Trabalho de Conclusão de Curso

Avaliação de Métodos de Agregação em Ambientes Federados: Aplicações em Classificação de Imagens com Redes Neurais Convolucionais

Ester de Lima Pontes Andrade
elpa@ic.ufal.br

Orientador:

Prof. Dr. André Luiz Lins de Aquino

Maceió, Agosto de 2024

Ester de Lima Pontes Andrade

Avaliação de Métodos de Agregação em Ambientes Federados: Aplicações em Classificação de Imagens com Redes Neurais Convolucionais

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação do Instituto de Computação da Universidade Federal de Alagoas.

Orientador:

Prof. Dr. André Luiz Lins de Aquino

Maceió, Agosto de 2024

Catalogação na fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico
Bibliotecária: Taciana Sousa dos Santos – CRB-4 – 2062

A554a Andrade, Ester de Lima Pontes.
Avaliação de métodos de agregação em ambientes federados : aplicações em classificação de imagens com redes neurais convolucionais / Ester de Lima Pontes Andrade. – 2024.
53 f. : il. color.

Orientador: André Luiz Lins de Aquino.
Monografia (Trabalho de Conclusão de Curso em Engenharia da Computação) – Universidade Federal de Alagoas. Instituto de Computação. Maceió, 2024.

Bibliografia: f. 51-53.

1. Aprendizado federado. 2. Agregação (Computação). 3. Classificação de imagens. 4. Redes neurais convolucionais. I. Título.

CDU: 004

UNIVERSIDADE FEDERAL DE ALAGOAS/UFAL
Instituto de Computação – IC
Campus A. C. Simões - Av. Lourival de Melo Mota, BL 12
Tabuleiro do Martins, Maceió/AL - CEP: 57.072-970
Telefone: (082) 3214-1401

Trabalho de Conclusão de Curso - TCC

Formulário de Avaliação

Nome do Aluno
Ester de Lima Pontes Andrade

Nº de Matrícula
18211015

Título do TCC (Tema)
Avaliação de Métodos de Agregação em Ambientes Federados: Aplicações em Classificação de Imagens com Redes Neurais Convolucionais

Banca Examinadora

André Luiz Lins de Aquino
Nome do Orientador

ANDRE LUIZ LINS DE AQUINO:03235015400
Assinado de forma digital por ANDRE LUIZ LINS DE AQUINO:03235015400
Dados: 2024.08.22 15:16:52 -03'00'
Assinatura

Douglas Moura
Nome do Professor

Documento assinado digitalmente
gov.br DOUGLAS LEITE LEAL MOURA
Data: 22/08/2024 15:53:04-0300
Verifique em https://validar.iti.gov.br
Assinatura

Geymerson Ramos
Nome do Professor

Documento assinado digitalmente
gov.br GEYMERSON DOS SANTOS RAMOS
Data: 22/08/2024 16:06:16-0300
Verifique em https://validar.iti.gov.br
Assinatura

Data da Defesa		Nota Obtida
22/08/2024		10 (dez)

Observações:

Coordenador do Curso
De Acordo

Documento assinado digitalmente
gov.br JOBSON DE ARAUJO NASCIMENTO
Data: 23/08/2024 13:45:03-0300
Verifique em https://validar.iti.gov.br
Assinatura

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação do Instituto de Computação da Universidade Federal de Alagoas, aprovada pela comissão examinadora que abaixo assina.

Prof. Dr. André Luiz Lins de Aquino - Orientador
Instituto de Computação
Universidade Federal de Alagoas

Douglas Leite Leal Moura - Examinador
Universidade Federal de Minas Gerais

Geymerson dos Santos Ramos - Examinador
Universidade Federal de Alagoas

Maceió, Agosto de 2024

Dedicatória

À minha vó Débora, que estudou só até a terceira
série, mas que sempre deixa a porta da casa dela
aberta para que possamos concluir nossos estudos.
Todas as nossas conquistas são suas, Beba.

Agradecimentos

Em primeiro lugar, agradeço a Deus por guiar meus passos, pensamentos e palavras. À minha família, cada um à sua maneira, foi imprescindível em minha jornada acadêmica. Em especial, expresso minha gratidão à minha avó Débora, à minha tia Leni e à minha prima Catarina por me acolherem em sua casa durante minha graduação, oferecendo-me muito mais que um lar. Aos meus pais, Leide e Ronaldo, e à minha irmã, Manuela, agradeço pelo apoio, amor e auxílio incondicionais.

Registro um agradecimento especial aos amigos que conquistei durante a graduação e que levarei para a vida: Kelly Bianca, Julios Rocha, Kamila Almeida e Bruno Severo, cujo acolhimento inicial foi fundamental. Agradeço também ao meu amigo e fiel companheiro de estudos, Anderson Clemente, por estar sempre ao meu lado, compartilhando as dificuldades e me incentivando a dar o meu melhor. No final, nós vivemos.

Agradeço a Karen Gomes e Larissa Santana, por serem muito mais que colegas de curso, por serem suporte e apoio em todos os momentos em que pensei não conseguir seguir em frente. A Rafael Augusto, por seu constante incentivo e suporte nas disciplinas.

Agradeço às colegas do Instituto de Computação, com quem compartilhei momentos desafiadores, mas que sempre estiveram prontas para nos apoiar mutuamente. Em especial, expresso minha gratidão a Rebeca Brandão, Natália Almeida, Luana Ferreira, Thalia Barbosa, Lillian Giselly, Lilian Fabrício, Ana Carolina Nesso e Lívia Soares.

Minha profunda gratidão à minha amiga, irmã e confidente Isabely Leal, por sempre me inspirar, incentivar, escutar e compreender; minhas vitórias são suas também. A Yasmin Medeiros, por toda parceria, cuidado e atenção, que foram fundamentais para a conclusão desta jornada.

Agradeço aos projetos de extensão dos quais fiz parte, especialmente ao Grupo Katie, bem como ao Octacore e à Secomp, que enriqueceram minha trajetória acadêmica e profissional. Ao Diretório Acadêmico e aos colegas de gestão, meu reconhecimento pelo aprendizado compartilhado.

Meus sinceros agradecimentos ao Laccan/Orion, laboratórios que me acolheu e estimulou meu desenvolvimento tanto acadêmico quanto profissional. Aos colegas de residência, que foram primordiais em minha trajetória profissional e uma constante fonte de inspiração. Agradecimento especial ao professor Dr. André Aquino, por possibilitar minha colaboração com os

laboratórios e por seu direcionamento e incentivo, que foram essenciais para a concretização deste projeto.

Expresso minha gratidão ao Instituto de Computação e ao corpo docente, cuja alta qualificação proporciona um ambiente de excelência acadêmica. Em particular, agradeço ao professor Thiago Cordeiro por me acolher nas primeiras aulas e me escutar ao longo de toda a graduação, e à professora Eliana Almeida, por ser uma grande inspiração acadêmica.

Manifesto minha sincera gratidão à banca examinadora por aceitar o convite para avaliar este trabalho. Sua participação neste processo é de grande valor para mim.

Estendo meus agradecimentos aos técnicos do Instituto de Computação, especialmente a Ana Ferreira, por sua constante disposição em ajudar, pelas conversas inspiradoras e pelo carinho sempre presente. Agradeço também ao Marcelo, por sua eficiência e disposição em lidar com as burocracias dos processos. À Tia Lúcia, minha gratidão por cuidar do Instituto com tanto carinho e por sempre me acolher com afeto.

Finalmente, agradeço a todos que contribuíram para a realização deste trabalho, ajudando a concluir um ciclo importante. A todos que cruzaram meu caminho, cada troca, gesto de incentivo e experiência vivida foi fundamental para o meu desenvolvimento.

