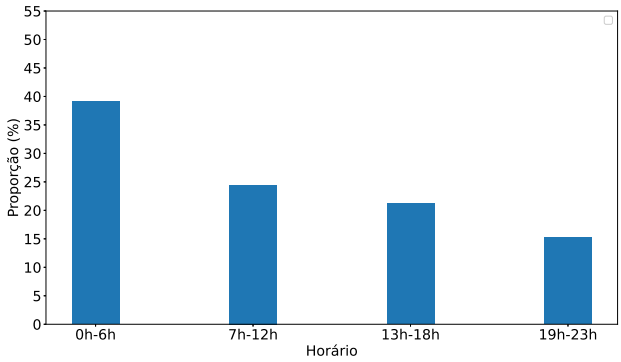
(a) Antes: Dia da Semana



(b) Antes: Faixa horária



(c) Depois: Dia da Semana



(d) Depois: Faixa horária

Figura 4.7 – Impactos do Oversampling na Distribuição das Amostras por Componente Temporal (Base de 2020).

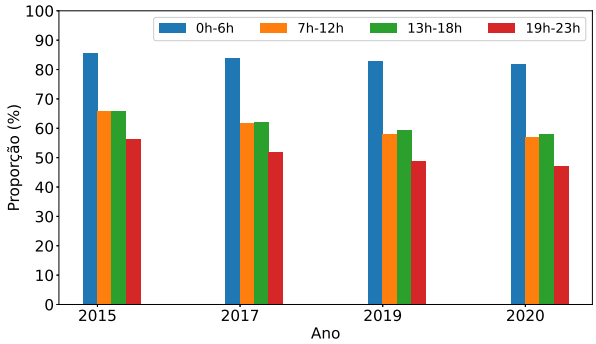


Figura 4.8 – Aderência ao atributo temporal faixas de horário da amostra base.

que a faixa de horário 0-6h apresenta uma proporção alta de aderência, acima de 80% em todos os anos avaliados. Por outro lado, a faixa de horário de 18-23h apresenta a menor taxa de aderência, tendo alcançado no máximo 56%.

A Tabela 4.2 apresenta as proporções em detalhes em cada faixa de horário. As faixas de horário de 7-12h e 13-17h apresentaram valores intermediários. A razão para o comportamento identificado é o fato de que as faixas horárias com maior aderência

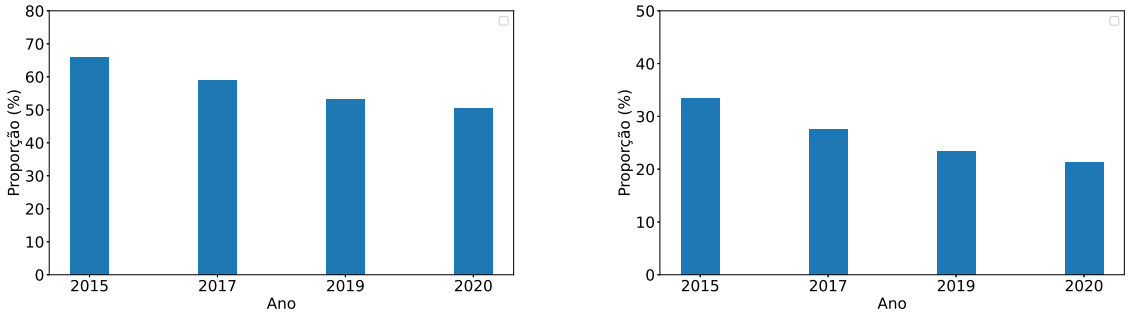
		Faixa Horária	
Ano	Faixa	Mesma(%)	Outra(%)
2015	0-6	85,51	14,49
	7-12	65,71	34,29
	13-17	65,98	34,02
	18-23	56,27	43,73
2017	0-6	84,01	15,99
	7-12	61,65	38,35
	13-17	62,20	37,80
	18-23	51,77	48,23
2019	0-6	82,82	17,18
	7-12	58,12	41,88
	13-17	59,24	40,76
	18-23	48,86	51,14
2020	0-6	81,96	18,04
	7-12	57,00	43,00
	13-17	58,03	41,97
	18-23	47,15	52,85

Tabela 4.2 – Aderência ao atributo temporal faixas de horário da amostra base.

apresentarem características bem específicas, por exemplo, menor flutuação de vazão ao longo da sessão. Isso torna as escolhas do KNN mais restritivas, ou seja, as sessões escolhidas têm características muito similares, que em grande proporção estão na mesma faixa de horário ou estão em faixas de horário bem próximas. Nesse último caso, a definição dos valores dos atributos deve tender para os valores da amostra base, que no caso do SMOTER é definido a partir de uma distribuição uniforme aleatória.

Em resumo, esse resultado ressalta as dificuldades que o método baseado em vizinhança tem de produzir amostras com aspectos temporais restritivos, ao não estabelecer claramente os limites das classes que compõem a base, a partir de um entendimento de que é preciso observar os múltiplos fatores que determinam as classes.

Outra avaliação verificou a aderência dos atributos temporais dia da semana e hora do dia a amostra base. Os números mostram uma aderência em torno de 50% para o atributo dia da semana e uma aderência em torno de 20% para o atributo hora do dia. Essa aderência do atributo hora do dia pode ser explicado pela composição da base e pela forma de geração das amostras. As amostras escolhidas pelo KNN precisariam ser na mesma hora, o que pela composição da base é possível de acontecer, já que há registros de medições na mesma hora. Entretanto, a aleatoriedade para definir os valores



(a) dia da semana (b) horário

Figura 4.9 – Aderência aos atributos temporais Dia da Semana e Hora do dia da amostra base.

	Dia da semana		Hora do dia	
Base	Mesmo(%)	Outro(%)	Mesma(%)	Outra(%)
2015	65,79	34,21	33,46	66,54
2017	58,99	41,01	27,62	72,38
2019	53,31	46,69	23,34	76,66
2020	50,58	49,42	21,35	78,65

Tabela 4.3 – Aderência aos atributos temporais Dia da Semana e Hora do dia da amostra base.

dos atributos numéricos também precisaria ser próximo de zero para que a amostra tivesse o mesmo horário. O uso de uma distribuição uniforme, pelo SMOTER, impede que haja um viés forte que produziria uma aderência ainda maior, ver linhas 10 e 11 do Algoritmo 4.1.

4.5 Connsiderações Finais

Este capítulo apresentou um estudo da adequação da abordagem baseada em vizinhança para tratar o desbalanceamento presente na base Neubot. O Algoritmo SMOTER foi apresentado e avaliado. A escolha das sessões da vizinhança, que serão utilizadas para gerar as novas amostras, foi investigada.

Além disso, analisou-se a aderência dos atributos temporais nas diferentes escala de tempo. A conclusão geral é que para escalas de tempo menores a aderência é reduzida. Em escalas de tempo mais amplas, o que inclui as faixas de horário, a combinação de geração aleatória e similaridade é eficiente. Adicionalmente, concluiu-se que as

mudanças significativas nas funções de distribuição dos planos introduziu ruído na base quando se verificou atributos temporais, e.g., dia da semana e horário, das amostras geradas.