Ester de Lima

“ [...]
Tudo, tudo, tudo que nóis tem é nóis
Tudo, tudo, absolutamente tudo
Tudo que nóis tem é isso, uns aos outros
[...] “

– Emicida, *Principia (2019)*

Resumo

Este trabalho apresenta uma análise técnica detalhada dos métodos de agregação FedAvg, FedProx e WeightedFedAvg no contexto do aprendizado federado, aplicados a redes neurais convolucionais para tarefas de classificação em três datasets: MNIST, FashionMNIST e CIFAR-10. O aprendizado federado, uma abordagem descentralizada de treinamento de modelos, permite a colaboração entre múltiplos dispositivos mantendo os dados localmente, o que é crucial para a preservação da privacidade em cenários distribuídos.

Os experimentos foram conduzidos em um ambiente de dados não IID (não independentemente e identicamente distribuídos), replicando condições do mundo real onde a distribuição dos dados entre os clientes é heterogênea. Para cada método de agregação, o treinamento local foi realizado por 5 épocas em cada cliente, seguido da agregação central das atualizações dos pesos, com o ciclo repetido por 10 rodadas. A avaliação do desempenho foi realizada com base em métricas tradicionais de acurácia e perda, além de análises mais sofisticadas utilizando curvas ROC/AUC e matrizes de confusão, proporcionando uma avaliação granular da capacidade discriminativa dos modelos e dos padrões de erro específicos.

Os resultados obtidos revelam que o método WeightedFedAvg supera consistentemente os demais em ambientes de alta heterogeneidade e desbalanceamento de dados, especialmente no CIFAR-10, atingindo uma acurácia superior e menor perda, o que reforça sua adequação para cenários federados desafiadores. O FedProx demonstrou estabilidade em cenários com grande variabilidade entre os dados dos clientes, mitigando a divergência das atualizações locais e preservando a coesão do modelo global. Por outro lado, o FedAvg, embora eficiente em ambientes com distribuição de dados mais homogênea, mostrou limitações significativas em cenários mais complexos, alinhando-se com os desafios discutidos na literatura.

A pesquisa conclui que, embora FedAvg permaneça uma solução eficaz para ambientes federados homogêneos, a crescente complexidade dos cenários reais exige a adoção de métodos de agregação mais avançados, como FedProx e WeightedFedAvg, que demonstram maior resiliência e precisão em contextos de alta variabilidade dos dados. Estes achados indicam a necessidade de futuras investigações que possam desenvolver novos algoritmos de agregação capazes de equilibrar ainda mais a estabilidade e a precisão dos modelos em ambientes federados complexos e desafiadores.

Palavras-chave: Aprendizado Federado, Métodos de Agregação, FedAvg, FedProx, WeightedFedAvg, Classificação de Imagens, MNIST, FashionMNIST, CIFAR-10.

Abstract

This study presents a detailed technical analysis of the FedAvg, FedProx, and WeightedFedAvg aggregation methods in the context of federated learning, applied to convolutional neural networks for classification tasks on three datasets: MNIST, FashionMNIST, and CIFAR-10. Federated learning, a decentralized model training approach, facilitates collaboration across multiple devices while maintaining data locally, which is crucial for preserving privacy in distributed scenarios.

The experiments were conducted in a non-IID (non-independently and identically distributed) data environment, replicating real-world conditions where data distribution among clients is heterogeneous. For each aggregation method, local training was performed for 5 epochs on each client, followed by central aggregation of the weight updates, with this cycle repeated for 10 rounds. Performance evaluation was based on traditional metrics such as accuracy and loss, as well as more sophisticated analyses using ROC/AUC curves and confusion matrices, providing a granular assessment of the models' discriminative capabilities and specific error patterns.

The results reveal that the WeightedFedAvg method consistently outperforms the others in environments characterized by high heterogeneity and data imbalance, particularly on CIFAR-10, achieving superior accuracy and lower loss, which reinforces its suitability for challenging federated learning scenarios. FedProx demonstrated stability in settings with significant variability among client data, mitigating divergence in local updates and preserving the cohesion of the global model. In contrast, while FedAvg is efficient in more homogeneous data distribution environments, it showed significant limitations in more complex scenarios, aligning with the challenges discussed in the literature.

The research concludes that, although FedAvg remains an effective solution for homogeneous federated environments, the increasing complexity of real-world scenarios necessitates the adoption of more advanced aggregation methods, such as FedProx and WeightedFedAvg, which exhibit greater resilience and accuracy in contexts of high data variability. These findings indicate the need for future investigations to develop new aggregation algorithms capable of further balancing the stability and accuracy of models in complex and challenging federated environments.

Keywords: Federated Learning, Aggregation Methods, FedAvg, FedProx, WeightedFedAvg, Image Classification, MNIST, FashionMNIST, CIFAR-10.

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

IoT	Internet das Coisas
Não-IID	Não Independentes e Identicamente Distribuídos
IID	Independentes e Identicamente Distribuídos (Independent and Identically Distributed)
FL	Aprendizado Federado (Federated Learning)
FedAvg	Federated Averaging
FedProx	Federated Proximal
WeightedFedAvg	Federated Weighted Averaging
AUC	Área Sob a Curva (Area Under the Curve)
ROC	Curva Característica de Operação do Receptor (Receiver Operating Characteristic)
MNIST	Modified National Institute of Standards and Technology
FashionMNIST	Fashion Modified National Institute of Standards and Technology
CIFAR-10	Canadian Institute For Advanced Research
FedMA	Federated Matched Averaging
FedBe	Federated Bootstrapping Ensemble
Agnostic Federated Learning	Aprendizado Federado Agnóstico
SGD	Stochastic Gradient Descent
ReLU	Rectified Linear Unit
FC	Fully Connected (Camadas Totalmente Conectadas)
RGB	Red, Green, Blue (Canais de cor)
Dropout	Técnica para evitar o Overfitting (Regularização)
Loss	Função de Perda (Entropia Cruzada Negativa)
Max Pooling	Técnica de Redução de Dimensionalidade
Google Colab	Plataforma de Desenvolvimento com Suporte a GPUs

GANs	Redes Generativas Adversariais (Generative Adversarial Networks)
GPU	Graphics Processing Units

Conteúdo

1

Introdução

1.1 Contextualização e Motivação

Nos últimos anos, a privacidade dos dados tem se tornado uma preocupação crescente em diversas áreas, desde a saúde até as finanças. À medida que a quantidade de dados pessoais coletados e processados por dispositivos inteligentes aumenta, surge a necessidade de desenvolver métodos de aprendizado de máquina que respeitem a privacidade dos usuários. O Aprendizado Federado (Federated Learning) emerge como uma solução atraente para distribuição de treinamento, permitindo que modelos de aprendizado sejam treinados diretamente nos dispositivos locais, sem a necessidade de centralizar os dados em servidores remotos. Essa abordagem mantém os dados privados em cada dispositivo, ao mesmo tempo, em que permite o desenvolvimento de modelos robustos e eficazes (Kairouz et al., 2021).

O conceito de Aprendizado Federado foi introduzido pela primeira vez pelo Google em 2016, visando habilitar o treinamento colaborativo de modelos em larga escala, preservando a privacidade dos dados dos usuários (Konečnỳ et al., 2016). Essa técnica se tornou particularmente relevante em cenários onde os dados são distribuídos em dispositivos móveis ou sistemas de Internet das Coisas (IoT) onde a coleta centralizada de dados não é viável ou desejável. A arquitetura distribuída do Aprendizado Federado tem não só o potencial de transformar a maneira como o aprendizado de máquina é aplicado em ambientes sensíveis à privacidade, como também levanta novos desafios, especialmente no que diz respeito à eficácia e eficiência dos modelos treinados (Konečnỳ et al., 2016) (Bonawitz et al., 2019).