5

OVERSAMPLING BASEADO EM REGIÃO DE INTERESSE

Um método que produza amostras sintéticas com melhor qualidade corrobora com a hipótese de que as bases geradas têm o potencial para melhora de qualidade dos modelos criados a partir de sua utilização. Neste capítulo apresenta-se um método para geração de amostras que é baseado no conceito de região de interesse. A racionalidade dessa abordagem é a exploração das regiões estabelecidas no plano definido a partir de duas características (features) da base. A Seção 5.1 apresenta o conceito de região de interesse e mostra por que as abordagens usando vizinhança falham em tratar o desbalanceando em bases com relações não lineares entre seus atributos. As Seções 5.2 e 5.3 mostram as abordagens usadas para tratar o desbalanceamento nessas bases. Os Algoritmos RBO e RBO-QS são apresentados e discutidos. Finalmente, na Seção 5.4, a avaliação daqueles algoritmos é apresentada.

5.1 Regiões de Interesse

Um kernel é uma função que quantifica a similaridade entre duas observações dadas. Métodos de *oversampling* baseados em vizinhança, e.g., SMOTE e derivados, fazem uso de kernels que são definidas a partir de medidas de distância entre os elementos do conjunto minoritário. Ao mesmo tempo que essa abordagem tem sido amplamente utilizada para a geração de novas amostras, devido a sua simplicidade e relativa efeti-

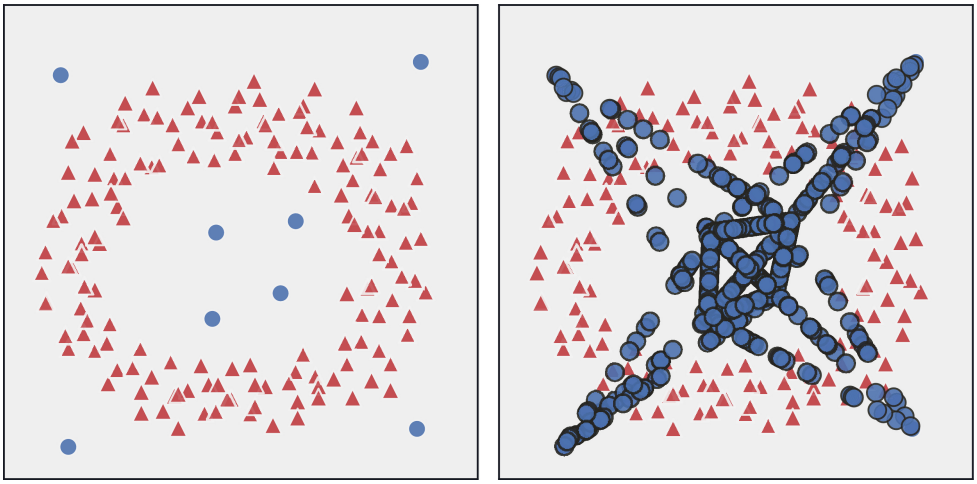


Figura 5.1 – Inadequação de métodos baseado em Vizinhança. Painel à esquerda caracteriza base com classe minoritária formada por conjuntos disjuntos. Painel à direita caracteriza o efeito da relação linear usada por métodos baseado em vizinhança. Fonte: (KOZIARSKI; KRAWCZYK; WOŹNIAK, 2019)

vidade, existem limitações para o seu uso. Especificamente, ao ignorar a posição das amostras da classe minoritária, envolvidas na geração, é possível que as novas amostras apresentem características da classe majoritária.

A Figura 5.1 mostra um exemplo em que as amostras formam duas classes, e as suas regiões de ocorrência inviabilizam caracterizações lineares. A expansão do conjunto minoritário em direções inadequadas, definida pela relação de similaridade entre as amostras, pode gerar amostras que alteram a distribuição da classe majoritária, com efeito limitado para a classe minoritária.

Observa-se que a direção tomada pelas novas amostras avança sobre a região caracterizada inicialmente como sendo definida pela classe majoritária. Ao mesmo tempo que, quantitativamente os procedimentos adotados podem sugerir correção, existe o potencial de alteração na função de distribuição da classe majoritária.

Para lidar com a limitação de métodos baseados em vizinhança, o método RBO (KOZIARSKI; KRAWCZYK; WOŹNIAK, 2019) apresenta a definição de regiões de interesse reais com base na estimativa da distribuição de desbalanceamento da classe minoritária usando função de base radial. Novas amostras são geradas a partir da expansão dessas regiões de interesse.

5.2 O RBO e suas Limitações

Enquanto que métodos baseados em vizinhança, em especial o SMOTE e suas variações (TORGO et al., 2013), geram regiões binárias entre as instâncias da classe minoritária, considerando o subconjunto de amostras dessa classe, o método RBO precisa olhar tanto para instâncias da classe majoritária quanto minoritária caracterizando assim de forma mais precisa o conjunto de dados. Para gerar as regiões reais de interesse, o método RBO calcula o potencial de cada ponto do conjunto pertencer à classe majoritária ou minoritária. Esse cálculo é feito aplicando uma função gaussiana de base radial com polaridade definida pela classe, em todo conjunto de treino.

Para assegurar maior precisão, o método RBO apresenta custo computacional definido pelo tamanho da base de dados em que é aplicado. Quando se trata de grandes bases de dados, como é o caso do projeto Neubot, o custo computacional verificado é proibitivo. Especificamente, o custo computacional está relacionado ao cálculo da função radial, que é realizado para cada amostra da classe minoritária, em relação ao todo do conjunto de dados. Também para cada iteração de expansão realizada em uma amostra base para gerar uma nova. O Algoritmo 5.1 mostra o pseudo-código do RBO para geração das amostras sintéticas para o conjunto minoritário.

O método RBO resolve o desbalanceamento igualando o número de amostras do grupo minoritário ao do grupo majoritário, essa condição é avaliada na linha 3 do Algoritmo 5.1. Para gerar cada amostra, um ponto da classe minoritária é tomado como ponto inicial, definido na linha 4. Este ponto pode ser modificado por no máximo uma quantidade de vezes, definido pelo número de iterações na linha 5 ou ter uma parada antecipada definida na linha 6.

A cada iteração, a direção em que o ponto será transladado, ou seja qual característica será modificada e o sinal dessa mudança são escolhidos nas linhas 7 e 8 do algoritmo, respectivamente. O novo ponto transladado é calculado na linha 9 e o seu valor de função radial é calculado e avaliado na linha 10, se tiver valor absoluto menor que o ponto inicial, então o ponto inicial é modificado, tomando o valor do ponto transladado, senão é descartado e a operação é repetida com o ponto modificado ou não.

Data: Coleção de objetos da classe Majoritária K e coleção de objetos da classe minoritária κ
Result: Coleção de objetos sintético minoritário S
Parameter: Raio da função radial base γ, tamanho do passo de otimização ($stepsize$), número de iterações por amostra sintética, probabilidade de parada prematura p;
inicializa S = {} ;
while $|\kappa| + |S| < |K|$ **do**
    $point \leftarrow$ objeto escolhido aleatoriamente de κ;
    for $i \leftarrow 1$ **to** $iteractions$ **do**
        interromper For *com probabilidade* p ;
        $direction \leftarrow$ *escolher aleatória usando os vetores base no espaço Euclidiano com n-dimensões, onde* n *sendo o número de características* ;
        $sign \leftarrow$ *escolha aleatória no intervalo {-1,1}*;
        $translated \leftarrow$ *point* $+$ *direction* $\times$ *sign* $\times$ $stepsize$;
        if $|\Theta(translated, K, \kappa, \lambda)| < |\Theta(point, K, \kappa, \gamma)|$ **then**
            $point \leftarrow translated$;
        end
    end
    incluir $point$ *em* S;
end

Algorithm 5.1: Oversampling baseado em região de interesse.

5.3 Exploração de Região baseado Quadrante

O método RBO descrito anteriormente tem um custo computacional inviável quando a base tem grande dimensão e um volume grande de instâncias. Para lidar com a inviabilidade do método nesse contexto, duas abordagens foram utilizadas conjuntamente, são elas: i) amostragem da base de dados; e ii) modificação do método de exploração das regiões. Na primeira abordagem, o cálculo feito com o kernel é limitado a um subconjunto de amostras, escolhida da base original, e o resultado final é obtido com a união das amostras sintéticas obtidas de cada subconjunto. Esses subconjuntos respeitam a proporção de ocorrências entre as classes minoritárias e minoritárias e, dentro de cada classe, a proporção de ocorrências de amostras com características específica também é observada. Em outras palavras, os subconjuntos utilizados são estratificações da base original, que respeitam características chave presentes nessa base.

Na base Neubot, as características como dia da semana e faixas de horário foram consideradas no processo de escolha das amostras que formaram os subconjuntos. Assim, as amostras que formam os subconjuntos apresentam, nas mencionadas carac-

terísticas, proporções similares àquelas identificadas na base original. A amostragem torna possível a aplicação do método sobre conjuntos de dados maiores, à medida que o cálculo da kernel é feito sobre representações parciais do conjunto de dados, embora custo da operação de geração não seja modificado.

A segunda abordagem consistiu em modificar a forma como ocorre a exploração das regiões de interesse pelo método do RBO original. Originalmente, o ponto inicial é transladado por várias iterações. A cada iteração, uma característica é modificada de acordo com o $stepsize$ e a kernel do ponto resultante é então calculada e avaliada, se aprovada, então a translação é aplicada ao ponto modificando-o, senão é descartado e segue-se para a próxima iteração.

A modificação proposta tem como princípio a redução do $stepsize$ a partir de um valor inicial. Esse valor é usado para calcular o ponto central de quatro quadrantes formados pelos eixos de duas características escolhidas aleatoriamente a cada iteração, os pontos dos quadrantes são calculados de forma paralela. Destes, escolhe-se o ponto com menor valor dado pela kernel, modificando o ponto de origem e este torna-se o ponto de partida para a geração dos pontos centrais dos quadrantes na próxima iteração, reduzindo o $stepsize$ pela metade até a condição de parada seja alcançada.

A Figura 5.2 exemplifica como a translação ocorre no algoritmo proposto. Em cada iteração, duas características (x_i, x_j) são escolhidas para definir a região de interesse. Um valor inicial para o d é definido pelo $stepsize$ para limitar o espaço de translação do ponto inicial P, a partir do qual será feita a translação em cada quadrante, gerando seus respectivos pontos centrais P_1, P_2, P_3 e P_4. Para a iteração seguinte, P é atualizado com o ponto gerado com menor valor de função e o d é reduzido pela metade. Isso se repete até que se obtenha o d mínimo e o ponto resultante é adicionado ao conjunto de pontos gerados.

O ponto central de cada quadrante P_i dessa região é avaliado em relação a sua similaridade com o ponto central P, e aquele que apresentar maior similaridade será usado como ponto central na próxima iteração (Figura 5.2). A região é então explorada a partir de sucessivas reduções do seu raio, $d = d/2$, com decisões gulosas a cerca da direção de exploração. O Algoritmo 5.1 descreve o pseudo-código da versão proposta.

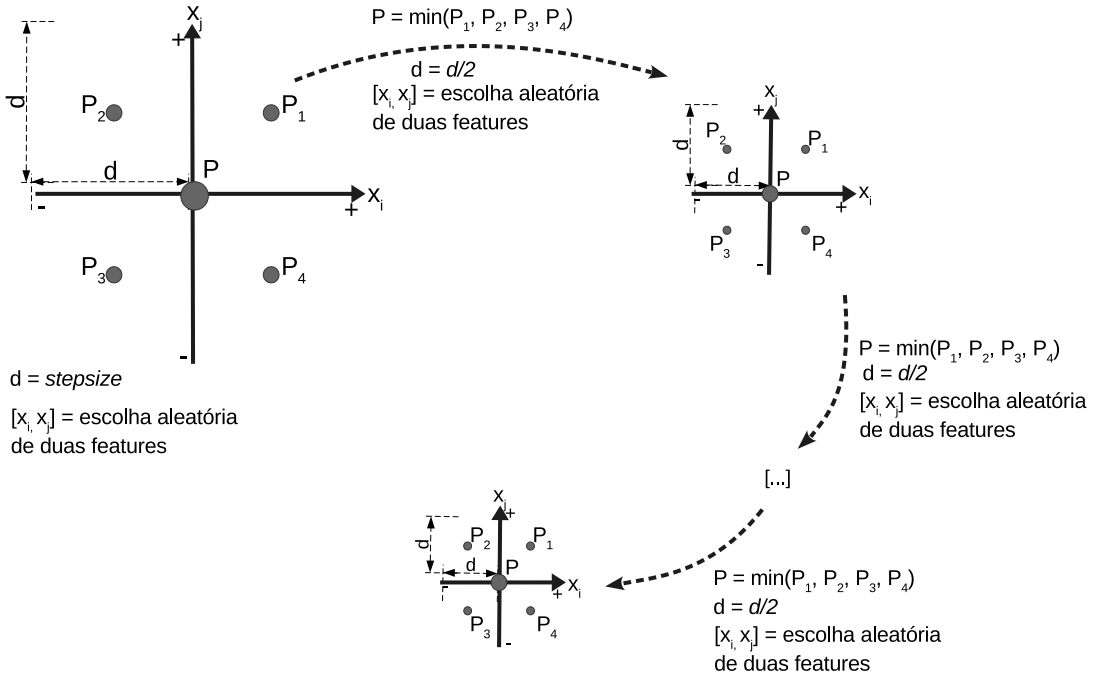


Figura 5.2 – RBO-QS: exploração da região de interesse. Fonte: de autoria própria

Assim como o método original, o desbalanceamento é resolvido igualando-se o número de amostras do grupo minoritário e majoritário, como definida na linha 3 do Algoritmo 5.1. Porém, a translação do ponto inicial, definido na linha 4, agora é baseada em consecutivas reduções do $stepsize$, até que um valor limite seja atingido, condição da linha 5. A direção da translação tem dois componentes, ou seja, duas características escolhidas aleatoriamente na linha 6. Essas características serão modificadas na translação realizada para cada um dos quadrantes definidos na linha 7.