Um dos principais desafios é a heterogeneidade dos dados distribuídos entre os diferentes dispositivos. Em muitos casos, os dados disponíveis em cada dispositivo não são Independentes e Identicamente Distribuídos (não-IID), o que pode levar a dificuldades na convergência do modelo global e na generalização do modelo para novos dados (McMahan et al., 2017) (Li et al., 2020). Além disso, a variação na quantidade de dados disponíveis em cada dispositivo

pode impactar a eficácia do modelo federado, exigindo métodos de agregação robustos que sejam capazes de lidar com essas disparidades (Qi et al., 2023).

Os métodos de agregação são um componente crucial no Aprendizado Federado, pois determinam como as atualizações dos modelos locais, treinados em diferentes dispositivos, são combinadas para formar um modelo global (Qi et al., 2023). O método *Federated Averaging* (FedAvg) é o mais utilizado, pois é simples e eficaz em muitos cenários (Reyes et al., 2021). No entanto, em situações com dados altamente heterogêneos, o FedAvg pode não ser suficiente para garantir um desempenho robusto, levando ao desenvolvimento de métodos alternativos, como o *Federated Proximal* (FedProx) e o *Federated Weighted Averaging* (WeightedFedAvg), que buscam mitigar as limitações do FedAvg (Li et al., 2020) (Almodóvar et al., 2024).

O FedProx, por exemplo, introduz um termo de regularização que penaliza grandes desvios entre os parâmetros dos modelos locais e o modelo global, buscando melhorar a estabilidade e a convergência do modelo em cenários com alta heterogeneidade de dados (Li et al., 2020). Por outro lado, o WeightedFedAvg pondera as contribuições de cada cliente na agregação conforme o tamanho do seu conjunto de dados, buscando equilibrar a influência de cada cliente no modelo global (Almodóvar et al., 2024). Essas variações destacam a importância de escolher o método de agregação adequado, dependendo das características dos dados e do ambiente de aplicação.

O contexto de aplicação do Aprendizado Federado é amplamente variado, abrangendo desde a personalização de modelos de predição em dispositivos móveis até a criação de sistemas de saúde que respeitam a privacidade dos pacientes (Hayat et al., 2022). Em particular, a aplicação de Aprendizado Federado em tarefas de classificação de imagens tem atraído grande atenção, devido à abundância de dados visuais em dispositivos pessoais e ao potencial de melhoria dos modelos preditivos sem comprometer a privacidade dos usuários (Reyes et al., 2021). Datasets como MNIST, FashionMNIST, e CIFAR-10 são frequentemente utilizados para avaliar a eficácia de diferentes métodos de Aprendizado Federado, dada sua popularidade e a complexidade das tarefas de classificação que eles representam (Zhao et al., 2018).

Além da privacidade, outro fator motivador para o uso de Aprendizado Federado é a eficiência. Treinar modelos diretamente nos dispositivos reduz a necessidade de transferência de grandes volumes de dados pela rede, o que não apenas protege a privacidade, porém melhora a eficiência em termos de latência e uso de largura de banda (Bonawitz et al., 2019). No entanto, essa abordagem também impõe desafios computacionais significativos, especialmente em dispositivos com recursos limitados, como smartphones e sensores IoT. Isso torna a escolha do método de agregação ainda mais crítica, pois impacta diretamente a carga computacional e a eficiência do sistema todo.

Apesar de promissor, o Aprendizado Federado ainda precisa ser amplamente testado e validado. Neste contexto, o presente trabalho se insere na linha de pesquisa que visa explorar e comparar diferentes métodos de agregação aplicados a diferentes datasets, com o intuito de entender melhor suas vantagens e limitações em cenários de Aprendizado Federado. Ao avaliar a forma que métodos como FedAvg, FedProx, e WeightedFedAvg se comportam em datasets

populares de classificação de imagens, esta pesquisa busca fornecer percepções práticas para a implementação de sistemas de Aprendizado Federado mais eficazes e eficientes.

1.2 Objetivos do trabalho

1.2.1 Objetivo Geral

Investigar e comparar o desempenho de diferentes métodos de agregação no contexto do Aprendizado Federado, aplicados a diversas tarefas de classificação de imagens utilizando redes neurais.

Este objetivo geral reflete a intenção de entender como métodos de agregação, como FedAvg, FedProx, e WeightedFedAvg, se comportam em diferentes cenários e com diferentes datasets (MNIST, FashionMNIST, CIFAR-10). O foco está em avaliar a eficácia desses métodos em termos de acurácia, perda e outras métricas relevantes.

1.2.2 Objetivos específicos

- **Avaliar o Desempenho dos Métodos de Agregação:** Medir e comparar o desempenho dos métodos FedAvg, FedProx, e WeightedFedAvg em termos de acurácia, perda, e AUC, aplicados a datasets de classificação de imagens como MNIST, FashionMNIST e CIFAR-10.
- **Analisar o Impacto da Heterogeneidade dos Dados:** Estudar como a heterogeneidade dos dados entre os diferentes clientes (distribuição não-IID) afeta o desempenho dos métodos de agregação e a qualidade do modelo global no Aprendizado Federado.
- **Comparar a Robustez dos Métodos de Agregação em Diferentes Datasets:** Comparar como cada método de agregação se comporta em diferentes datasets, identificando quais métodos são mais eficazes para cada tipo de tarefa de classificação.
- **Propor Recomendações para a Implementação Prática do Aprendizado Federado:** Com base nos resultados obtidos, fornecer recomendações sobre a escolha de métodos de agregação para diferentes cenários de aplicação, destacando suas vantagens e limitações.

1.3 Estrutura do Trabalho

Nesta seção será apresentada a organização desta monografia. Após este capítulo de introdução, o trabalho está estruturado em mais seis capítulos.

No **Capítulo 2**, é realizada a Revisão da Literatura, onde são discutidos os conceitos fundamentais do Aprendizado Federado, métodos de agregação, e suas aplicações em diferentes contextos, além de explorar os principais desafios e avanços na área.

No **Capítulo 3**, são apresentados os Trabalhos Relacionados, onde são discutidos os estudos mais relevantes e citados na literatura sobre métodos de agregação em Aprendizado Federado, comparando abordagens e resultados obtidos por outros pesquisadores.

No **Capítulo 4**, é detalhada a Metodologia utilizada para o desenvolvimento do trabalho, incluindo a descrição dos datasets, as configurações experimentais, os métodos de agregação implementados, e as métricas utilizadas para avaliação dos modelos.

No **Capítulo 5**, são expostos os Resultados e Discussões, onde são analisados os resultados obtidos nos experimentos, incluindo comparações de acurácia, perda, curvas ROC/AUC, e matrizes de confusão, discutindo-se as implicações e relevância dos achados.

No **Capítulo 6**, são apresentadas as Considerações Finais, onde se conclui o trabalho, discutindo suas contribuições e impacto na área de Aprendizado Federado. Além disso, são abordadas as limitações do estudo e propostas de Trabalhos Futuros, sugerindo direções e passos subsequentes para o aprofundamento da pesquisa.

2

Fundamentação Teórica

Este capítulo apresenta a fundamentação teórica necessária para embasar o desenvolvimento desta pesquisa. São abordados os principais conceitos relacionados ao Aprendizado Federado, os métodos de agregação utilizados, e suas aplicações em diferentes cenários. Esta síntese teórica fornece o suporte conceitual para os experimentos e análises realizadas ao longo do trabalho.

2.1 Aprendizado Federado

O conceito de Aprendizado Federado foi introduzido por McMahan et al. (2017) como uma nova abordagem para treinar modelos de aprendizado de máquina em dispositivos distribuídos, mantendo os dados localmente ao invés de centralizá-los em um servidor. No Aprendizado Federado, os dispositivos locais, conhecidos como clientes, atualizam seus modelos de forma independente. Em vez de enviar os dados brutos, esses dispositivos enviam apenas as atualizações dos modelos para um servidor central, onde essas atualizações são agregadas para formar um modelo global.