Para cada quadrante, a função de *translate* é chamada e executada de forma paralela, como na linha 10. A função então faz a translação do ponto passado para um ponto central do quadrante considerando as características e retorna uma tupla contendo o ponto e o valor de sua função radial, como mostra o Algoritmo 5.2.

Em seguida, na linha 13, escolhe-se o ponto com o menor valor de função entre os pontos gerados para cada quadrante e então o ponto inicial é atualizado com o ponto transladado e passa a ser o ponto de partida para a próxima iteração, definido na linha 14 e o $stepsize$ é diminuído pela metade. Ao final, o ponto é adicionado ao conjunto de pontos gerados na linha 14 e este é retornado ao término do algoritmo.

Data: Coleção de objetos da classe Majoritária K e coleção de objetos da classe minoritária κ
Result: Coleção de objetos sintético minoritário S
Parameter: Raio da função radial base γ, tamanho do passo de otimização ($stepsize$), número de iterações por amostra sintética, probabilidade de parada prematura p;
inicializa S = {} ;
while $|\kappa| + |S| < |K|$ **do**
  $point \leftarrow$ objeto escolhido aleatoriamente de κ;
  while $stepsize > c$ **do**
    $direction \leftarrow$ *escolha aleatória de um par de características* $\{x_1, x_2\}$ *usando os vetores base no espaço Euclidiano com n-dimensões, onde* n *sendo o número de características* ;
    $sign \leftarrow$ *{{1,-1}, {-1,1}, {1,1}, {-1,-1}}* ;
    inicializa P = {} ;
    for $i \leftarrow 1$ **to** $lenght(sign)$ **do**
      $(translated, score_t) \leftarrow translate(point, sign[i], stepsize, \gamma, direction)$;
      incluir $(translated, score_t)$ *em* P;
    end
    $min_q \leftarrow min(R)$;
    $point \leftarrow min_q.translated$;
    $stepsize \leftarrow stepsize/2$;
  end
  incluir $point$ *em* S;
end

Algorithm 5.1: Oversampling baseado em área de interesse - RBO-QS.

Input: Ponto base $point$, direção do quadrante $\{sign_1, sign_2\}$, tamanho do passo de otimização ($stepsize$), Raio da função radial base γ, par de características $\{x_1, x_2\}$
Output: Tupla com ponto transladado e o valor de sua função radial $(translated, score_t)$
Data: Coleção de objetos da classe Majoritária K e coleção de objetos da classe minoritária κ
Function `translate(`F`)`**:**
  $translated \leftarrow \{0\}$;
  $translated[x_1, x_2] \leftarrow translated[x_1, x_2]$ + $\{sign_1, sign_2\}$ * $stepsize/2$;
  $translated \leftarrow point$ + $translated$;
  $score_t \leftarrow \Theta(translated, K, \kappa, \lambda)$
  return $(translated, score_t)$;
End Function

Algorithm 5.2: Função de translação do ponto em um quadrante

5.4 Resultados Numéricos

Esta sessão apresenta os resultados obtidos com a exploração de regiões de interesse. O volume de dados e dimensionalidade da base tornam a aplicação do método RBO computacionalmente inviável. Dessa forma, um protocolo de estratificação foi proposto e utilizado para obter conjuntos parciais de amostras que posteriormente foram submetidos ao método. Primeiramente, avaliou-se a estratificação realizada e posteriormente mostra-se o potencial de geração do método RBO. Finalmente, avalia-se a modificação proposta para o método RBO que o torna computacionalmente viável para atuar sobre grandes bases de dados.

5.4.1 O Protocolo de Estratificação

Esta seção apresenta o protocolo de estratificação utilizado para produzir uma base viável para a aplicação do RBO, devido ao seu alto custo computacional. Seja S a base dada considerada que foi dividida em duas: treinamento e teste. A base de treinamento, S_{Tr}, representa 70% das amostras em S, enquanto que na base de teste, S_{Ts} tem-se o restante da amostra, 30%.

A estratificação da base S_{Tr} ocorreu como segue: i) retirou-se 0.1% de amostras, de forma aleatórias; ii) manteve-se a proporção das amostras da classe minoritária e majoritária; e iii) manteve-se a proporção de características temporais, i.e., dias da semana e horário. Em resumo, seja M o tamanho de fração retirada de S_{Tr}. Nesse quantitativo foi mantida a proporção de amostras que formam a classe majoritária e minoritária na base S_{Tr}. Adicionalmente, a fração de amostras registradas em um dia da semana, em uma faixa de horário específica, também está representada na estratificação. A Figura 5.3 demonstra o protocolo de estratificação no qual, a partir do conjunto S_{Tr}, gera-se n subconjuntos, cada um correspondendo à fração M de S_{Tr} e mantendo a proporção entre o grupo majoritário e minoritário por meio de escolha aleatória das amostras.

A Tabela 5.1 mostra os tamanhos das bases original e estratificada, agrupadas por classes. Percebe-se uma redução drástica no tamanho da base, que ocorreu devido

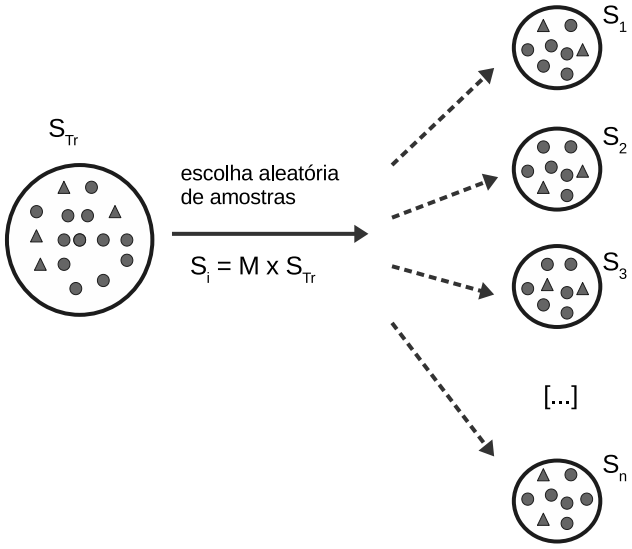


Figura 5.3 – Protocolo de Estratificação. Fonte: de autoria própria

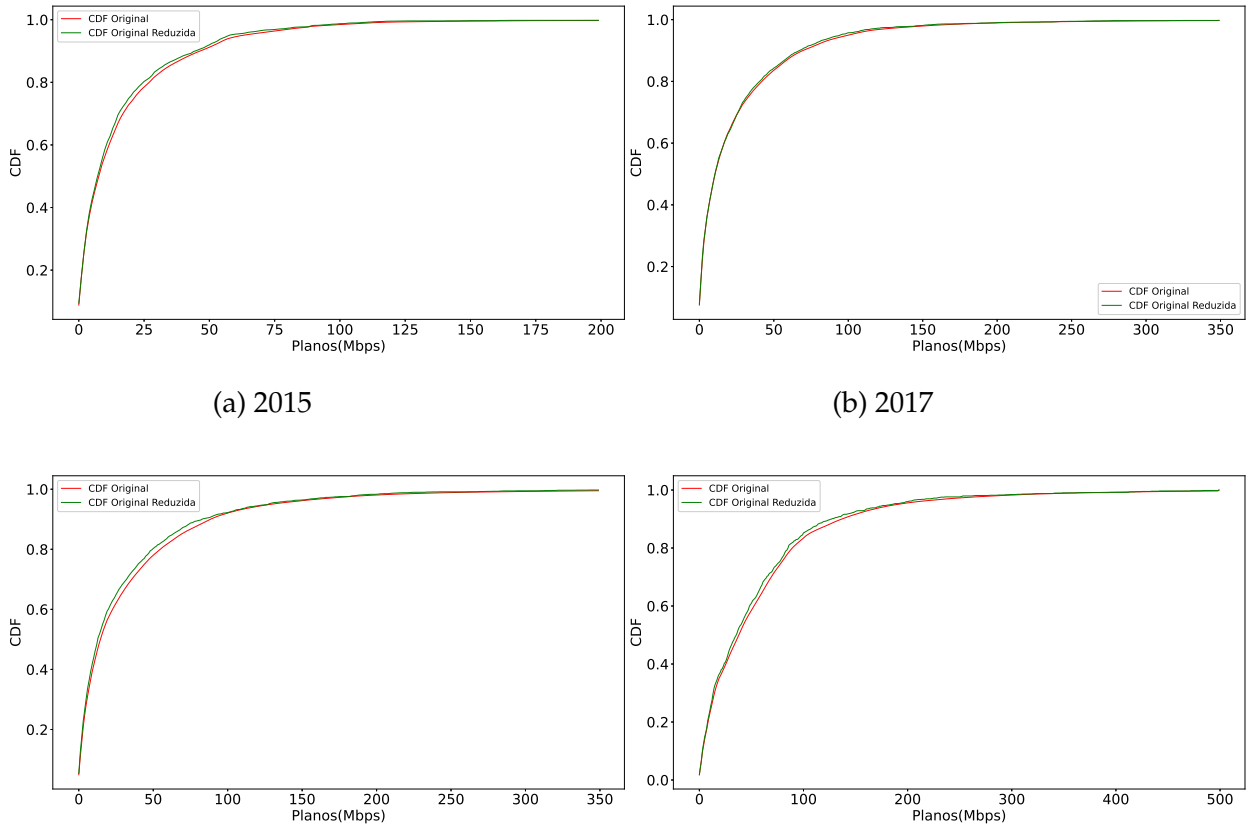
as demandas de processamento que o RBO apresenta. Além disso, o RBO demanda que a base apresente uma classificação das amostras, como raras e não raras. Para isso foi utilizada uma função de relevância baseada no algoritmo *piecewise cubic Hermite interpolation polynomials (pchip)*, descrita em (MONIZ; BRANCO; TORGO, 2017), que usa estatísticas de boxplot e mapeia os valores da variável alvo em uma escala de $[0, 1]$ de relevância. O algoritmo RBO demanda essa classificação para poder processar a base.

Ano		Classe Majoritária	Classe Minoritária	Total
2015	Original	3287183	739940	4027123
	Reduzida	3288	739	4027
2017	Original	2911452	666649	3578101
	Reduzida	2912	666	3578
2019	Original	2185643	488742	2674385
	Reduzida	2186	488	2674
2020	Original	1069113	177278	1246391
	Reduzida	1056	163	1219

Tabela 5.1 – Tamanhos de bases de dados originais e reduzidas.

Para avaliar o resultado da estratificação, na distribuição dos planos de dados, calculou-se suas CDFs, ver Figura 5.4. A conclusão geral é que o padrão das CDFs,

calculadas para a base original, é pouco modificado com relação ao padrão observado na base estratificada.



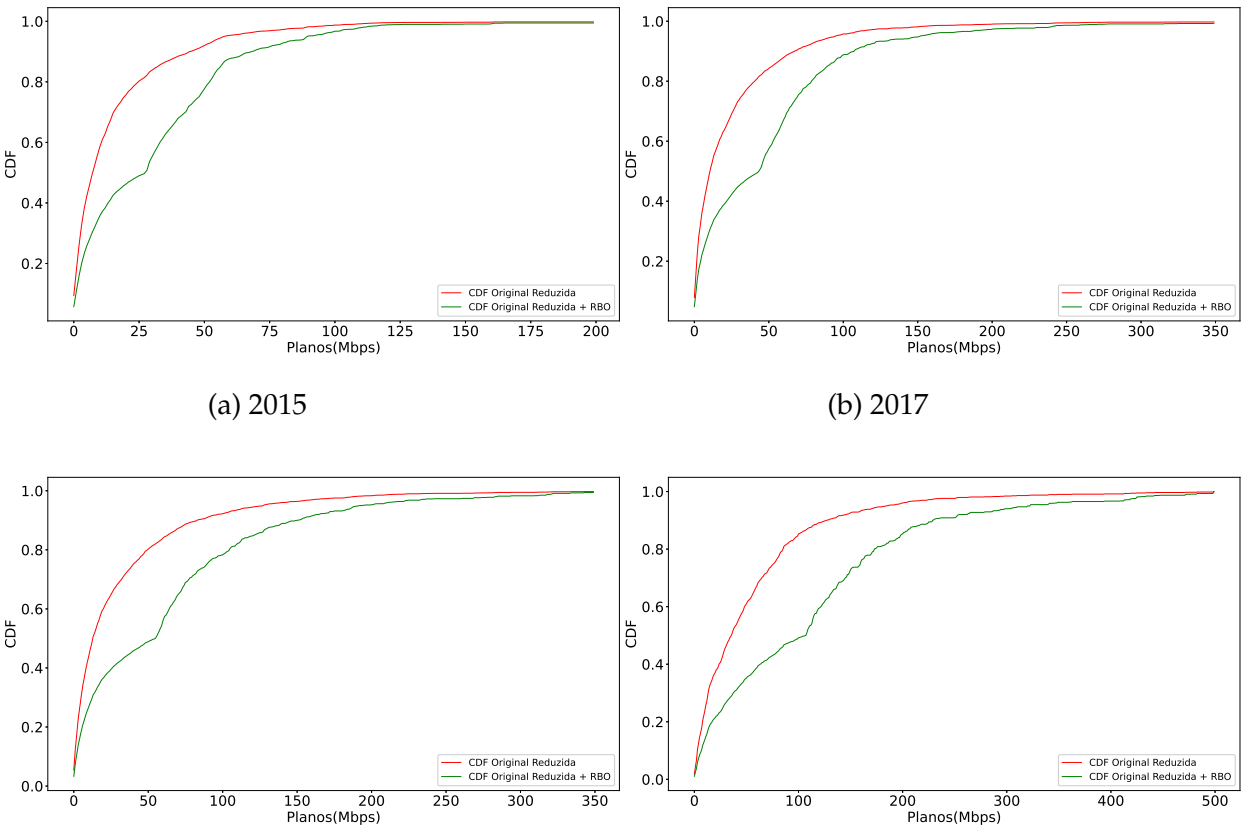
(a) 2015 (b) 2017

(c) 2019 (d) 2020

Figura 5.4 – CDF da base de dados original e reduzida.