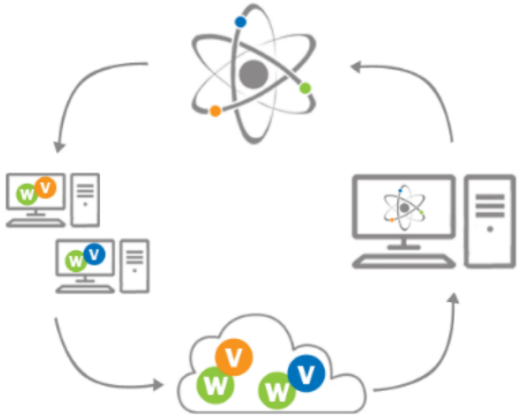


Figura 1: Fluxo de Dados no Aprendizado Federado (FL). Fonte: Reyes et al. (2021)

Na Figura 1, observa-se o processo de Aprendizado Federado onde cada dispositivo (cliente) realiza o treinamento localmente usando seus próprios dados, representados pelos ícones coloridos V e W. Esses dispositivos não compartilham os dados brutos com o servidor central; em vez disso, eles enviam apenas as atualizações dos modelos locais para o servidor. O servidor central, representado pelo ícone central superior, agrega essas atualizações para formar um modelo global mais robusto. O modelo global atualizado é então distribuído de volta aos dispositivos, onde o processo de treinamento continua iterativamente, respeitando a privacidade dos dados locais (Reyes et al., 2021).

Essa metodologia de preservação da privacidade é particularmente crucial em contextos delicados, como a saúde, onde os dados dos pacientes não podem ser compartilhados facilmente devido a regulamentos de confidencialidade (Kairouz et al., 2021). Além disso, ele se destaca na redução do custo computacional, evitando a centralização de grandes volumes de dados ao distribuir o processamento pelos próprios dispositivos dos usuários (Yang et al., 2019). Seu potencial para minimizar riscos de privacidade e otimizar recursos computacionais o torna uma abordagem promissora para diversas aplicações industriais e de pesquisa.

Nesse contexto, além dos métodos FedAvg, FedProx e WeightedFedAvg, outros métodos de agregação têm sido propostos para enfrentar os desafios do Aprendizado Federado, especialmente em redes heterogêneas. Por exemplo, Wang et al. (2020) introduziram o Federated Matching Averaging (FedMA), que permite a agregação de modelos sem a necessidade de alinhamento prévio das camadas, otimizando o desempenho em ambientes onde a arquitetura dos modelos locais pode variar entre os clientes. Este método é particularmente útil em cenários onde a heterogeneidade dos modelos e dos dados é alta, permitindo uma flexibilidade maior na integração dos resultados.

Outra abordagem é o Federated Bayesian Ensemble (FedBe), proposto por Chen and Chao (2020). O FedBe é um método baseado em *ensembles* que evita a necessidade de agregação direta dos parâmetros dos modelos locais. Em vez disso, ele combina as predições dos modelos locais de maneira probabilística, gerando um modelo global que incorpora a diversidade dos modelos treinados localmente. Este método demonstrou melhorias significativas em termos de robustez e acurácia, especialmente em cenários onde a variabilidade dos dados entre os clientes é elevada. Essas abordagens, FedMA e FedBe, ampliam a gama de possibilidades de métodos de agregação presentes no Aprendizado Federado, oferecendo alternativas que lidam com as limitações dos métodos tradicionais em diferentes contextos.

2.2 Métodos de Agregação no Aprendizado Federado

Outro ponto muito estudado na literatura é a agregação das atualizações dos modelos locais. O método mais comumente utilizado é o Federated Averaging (FedAvg), que foi proposto por McMahan et al. (2017). O FedAvg combina as atualizações dos modelos locais calculando a

média ponderada dessas atualizações, o que funciona bem em muitos cenários. No entanto, FedAvg pode enfrentar dificuldades em cenários onde os dados são não-IID (não independente e identicamente distribuídos), levando ao desenvolvimento de métodos alternativos, como o FedProx e o WeightedFedAvg.

O FedProx, introduzido por Li et al. (2020), é uma variação do FedAvg que adiciona um termo proximal à função de custo, penalizando grandes desvios entre os parâmetros dos modelos locais e o modelo global. Isso melhora a estabilidade e a convergência em cenários com dados heterogêneos, garantindo que o modelo global seja menos suscetível a divergências causadas por clientes com distribuições de dados muito diferentes (Li et al., 2020).

O WeightedFedAvg, por sua vez, ajusta a influência de cada cliente na agregação com base no tamanho do seu conjunto de dados, equilibrando a contribuição dos clientes e melhorando a desempenho em cenários com clientes com quantidades muito diferentes de dados. Esse método ponderado tem mostrado resultados promissores, especialmente em situações onde alguns dispositivos possuem quantidades muito maiores de dados do que outros, evitando que modelos globais sejam fortemente influenciados por clientes com menos dados (Wang et al., 2020).

2.3 Heterogeneidade dos Dados e Desafios no Aprendizado Federado

Um dos desafios mais complexos no Aprendizado Federado é lidar com a heterogeneidade dos dados distribuídos entre os clientes. Em muitos cenários práticos, os dados presentes em cada dispositivo são altamente não-IID, ou seja, os conjuntos de dados em cada cliente podem apresentar características substancialmente diferentes (Zhao et al., 2018). Por exemplo, em uma aplicação móvel, as preferências e comportamentos dos usuários podem variar significativamente, resultando em distribuições de dados que diferem amplamente entre dispositivos (Smith et al., 2017).

Essa heterogeneidade pode levar a problemas na convergência do modelo global e na generalização para novos dados. Diversas pesquisas têm se concentrado em desenvolver métodos de agregação que sejam resilientes a essa heterogeneidade, visando melhorar a eficácia do Aprendizado Federado em ambientes não controlados (Hsu et al., 2019). Estratégias como a introdução de regularizadores específicos ou o ajuste dinâmico das taxas de aprendizado para diferentes clientes têm sido propostas para mitigar esses problemas (Li et al., 2020).

Adicionalmente, Konečnỳ et al. (2016) abordaram a eficiência da comunicação em ambientes federados, propondo estratégias para reduzir a quantidade de dados transmitidos entre clientes e servidor, crucial em cenários com conexões de rede limitadas. Essa pesquisa é particularmente relevante em contextos de IoT, onde a largura de banda e a latência são restrições significativas.

2.4 Aplicações do Aprendizado Federado

O Aprendizado Federado é especialmente relevante em setores onde a proteção dos dados é fundamental. No campo da saúde, por exemplo, essa técnica permite a construção de modelos preditivos a partir de dados sensíveis sem necessidade de centralização (Rieke et al., 2020). Isso possibilita o desenvolvimento de sistemas de diagnóstico baseados em Machine Learning, em que hospitais e instituições médicas podem colaborar para criar modelos precisos e mantendo a privacidade das informações. Um estudo conduzido por Sheller et al. (2020) demonstrou a eficácia do Aprendizado Federado em radiologia, ao combinar modelos treinados em diferentes hospitais para aprimorar a detecção de tumores cerebrais em ressonâncias magnéticas.

A aplicação nos dispositivos móveis, como teclados preditivos, assistentes virtuais, e recomendação de conteúdo utilizam Aprendizado Federado para melhorar a experiência do usuário sem a necessidade de transferir dados pessoais para servidores centralizados (Hard et al., 2018). Datasets como MNIST, FashionMNIST, e CIFAR-10 são frequentemente utilizados em pesquisas para testar e validar métodos de Aprendizado Federado devido à sua popularidade e à complexidade que apresentam para tarefas de classificação (Xiao et al., 2017; Reyes et al., 2021).