5.4.2 O potencial de geração do RBO

Esta seção apresenta o estudo sobre o potencial de geração do método RBO aplicado a conjuntos de dados reduzidos obtidos pelo protocolo de estratificação. A Figura 5.5 mostra a diferença entre as CDFs quanto aos planos de dados dos conjuntos reduzidos antes e depois da geração de amostras pelo método RBO. Observa-se que para cada conjunto a CDF é modificada em maior ou menor grau. O conjunto de 2019 apresentou maior grau de mudança em relação ao planos de dados de valores mais altos. Deve ser levado em conta a seleção aleatória de planos de diferentes valores dentro do conjunto de classe majoritária e minoritária.



(a) 2015

(b) 2017

(c) 2019

(d) 2020

Figura 5.5 – CDF da base de dados reduzida e com *oversample* do RBO.

Analisando-se a similaridade entre amostras bases e amostras geradas a partir delas em relação a características de dia da semana e hora do dia, observa-se pela Tabela 5.2 um alto grau de similaridade, bem superior ao observado quando o método SMOTER é aplicado, embora deva-se considerar que este foi aplicado sobre a base de treino completa. Enquanto no caso do SMOTER a maior aderência observada para dia da semana e hora do dia é de $65,79\%$ e $33,46\%$ respectivamente para o ano de 2015, na Tabela 5.2, a menor aderência apresentada para essas mesmas características é de $88,58\%$ e $88,69\%$ respectivamente para o ano de 2020.

Ano	Dia da Semana	Hora
2015	90,59	90,12
2017	89,60	89,42
2019	91,39	90,10
2020	88,58	88,69

Tabela 5.2 – Porcentagem de amostras sintéticas com característica igual à amostra base no conjunto reduzido.

Considerando apenas a hora da amostra base e sua respectiva amostra sintética já observa-se um alto grau de similaridade. Quando analisa-se a similaridade entre faixa de horário da amostra base e da gerada, observa-se um grau de similaridade ainda maior, como mostrado na Tabela 5.3 a seguir.

Ano	Faixa de Horário	Dentro da Faixa de Horário	Fora da Faixa de Horário
2015	0h – 6h	100	0
	7h – 12h	98,74	1,26
	13h – 18h	98,12	1,88
	19h – 23h	96,50	3,50
2017	0h – 6h	100	0
	7h – 12h	98,13	1,87
	13h – 18h	97,69	2,31
	19h – 23h	98,08	1,92
2019	0h – 6h	100	0
	7h – 12h	98,80	1,20
	13h – 18h	98,89	1,11
	19h – 23h	99,40	0,60
2020	0h – 6h	100	0
	7h – 12h	99,60	0,04
	13h – 18h	99,12	0,08
	19h – 23h	98,35	1,65

Tabela 5.3 – Porcentagem de amostras sintéticas com mesma faixa de horário da amostra base no conjunto reduzido.

5.4.3 Analisando o RBO-QS

Embora o uso de conjuntos menores tenha tornado o custo menor, a repetição por várias iterações do método RBO ainda apresenta um alto custo quanto ao tempo de

Ano	Faixa de Horário	Dentro da Faixa de Horário	Fora de Faixa de Horário
2019	0h – 6h	97,64	2,36
	7h – 12h	97,34	2,66
	13h – 18h	97,40	2,60
	19h – 23h	97,09	2,91
2020	0h – 6h	97,69	2,31
	7h – 12h	97,34	2,66
	13h – 18h	97,30	2,70
	19h – 23h	97,12	2,88

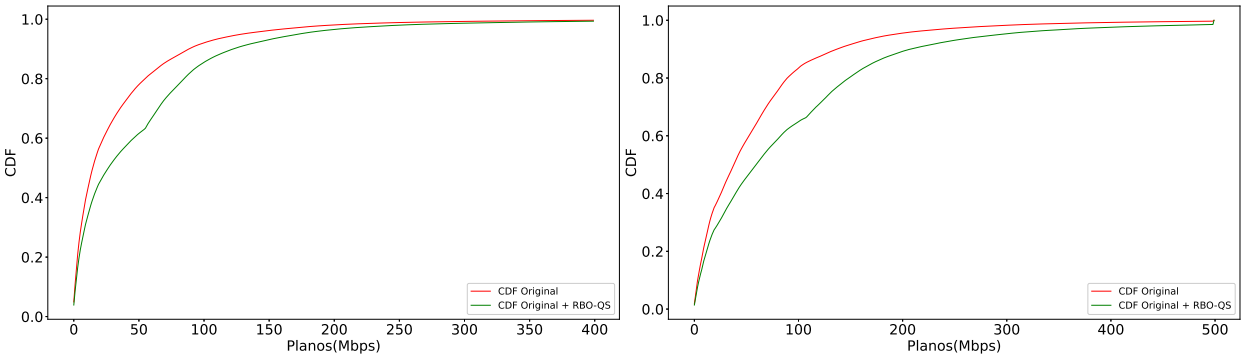
Tabela 5.4 – Porcentagem de amostras sintéticas na faixa de horário da amostra base.

Ano	Dia da Semana	Hora
2019	94,89	94,79
2020	94,90	94,85

Tabela 5.5 – Porcentagem de amostras sintéticas com característica igual à amostra base com RBO modificado

processamento. Assim, combinado ao uso de subconjuntos, usou-se o método RBO-QS.

Neste estudo adotou-se de forma arbitrária um *stepsize* de 16 sobre as bases de 2019 e 2020. O impacto da escolha do valor inicial do *stepsize* sobre a geração das amostras é um aspecto que pode ser melhor discutido em um estudo mais aprofundado. A Figura 5.6 apresenta o resultado obtido na CDF de planos.



(a) 2019 (b) 2020

Figura 5.6 – CDF da base de dados com estratificação e *oversample* do RBO-QS.

A Tabela 5.4 apresenta a similaridade de faixas de horários entre as amostras geradas e suas respectivas amostras base, com valores próximos aos da Tabela 5.3. Considerando o dia da semana e hora, a Tabela 5.5 apresenta os valores para similaridade entre dia da semana e hora, respectivamente, na qual é observado um resultado similar ao método original empregado no conjunto reduzido, como mostra a Tabela 5.2.

5.4.4 Caso de Uso: Estimativa de vazão das sessões

Esta seção apresenta os resultados de avaliação realizada sobre o efeito do método proposto em modelos de aprendizado de regressão. A avaliação foi feita usando as bases de dados do projeto Neubot dos anos de 2019 e 2020 completas. Para cada base

de dados, o conjunto de treino e teste são compostos por 70% e 30% das amostras, respectivamente. Essa proporção também é mantida em relação a características de dia da semana e faixas de horário, ou seja, para cada dia da semana e faixa de horário, 70% das amostras observadas estão no conjunto de treino e os 30% restante no conjunto de teste. Desta forma, sendo 70% das amostras de segunda-feira, na faixa de 0h-6h usadas para treino e os 30% restante para teste. Essa relação se mantém para os demais dias da semana e faixas de horário, sendo estas das 0h - 6h, 7h - 12h, 13h - 18h e 18h - 23h.