Estudos sobre personalização em dispositivos IoT, como o realizado por Li et al. (2020), evidenciam como o Aprendizado Federado permite adaptar modelos de aprendizado às necessidades individuais de cada dispositivo, garantindo que os dados sensíveis permaneçam localmente, sem necessidade de compartilhamento externo. Além disso, Brisimi et al. (2018) exploraram a aplicação do modelo federado na detecção de intrusões em redes de computadores, demonstrando que diferentes servidores podem colaborar para identificar ameaças cibernéticas sem expor logs de rede mutualmente.

2.5 Aprendizado Federado aplicados nos Datasets MNIST, FashionMNIST e CIFAR-10

2.5.1 MNIST e FashionMNIST

Deng (2012) destacou a importância do MNIST como benchmark padrão para tarefas de classificação de dígitos manuscritos, sugerindo que, embora seja um dataset essencial para testar algoritmos de aprendizado de máquina, ele não captura a complexidade dos problemas do mundo real. Por isso, datasets como o FashionMNIST foram propostos para oferecer desafios adicionais, apresentando imagens de artigos de moda que possuem características mais variadas e desafiadoras para modelos de visão computacional. (Xiao et al., 2017).

2.5.2 CIFAR-10

Krizhevsky et al. (2012) introduziu o CIFAR-10, um dataset composto por 60.000 imagens de 10 classes, que se tornou um dos benchmarks mais utilizados para a avaliação de redes neurais convolucionais. Estudos como o de Cubuk et al. (2020) demonstraram que redes treinadas no CIFAR-10 podem ser eficazmente transferidas para novas tarefas visuais por meio de técnicas de transfer learning, onde as camadas iniciais da rede, que capturam características de baixo nível, são reutilizadas enquanto as camadas superiores são ajustadas para tarefas específicas.

3

Trabalhos relacionados

O Aprendizado Federado tem se consolidado como uma abordagem promissora para o treinamento de modelos de Machine Learning em ambientes distribuídos, onde a privacidade dos dados é uma preocupação central. Diversos estudos têm explorado diferentes métodos de agregação para otimizar o desempenho dos modelos nesse contexto, levando em conta as particularidades dos dados distribuídos e não independentes. Este capítulo apresenta uma revisão dos principais trabalhos relacionados ao tema, destacando as contribuições mais relevantes na literatura, as técnicas utilizadas e os resultados alcançados. A análise desses trabalhos oferece uma base comparativa que fundamenta o desenvolvimento e a avaliação das estratégias adotadas nesta monografia.

3.1 Comparação de Métodos de Agregação

A comparação de métodos de agregação no Aprendizado Federado tem sido uma área de intenso estudo, para identificar quais técnicas são mais eficazes em diferentes cenários. Li et al. (2020) realizaram uma análise comparativa detalhada entre os métodos FedAvg e FedProx, ambos amplamente utilizados em Aprendizado Federado. A pesquisa revelou que o FedProx oferece uma vantagem significativa em cenários com alta heterogeneidade de dados, ou seja, quando os dados distribuídos entre os clientes não seguem uma distribuição independente e identicamente distribuída (não-IID). O FedProx se destaca por introduzir um termo proximal que penaliza desvios extremos nos parâmetros dos modelos locais em relação ao modelo global, promovendo uma convergência mais estável em situações onde os dados locais são muito variados. Esse método é especialmente relevante para aplicações em que diferentes dispositivos ou usuários possuem dados com características substancialmente diferentes, como em dispositivos móveis ou em sistemas de saúde distribuídos (Li et al., 2019).

Li et al. (2018) expandiram essa discussão ao explorar o impacto de diferentes estratégias de ponderação na agregação dos modelos, com foco no WeightedFedAvg. A técnica proposta

ajusta o peso de cada cliente na agregação com base na quantidade de dados disponíveis em cada dispositivo. Isso significa que clientes com mais dados influenciam mais o modelo global, uma abordagem que pode evitar que modelos globais sejam enviesados por clientes com dados insuficientes. O estudo concluiu que o WeightedFedAvg melhora significativamente a robustez dos modelos globais, especialmente em cenários onde há uma disparidade substancial no volume de dados entre os clientes.

3.2 Aprendizado Federado com Dados não-IID

O desafio dos dados não-IID no Aprendizado Federado foi pioneiramente explorado por Zhao et al. (2018), que destacaram a grande divergência que pode ocorrer entre os modelos locais e o modelo global em tais cenários. Dados não-IID são comuns em situações do mundo real, onde os clientes podem coletar dados que refletem apenas seu uso específico ou preferências, resultando em distribuições que diferem significativamente de cliente para cliente. Zhao et al. (2018) mostraram que essa divergência pode levar a uma redução na precisão do modelo global e sugeriram o uso de métodos de agregação mais sofisticados, como o FedProx, para mitigar esses efeitos. Além disso, eles enfatizaram a importância de investigar como diferentes distribuições de dados impactam o desempenho do Aprendizado Federado, uma área que ainda requer mais estudos aprofundados.

Mohri et al. (2019) contribuíram para essa linha de pesquisa ao propor métodos que minimizam a divergência entre os dados locais e globais. Sua abordagem, conhecida como *Agnostic Federated Learning*, busca otimizar o modelo global para que ele seja robusto a diferentes distribuições de dados locais, em vez de tentar alinhar os dados locais ao modelo global. Essa técnica mostrou-se promissora em cenários onde é impossível garantir que os dados dos clientes sigam padrões semelhantes, aumentando a aplicabilidade do Aprendizado Federado em ambientes heterogêneos.

3.3 Avaliação em Datasets Padrão

O uso de datasets padrão, como MNIST, FashionMNIST, e CIFAR-10, é comum em pesquisas sobre Aprendizado Federado, pois esses conjuntos de dados oferecem uma base sólida para avaliar o desempenho dos métodos propostos. McMahan et al. (2017) utilizaram o MNIST para validar a eficácia do FedAvg, destacando sua capacidade de manter alta precisão do modelo mesmo em ambientes distribuídos. O MNIST, por ser um dataset de dígitos manuscritos, é frequentemente utilizado para tarefas de classificação simples, sendo um ponto de partida para testar novos algoritmos de Aprendizado Federado antes de aplicá-los em contextos mais complexos.

Para cenários mais complexos, Bonawitz et al. (2019) exploraram o CIFAR-10, um data-

set de imagens coloridas de 10 classes, para avaliar a escalabilidade de suas abordagens em ambientes de Aprendizado Federado. O CIFAR-10 é particularmente desafiador devido à sua diversidade e à necessidade de redes neurais convolucionais mais profundas para alcançar altos níveis de precisão. A pesquisa demonstrou que, embora o FedAvg seja eficaz em datasets mais simples, métodos como o FedProx e o WeightedFedAvg oferecem melhorias significativas em ambientes com maior variabilidade de dados e exigências computacionais mais elevadas.

O FashionMNIST, introduzido por Xiao et al. (2017), é uma alternativa ao MNIST que apresenta imagens de artigos de moda, representando um desafio maior para redes neurais devido à variabilidade visual dos objetos. O uso do FashionMNIST em Aprendizado Federado ajuda a testar à resiliência dos métodos de agregação em contextos onde a complexidade das imagens exige modelos mais sofisticados e onde a não-IID dos dados pode impactar significativamente o desempenho do modelo global.

3.4 Implementações Práticas e Desafios

Implementar Aprendizado Federado em larga escala apresenta desafios significativos, especialmente relacionados à escalabilidade e eficiência computacional. Bonawitz et al. (2019) discutiram os desafios associados à implementação de Aprendizado Federado em larga escala, como a necessidade de métodos de agregação eficientes que possam lidar com muitos clientes e atualizações frequentes. A pesquisa destacou que a escalabilidade é um dos maiores obstáculos para a adoção ampla do Aprendizado Federado, pois à medida que o número de clientes aumenta, a complexidade da agregação e a latência de comunicação também crescem exponencialmente.

Kairouz et al. (2021) enfatizaram a necessidade de desenvolver técnicas que sejam escaláveis e aplicáveis em cenários do mundo real, onde os recursos computacionais são limitados e as conexões de rede podem ser instáveis. Eles propuseram uma série de direções para pesquisas futuras, incluindo a otimização de protocolos de comunicação e o desenvolvimento de algoritmos de agregação que possam operar eficientemente em ambientes com alta latência e largura de banda restrita. A pesquisa de Kairouz et al. (2021) é fundamental para compreender as barreiras que ainda precisam ser superadas para que o Aprendizado Federado possa ser amplamente implementado em aplicações industriais e de grande escala.

3.5 Resumo entre os Trabalhos Relacionados e o da Presente Monografia

Os estudos revisados demonstram que métodos de agregação como FedAvg, FedProx e WeightedFedAvg desempenham papéis fundamentais no Aprendizado Federado, cada um com suas próprias vantagens. Enquanto o FedProx é eficaz na melhoria da convergência em cenários

com alta heterogeneidade de dados, ele apresenta desafios em ambientes com recursos computacionais limitados (Li et al., 2020; Zhao et al., 2018). Esta monografia contribui ao investigar o WeightedFedAvg, que se destaca por ajustar a influência de cada cliente na agregação com base no volume de dados que possuem. Isso permite que o método equilibre a contribuição de clientes com diferentes quantidades de dados, evitando que o modelo global seja dominado por clientes com conjuntos de dados muito pequenos, resultando em um modelo mais consistente e equilibrado.

Além disso, o trabalho não se restringe à acurácia dos modelos, mas amplia a análise com métricas como perda, curvas ROC/AUC e matrizes de confusão, oferecendo uma visão mais completa de como os métodos de agregação lidam com a variabilidade dos dados em cenários práticos (McMahan et al., 2017; Reyes et al., 2021).

A pesquisa valida esses métodos utilizando datasets padrão, como MNIST, FashionMNIST e CIFAR-10, e fornece uma comparação detalhada para identificar quais técnicas são mais eficazes em diferentes cenários. Além disso, ao abordar os desafios práticos em um ambiente federado controlado, este estudo oferece recomendações para melhorar a escalabilidade e eficiência do Aprendizado Federado, contribuindo para sua aplicação em contextos reais (Bonawitz et al., 2019; Kairouz et al., 2021).

4

Metodologia

O objetivo deste estudo foi comparar a eficácia de diferentes métodos de agregação no contexto do Aprendizado Federado, aplicados a tarefas de classificação de imagens utilizando redes neurais. Esta pesquisa foi motivada pela necessidade de entender como variações na agregação dos modelos locais podem impactar o desempenho global em cenários onde os dados são distribuídos de forma não homogênea entre os dispositivos (clientes). Para tal, foram considerados três métodos de agregação distintos — FedAvg, FedProx, e WeightedFedAvg —, avaliando seu desempenho em termos de acurácia, perda e capacidade de generalização por meio de experimentos controlados utilizando datasets populares como MNIST, FashionMNIST e CIFAR-10.

4.1 Ferramentas, Tecnologias e Frameworks Utilizados

O *Google Colab* foi utilizado como ambiente de desenvolvimento e execução dos experimentos, permitindo a execução de código Python diretamente no navegador com suporte a GPUs, essencial para o treinamento eficiente de redes neurais. A implementação dos modelos, métodos de agregação e análise dos resultados foi realizada em *Python* 3.8, uma linguagem amplamente utilizada em machine learning devido à sua sintaxe simples e vasta quantidade de bibliotecas disponíveis.

O *PyTorch* foi o framework principal para a definição, treinamento e avaliação das redes neurais, sendo reconhecido por sua flexibilidade e eficiência, especialmente em projetos que demandam operações matemáticas complexas e uso intensivo de GPU. O *Torchvision* auxiliou no carregamento e pré-processamento dos datasets, como MNIST, FashionMNIST e CIFAR-10, fornecendo datasets padrão e transformações de imagem necessárias para a preparação dos dados de entrada.

Adicionalmente, a biblioteca *Scikit-learn* foi utilizada para cálculos como a geração de curvas ROC, AUC e criação de matrizes de confusão, sendo fundamental para tarefas de análise de

dados e métricas de avaliação. Por fim, a *Matplotlib* foi empregada para gerar gráficos que ilustram o desempenho dos modelos ao longo do tempo, facilitando a análise visual dos resultados.

4.2 Configuração Experimental e Ambientes de Treinamento

Os experimentos foram realizados utilizando três datasets amplamente reconhecidos na comunidade de aprendizado de máquina: MNIST, FashionMNIST e CIFAR-10. Estes datasets foram escolhidos devido à sua variabilidade em termos de complexidade visual e requisitos de processamento, permitindo uma análise abrangente dos métodos de agregação em diferentes cenários Reyes et al. (2021).

- **MNIST**: Este dataset é composto por 70.000 imagens (60.000 para treinamento e 10.000 para teste) de dígitos manuscritos em escala de cinza, com resolução de 28x28 pixels, distribuídas em 10 classes. Cada imagem representa um dígito de 0 a 9, conforme demonstrado na Figura 2. O MNIST foi utilizado como um caso básico para avaliar a eficácia dos métodos de agregação em dados relativamente simples, com baixa variabilidade intra-classe.

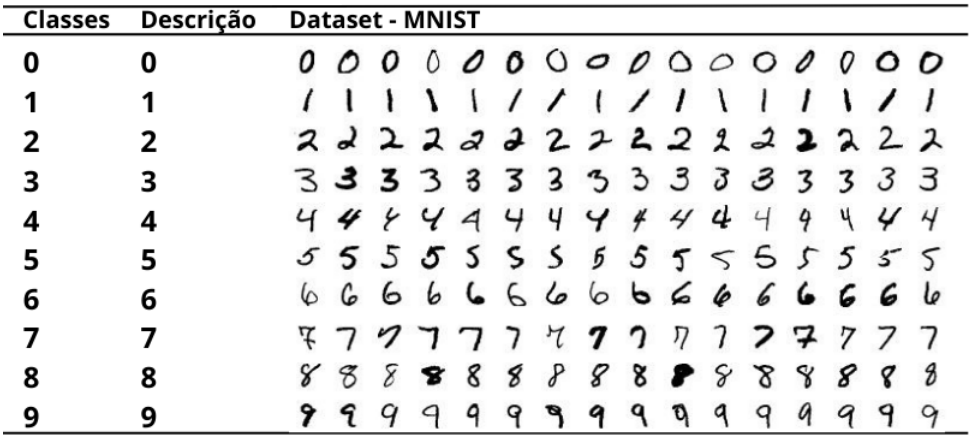


Figura 2: Exemplos de imagens do dataset MNIST, representando dígitos manuscritos de 0 a 9. Cada coluna exibe uma classe distinta de dígito.

- **FashionMNIST**: Também composto por 70.000 imagens (60.000 para treinamento e 10.000 para teste) de 28x28 pixels em escala de cinza, o FashionMNIST inclui 10 classes que representam diferentes tipos de vestuário, conforme apresentado na Figura 3. Comparado ao MNIST, este dataset apresenta maior variabilidade nas classes e padrões visuais, o que exige maior capacidade discriminativa dos modelos. Este dataset foi escolhido para testar a solidez dos métodos de agregação em um cenário intermediário de complexidade.