Foram usados três modelos rasos de regressão, *random forest*, *linear regression* e *bayes ridge*, considerando os conjuntos de dados originais e após a geração de amostras com o método proposto, tendo a variável de tempo como variável alvo. As métricas utilizadas para avaliação dos resultados dos modelos foram MAE (*Mean Absolute Error*) e RMSE (*Root Mean Squared Error*).

Para aplicação do método, o conjunto de dados foi agrupado em sessões, ou seja, conjunto de medições referentes aos 15 segmentos de requisitados durante cada sessão de *streaming* de vídeo. Para treinamento e teste dos modelos, estas sessões foram separadas em cada segmento. Após a geração, fez-se um tratamento para valores negativos e características com intervalos definidos. As características usadas foram: *request_ticks*, *delta_sys_time*, *iteration*, *delta_user_time*, *connect_time*, *rate*, *received*, *second*, *minute*, *hour*, *month*, *day*, *weekday*.

As Tabelas 5.6a, 5.6b e 5.6c apresentam os resultados das avaliações para o treinamento dos modelos utilizando os dados em seu formato original e padronizado, com a característica *weekday* com valores de 0 a 6 que representam os dias da semana codificados como um *one-hot array*, ou seja, um vetor de binário de 7 posições, cuja posição do dia correspondente tem valor 1 e as demais 0. As avaliações foram separadas entre o conjunto majoritário e minoritário, usando o conjunto original e o conjunto modificado com as amostras geradas.

Os resultados apresentados indicam que os dados gerados tiveram um efeito positivo sobre o aprendizado da classe minoritária, a classe de interesse, para os dados em formato original e também padronizados nos modelos *linear regression* e *Bayes Ridge*. Para esses modelos, a classe majoritária apresentou um pequeno aumento nas métricas

		MAE		RMSE	
	Base	Min	Maj	Min	Maj
Dados originais	2019	0,4304	0,4280	0,6133	0,7679
	2019 + RBO-QS	0,4470	0,4369	0,6498	0,7706
	2020	0,2419	0,3403	0,3455	0,7004
	2020 + RBO-QS	0,2759	0,3475	0,4280	0,7056
Dados pradronizados	2019	0,4986	0,5352	0,7669	0,8952
	2019 + RBO-QS	1,0869	0,6710	1,3297	0,9914
	2020	1,3382	0,7640	1,5472	1,0997
	2020 + RBO-QS	1,8097	1,2487	2,2965	1,7408

(a) Random Forest

		MAE		RMSE	
	Base	Min	Maj	Min	Maj
Dados originais	2019	0,7085	0,4409	0,7991	0,7435
	2019 + RBO-QS	0,6094	0,4791	0,7012	0,7868
	2020	0,6829	0,5226	0,7396	0,8343
	2020 + RBO-QS	0,5690	0,5405	0,6096	0,8751
Dados pradronizados	2019	0,7021	0,4414	0,7932	0,7454
	2019 + RBO-QS	0,5757	0,4944	0,6684	0,8025
	2020	0,6478	0,5203	0,7029	0,8396
	2020 + RBO-QS	0,5438	0,5442	0,5860	0,8839

(b) Linear Regression

		MAE		RMSE	
	Base	Min	Maj	Min	Maj
Dados originais	2019	0,4455	0,5060	1,8771	0,7875
	2019 + RBO-QS	0,4433	0,5365	1,8400	0,8187
	2020	0,5948	0,5142	0,6600	0,8212
	2020 + RBO-QS	0,4811	0,5120	0,5479	0,8415
Dados pradronizados	2019	0,4773	0,5923	1,6290	0,8497
	2019 + RBO-QS	0,5579	0,6525	1,7363	0,9285
	2020	0,5058	0,5034	0,5757	0,8294
	2020 + RBO-QS	0,4286	0,5300	0,4996	0,8660

(c) Bayes Regression

Tabela 5.6 – Treinamento com dados em formato original

de erro, o que pode estar relacionado ao nível de ruído inserido pelas novas amostras. O modelo *random forest* não conseguiu melhorar seu aprendizado com as amostras geradas em ambos os cenários.

5.5 Considerações Finais

Neste capítulo foram discutidas abordagens usadas para a geração de amostras sintéticas em bases que apresentam alto grau de desbalanceamento. Apresentou-se as limitações das abordagens baseadas em vizinhança, que em certas bases não consideram a posição da amostra minoritária, no espaço definido pelas suas características, e pode acabar gerando amostras ambíguas em regiões de classe majoritária. Esse cenário motivou o desenvolvimento da abordagem baseado em regiões de interesse. O método descrito considera a expansão da região a partir de operações de translação de um ponto (amostra original), dentro de espaço definido pelos atributos relevantes da base.

Essa abordagem apresenta uma melhor qualidade nas amostras geradas, quando comparada com abordagens baseada em vizinhança, entretanto tem custo computacional proibitivo quando a base apresenta grande dimensionalidade e número grande de amostras. Para viabilizar o uso de regiões de interesse nesse contexto foi proposta uma melhoria no algoritmo original. Essa melhoria combina amostragem estratificada do conjunto de dados e uma nova forma de exploração do espaço de interesse.

6

Conclusões

Neste trabalho apresentou-se uma versão do método RBO de modo a reduzir seu custo computacional e tornar viável sua aplicação em bases de dados grandes, como a do projeto Neubot. Os potenciais de geração do método RBO e da versão proposta, RBO-QS, baseados em região de interesse definida por um kernel exponencial, foram avaliados. Os resultados dessa avaliação foram comparados com os resultados obtidos usando o SMOTER, que é um algoritmo de *oversampling* baseado em medida de similaridade. Ao fim, três modelos rasos de regressão foram usados para avaliar o impacto dos dados gerados pela versão proposta.

No estudo feito sobre bases de dados do projeto Neubot, com amostras dos anos de 2015 a 2020, verificou-se que sessões originadas a partir de conexões de velocidade baixa são mais frequentes. O mesmo foi observado na análise em diferentes escalas de tempo como dia da semana e horário. Os resultados sugerem um padrão nas características dos coletores que não se altera, mesmo com o passar do tempo e que leva à baixa representação de outros grupos de dados, e.g., planos de dados que apresentam maior vazão nominal. Avaliação feita usando um modelo de predição sobre a base 2018 indicou que tal modelo não conseguiu acompanhar a variação de planos de valores mais altos, o que pode ter efeitos no contexto do *streaming* adaptativo em relação a provedores e a audiência.