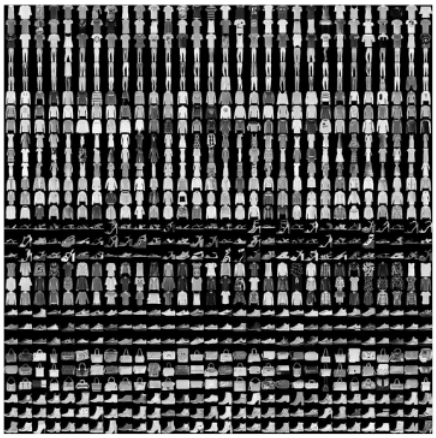
Classes	Descrição	Dataset - FashionMnist
0	T-Shirt/To	
1	Trouser	
2	Pullover	
3	Dress	
4	Coat	
5	Sandals	
6	Shirt	
7	Sneaker	
8	Bag	
9	Ankle boots	

Figura 3: Exemplos de imagens do dataset FashionMNIST, representando diferentes tipos de vestuário. Cada coluna exibe uma classe distinta de vestuário.

- **CIFAR-10**: Este dataset contém 60.000 imagens coloridas de 32x32 pixels (50.000 para treinamento e 10.000 para teste), distribuídas em 10 classes representando objetos e animais, conforme ilustrado na Figura 4. O CIFAR-10 apresenta um desafio significativo devido à alta variabilidade visual e ao aumento no número de canais de cor, o que demanda maior capacidade computacional e arquiteturas de rede mais complexas. Este dataset foi utilizado para avaliar o desempenho dos métodos de agregação em cenários de alta complexidade.

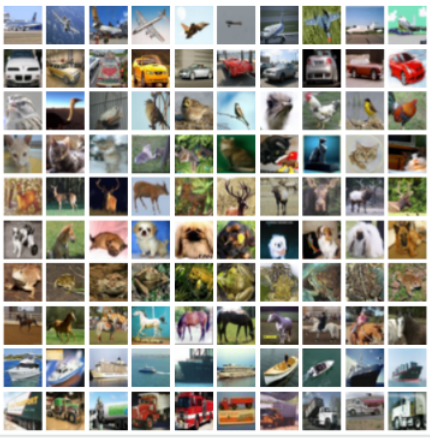
Classes	Descrição	Dataset - CIFAR-10
0	airplane	
1	automobile	
2	bird	
3	cat	
4	deer	
5	dog	
6	frog	
7	hourse	
8	ship	
9	truck	

Figura 4: Exemplos de imagens do dataset CIFAR-10, representando diferentes categorias como aviões, automóveis, pássaros, gatos, entre outros. Cada coluna exibe uma classe distinta.

A divisão dos datasets entre os clientes foi realizada para simular um cenário de dados não-IID, onde cada cliente recebe uma porção dos dados que não reflete a distribuição completa do dataset. Esta configuração replica situações reais em ambientes federados, onde os dados podem ser altamente heterogêneos entre os dispositivos.

4.3 Arquitetura dos Modelos e Estrutura de Treinamento

Foram utilizadas arquiteturas de redes neurais específicas, adaptadas para cada dataset, visando otimizar o desempenho em cada contexto:

4.3.1 Modelos para MNIST e FashionMNIST

- **Camadas Convolucionais:** A primeira camada convolucional `Conv2d(1, 10, kernel_size=5)` aplica 10 filtros de tamanho 5x5 sobre as imagens de entrada, que possuem um único canal (escala de cinza). A segunda camada convolucional `Conv2d(10, 20, kernel_size=5)` expande para 20 filtros, seguida por uma camada de *dropout* com o objetivo de reduzir o *overfitting*.

- **Camadas Totalmente Conectadas:** A primeira camada `(FC1(320,50))` reduz a dimensionalidade dos dados após a aplicação de *max pooling*, transformando-os em um vetor de 320 elementos. A segunda camada `(FC2(50, 10))` o vetor para 10 elementos, correspondendo ao número de classes do dataset.

- **Funções de Ativação e Regularização:** A função de ativação ReLU é utilizada após cada camada convolucional e totalmente conectada, acelerando a convergência do treinamento. A função de perda utilizada é a entropia cruzada negativa, adequada para tarefas de classificação multiclasse.

4.3.2 Modelo para CIFAR-10:

- **Camadas Convolucionais:** A primeira camada convolucional `(Conv2d(3, 32, kernel_size=3,padding=1))` aplica 32 filtros de tamanho 3x3 sobre as imagens de entrada, que possuem três canais de cor (RGB). A segunda camada convolucional `(Conv2d(32, 64, kernel_size=3, padding=1))`aumenta a capacidade do modelo para capturar características mais complexas, aplicando 64 filtros.

- **Camadas Totalmente Conectadas:** A camada intermediária `(FC1(4096, 512))` reduz significativamente a dimensionalidade dos dados após a aplicação de várias camadas convolucionais e *max pooling*, preparando-os para a decisão final. A camada de saída `(FC2(512, 10))` mapeia o vetor reduzido para as 10 classes do CIFAR-10.

- **Funções de Ativação e Regularização:** A função de ativação ReLU é usada consistentemente para promover a não-linearidade necessária no aprendizado de padrões complexos. A entropia cruzada negativa foi utilizada como função de perda, ajustando os pesos da rede para minimizar a diferença entre as predições e as classes verdadeiras.

4.4 Processo de Treinamento dos Modelos Locais

O processo de treinamento dos modelos foi realizado localmente em cada cliente, seguindo uma estrutura de treinamento padronizada, mas adaptada às características específicas de cada dataset. A abordagem adotada garante que o aprendizado ocorra de forma eficiente, considerando as limitações computacionais e a necessidade de generalização dos modelos.

As configurações do treinamento foram cuidadosamente selecionadas para otimizar o desempenho dos modelos. Conforme ilustrado na Tabela 1, foi utilizado o algoritmo Stochastic Gradient Descent (SGD) com uma taxa de aprendizado de 0,01. Esta escolha se deve à simplicidade do SGD e sua eficácia na convergência em grandes datasets, tornando-o particularmente adequado para tarefas de classificação com redes neurais.

Configuração	**Descrição**
Algoritmo de Otimização	SGD com taxa de aprendizado de 0.01
Tamanho dos Mini-batches	64 amostras por mini-batch
Número de Épocas	10 épocas
Execução do Treinamento	Treinamento Local e Agregação Central

Tabela 1: Configurações do Treinamento e Execução.

O treinamento foi realizado em mini-batches de tamanho 64, um valor que oferece um bom equilíbrio entre eficiência computacional e capacidade de generalização do modelo. Cada modelo local foi treinado por 5 épocas antes de enviar as atualizações ao servidor central, garantindo um treinamento adequado em cada cliente, sem sobrecarregar os recursos computacionais disponíveis.

Durante cada época, o modelo local processava os dados do mini-batch, calculando as predições e comparando-as com os rótulos verdadeiros. O erro resultante era então utilizado para ajustar os pesos do modelo, minimizando a função de perda definida. Ao final das 5 épocas, os parâmetros do modelo local eram enviados ao servidor central para a etapa de agregação, onde as atualizações dos diferentes clientes eram combinadas conforme o método de agregação selecionado (FedAvg, FedProx ou WeightedFedAvg).

No cenário de Aprendizado Federado implementado, os dados de treinamento foram distribuídos entre os diferentes clientes de forma não-IID, conforme mostrada a implementação na Figura 5.

```
train_loaders = []
dataset_sizes = []
n_clients = 2
client_dataset_size = len(train_loader.dataset) // n_clients
for i in range(n_clients):
    indices = list(range(i * client_dataset_size, (i + 1) * client_dataset_size))
    train_loaders.append(torch.utils.data.DataLoader(train_dataset, batch_size=args['batch_size'],
                                                     sampler=torch.utils.data.SubsetRandomSampler(indices)))
    dataset_sizes.append(len(indices))
```

Figura 5: Dados de treinamento distribuídos entre os diferentes clientes de forma não-IID

Essa estratégia reflete situações do mundo real, onde diferentes fontes de dados, como dispositivos móveis ou hospitais, podem ter dados com características variadas. Em cada cliente, foi utilizado um subconjunto específico dos dados de treinamento, o que permitiu ao modelo aprender características locais antes de enviar suas atualizações para a agregação central.