Nesse contexto, avaliou-se o potencial de geração de duas abordagens: uma baseada em vizinhança, o método SMOTER, e outra baseada na região de interesse, o método RBO. A conclusão geral é que ambos os métodos conseguem reduzir o

desbalanceamento da base quando a medida alvo era observada. Entretanto, ao medir a similaridade entre as amostras geradas e as amostras bases, o SMOTER apresentou um resultado inferior ao RBO. No caso do SMOTER observou-se baixa aderência em relação a escalas de tempo menores, como o horário, que não passou de 33,46%. Embora para escalas maiores como dia da semana e faixas de horário os valores sejam maiores, chegando ao máximo de 65,79% e 85,51% respectivamente. Nos resultados do RBO foi possível observar uma aderência superior, com o mínimo ficando próximo a 88% em relação a dia da semana e horário e com valores superiores a 95% em relação a faixas de horários. Assim, o método RBO apresentou um potencial de geração com amostras mais próximas das presentes originalmente nos conjuntos de dados.

Os resultados observados com o método RBO, principalmente em relação a aderência as características de cada classe observada, inspirou a elaboração do RBO-QS. O RBO-QS utiliza estratos da base de dados e a exploração da regiões de uma forma gulosa. Observou-se que não há perda significativa na qualidade na geração das amostras em relação às características temporais de dia da semana, horário e faixas de horário.

Por fim, para avaliar o impacto dos dados gerados, utilizou-se três modelos de regressão para aprenderem sobre as bases de 2019 e 2020 originais e com as amostras do método proposto. Pelas métricas usadas na avaliação, dois dos três modelos conseguiram se beneficiar dos dados gerados quanto ao grupo minoritário. O erro do grupo majoritário apresentou um pequeno incremento. No geral, os resultados indicam que o método proposto é efetivo e mais eficiente que o algoritmo original quando se tem uma base de dados grande.

6.1 Trabalhos Futuros

Uma questão em aberto no estudo apresentado e a caracterização de algumas variáveis utilizadas pelo método proposto. Especificamente, é preciso avaliar o procedimento utilizado para definir o tamanho do espaço (região) de busca e a sua redução. Uma abordagem possível é definir o $stepsize$ como função de cada característica, considerando

medidas mais descritivas como, por exemplo, média e variância do grupo minoritário. Quanto ao protocolo de estratificação, outras abordagens podem avaliadas, e.g., uso de algoritmos de clusterização. Outras formas para exploração da região de interesse, diferentes da abordagem apresentada neste trabalho, que usa o conceito de quadrantes, também são possibilidades a serem exploradas em trabalhos futuros.

Referências

BASSO, S. et al. Measuring dash streaming performance from the end users perspective using neubot. In: *Proceedings of the 5th ACM Multimedia Systems Conference*. New York, NY, USA: Association for Computing Machinery, 2014. (MMSys '14), p. 1–6. ISBN 9781450327053. Disponível em: <https://doi.org/10.1145/2557642.2563671>. 16, 28, 29, 30

Cao, H. et al. Integrated oversampling for imbalanced time series classification. *IEEE Transactions on Knowledge and Data Engineering*, v. 25, n. 12, p. 2809–2822, 2013. 22

CHAWLA, N. V. et al. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, v. 16, p. 321–357, 2002. 17, 24, 44, 45

COELHO, M. *Um Framework para Versionamento Dinâmico de Streaming ao Vivo Baseado em Paradigma DDN*. 2018. 11, 26

HE, H. et al. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: IEEE. *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. [S.l.], 2008. p. 1322–1328. 25

HIRTH, M. et al. Crowdsourced network measurements: Benefits and best practices. *Computer Networks*, v. 90, p. 85 – 98, 2015. ISSN 1389-1286. Crowdsourcing. 31

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural Comput.*, MIT Press, Cambridge, MA, USA, v. 9, n. 8, p. 1735–1780, nov. 1997. ISSN 0899-7667. Disponível em: <https://doi.org/10.1162/neco.1997.9.8.1735>. 39

HOßFELD, T. et al. Best practices for qoe crowdtesting: Qoe assessment with crowdsourcing. *IEEE Transactions on Multimedia*, v. 16, n. 2, p. 541–558, 2014. 16

HUANG, J. et al. A crowdsource-based sensing system for monitoring fine-grained air quality in urban environments. *IEEE Internet of Things Journal*, v. 6, n. 2, p. 3240–3247, 2019. 16

JAIN, M. T. M. Z. R. *WUSTL-IIOT-2018*. IEEE Dataport, 2020. Disponível em: <http://dx.doi.org/10.21227/kzgp-7t84>. 22

KANKANAMGE, N. et al. Can volunteer crowdsourcing reduce disaster risk? a systematic review of the literature. *International Journal of Disaster Risk Reduction*, v. 35, p. 101097, 2019. ISSN 2212-4209. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2212420918310410>. 16

KOZIARSKI, M.; KRAWCZYK, B.; WOŹNIAK, M. Radial-based oversampling for noisy imbalanced data classification. *Neurocomputing*, Elsevier, v. 343, p. 19–33, 2019. 12, 17, 18, 59

LAB, M. M-lab. Último acesso em Julho, 2020. 2020. Disponível em: <https://www.measurementlab.net/data>. 29

LEEVY, J. L. et al. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, Springer, v. 5, n. 1, p. 42, 2018. 21, 23

LI, Z. et al. Imbalanced adversarial learning for weather image generation and classification. In: IEEE. *2018 14th IEEE International Conference on Signal Processing (ICSP)*. [S.l.], 2018. p. 1093–1097. 16, 24

LUCIC, M. C. et al. Leveraging intelligent transportation systems and smart vehicles using crowdsourcing: An overview. *Smart Cities*, v. 3, n. 2, p. 341–361, 2020. ISSN 2624-6511. Disponível em: <https://www.mdpi.com/2624-6511/3/2/18>. 16

MAXMIND. https://www.maxmind.com/en/home. Último acesso em Junho, 2019. 2019. 30

MONIZ, N.; BRANCO, P.; TORGO, L. Resampling strategies for imbalanced time series forecasting. *International Journal of Data Science and Analytics*, Springer, v. 3, n. 3, p. 161–181, 2017. 22, 66

PAZ, J. d. M. L. F. Tonny Frank Osaki da; MELO, C. A. V. Assessing tcp throughput stability of sub-30s video streaming sessions. Submitted to IEEE LatinCom2020. 2020. 31

Schuster, M.; Paliwal, K. K. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, v. 45, n. 11, p. 2673–2681, 1997. 39

SILVA, M. P. *Predição de vazão em fluxo contínuo adaptativo de vídeo utilizando aprendizado profundo*. Dissertação (Mestrado) — Instituto de Computação, Universidade Federal do Amazonas, Manaus, 2020. 39

TORGO, L. et al. Smote for regression. In: SPRINGER. *Portuguese conference on artificial intelligence*. [S.l.], 2013. p. 378–389. 17, 25, 44, 45, 60

XIE, Y. et al. Generative learning for imbalanced data using the gaussian mixed model. *Applied Soft Computing*, Elsevier, v. 79, p. 439–451, 2019. 16, 24

YI, W.; SUN, Y.; HE, S. Data augmentation using conditional gans for facial emotion recognition. In: IEEE. *2018 Progress in Electromagnetics Research Symposium (PIERS-Toyama)*. [S.l.], 2018. p. 710–714. 24

ZHANG, X. et al. Cgmos: Certainty guided minority oversampling. In: ACM. *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. [S.l.], 2016. p. 1623–1631. 24