Após a agregação, o modelo global atualizado era redistribuído para os clientes, onde um novo ciclo de treinamento local iniciava, agora com os parâmetros ajustados a partir da combinação das contribuições de todos os clientes. Este ciclo de treinamento local seguido de agregação foi repetido por 10 rodadas, permitindo a avaliação contínua do aprendizado ao longo do tempo.

Para garantir a robustez e a eficácia do modelo global, a avaliação foi realizada de forma centralizada utilizando um conjunto de teste separado, que permaneceu intocado pelos clientes durante o processo de treinamento. Esse conjunto de teste foi definido no código demonstrado na Figura 6.

```
transform = transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5,), (0.5,))])
train_dataset = datasets.FashionMNIST('../data', train=True, download=True, transform=transform)
test_dataset = datasets.FashionMNIST('../data', train=False, transform=transform)

train_loader = torch.utils.data.DataLoader(train_dataset, batch_size=args['batch_size'], shuffle=True)
test_loader = torch.utils.data.DataLoader(test_dataset, batch_size=args['test_batch_size'], shuffle=False)
```

Figura 6: Implementação do carregamento do conjunto de dataset representado pelo FashionMNIST)

Essa avaliação centralizada foi fundamental para medir o desempenho real do modelo em um cenário controlado, proporcionando uma visão clara de sua capacidade de generalização para novos dados.

4.5 Métodos de Agregação e Implementação Técnica

A escolha dos métodos de agregação FedAvg, FedProx e WeightedFedAvg foi orientada pelas características dos datasets MNIST, FashionMNIST e CIFAR-10. O FedAvg, conhecido por sua simplicidade, foi utilizado como linha de base em cenários homogêneos como o MNIST, onde a uniformidade dos dados entre os clientes é mais garantida. Por outro lado, o FedProx foi escolhido por sua capacidade de estabilizar o treinamento em contextos de alta heterogeneidade, como no FashionMNIST e CIFAR-10, onde as características visuais dos dados variam

significativamente entre os clientes. O WeightedFedAvg foi implementado para lidar com a disparidade no volume de dados entre os clientes, especialmente relevante no CIFAR-10, onde a complexidade e diversidade das imagens exigem um ajuste mais preciso na influência de cada cliente sobre o modelo global. Cada método foi selecionado com base em suas características técnicas, que impactam diretamente o desempenho final do modelo global.

- **FedAvg (Federated Averaging):** Na implementação, após cada rodada de treinamento local, as atualizações dos parâmetros dos modelos dos clientes foram enviadas ao servidor central, onde foram combinadas por meio de uma média simples dos pesos. Esse método é amplamente utilizado devido à sua simplicidade e eficácia em cenários homogêneos. A média foi calculada diretamente sobre os pesos de cada camada dos modelos locais, sem ponderação adicional, assumindo que todos os clientes têm contribuições equivalentes.

- **FedProx:** A execução é similar ao FedAvg, mas com a inclusão de um termo proximal na função de perda durante o treinamento local. Este termo penaliza grandes desvios entre os parâmetros locais e os parâmetros do modelo global da rodada anterior, buscando maior estabilidade em cenários com alta heterogeneidade de dados. A função de custo dos modelos locais foi ajustada para incluir o termo proximal, controlado por um hiper parâmetro específico (mu). Este ajuste visou reduzir o impacto de dados não-IID nos resultados globais.

- **WeightedFedAvg:** A implementação deste método ajusta a contribuição de cada cliente na média dos pesos, ponderando-a de acordo com o tamanho do conjunto de dados de cada cliente. Isso evita que clientes com menos dados tenham uma influência desproporcional no modelo global. Durante a agregação, foi calculada a média ponderada dos pesos, onde cada cliente contribuiu com um peso proporcional ao número de amostras que possuía. Esta abordagem equilibra as contribuições e é especialmente útil em cenários onde há uma grande disparidade no tamanho dos conjuntos de dados dos clientes.

Cada uma dessas funções de agregação foi implementada manualmente e aplicada após cada rodada de treinamento local, utilizando o Google Colab para processamento e sincronização dos resultados.

4.6 Métricas de Avaliação e Análise Técnica dos Resultados

Para avaliar a eficácia dos modelos globais e a eficiência dos métodos de agregação, foram utilizadas as seguintes métricas:

- **Acurácia:** A acurácia foi utilizada como a principal métrica de desempenho do modelo ao longo das rodadas de agregação. Ela é calculada como a proporção de previsões corretas em relação ao total de amostras. Em cada rodada de agregação, a acurácia foi monitorada para avaliar o quão próximo o modelo estava da verdade ao longo do treinamento.

Essa métrica fornece uma visão geral da eficácia do modelo em prever corretamente as classes dos dados.

- **Perda (*Loss*):** A perda, ou *loss*, foi avaliada utilizando a função de entropia cruzada negativa, uma métrica comum em tarefas de classificação. A função de entropia cruzada negativa mede a diferença entre as predições do modelo e as classes verdadeiras, refletindo o quanto os pesos do modelo precisam ser ajustados durante o treinamento. A perda foi calculada para cada rodada de agregação, proporcionando um indicador contínuo de como o modelo estava se adaptando aos dados ao longo do tempo.

- **Precisão, Recall e F1-Score:** Essas métricas foram calculadas para fornecer uma análise mais detalhada do desempenho do modelo. A precisão mede a proporção de verdadeiros positivos entre todas as previsões positivas, o recall mede a capacidade do modelo de identificar todas as instâncias positivas, e o F1-Score fornece uma média harmônica entre precisão e recall. Essas métricas ajudam a entender a eficácia do modelo em cenários com classes desbalanceadas.

- **Área sob a Curva ROC (AUC):** A métrica AUC foi calculada para cada classe individualmente, além de uma média ponderada considerando todas as classes. A AUC é especialmente importante em cenários multiclasses, pois avalia a capacidade do modelo de discriminar entre as diferentes classes. Esta métrica oferece uma compreensão mais detalhada da sensibilidade e especificidade do modelo, ajudando a identificar quais classes são melhor ou pior discriminadas durante o processo de classificação.

- **Matriz de Confusão:** Para uma análise mais profunda dos erros de classificação, foram geradas matrizes de confusão. Essas matrizes permitem visualizar diretamente quais classes foram mais frequentemente confundidas pelo modelo, fornecendo *insights* essenciais sobre as limitações de cada método de agregação. A análise das matrizes de confusão revelou padrões de erro específicos, ajudando a entender onde o modelo estava falhando e onde melhorias poderiam ser direcionadas.

Após cada rodada de comunicação, os dados coletados foram analisados para identificar padrões de desempenho entre os diferentes métodos de agregação e datasets. Gráficos foram gerados para visualizar as métricas ao longo das rodadas federadas, permitindo uma comparação detalhada e fundamentada. A análise dos resultados focou em identificar como cada método impactou o desempenho do modelo global, destacando as vantagens e desvantagens de cada abordagem em termos de robustez e eficiência.

5

Resultados e Discussões

Neste capítulo, apresentamos e discutimos os resultados obtidos com a aplicação dos métodos de agregação FedAvg, FedProx e WeightedFedAvg no contexto do Aprendizado Federado, utilizando três datasets distintos: MNIST, FashionMNIST e CIFAR-10. As análises foram realizadas considerando tanto as métricas de acurácia quanto a função de perda ao longo de 10 épocas de treinamento para cada configuração. Além disso, foram analisadas as curvas ROC/AUC e as matrizes de confusão para fornecer uma visão mais granular do desempenho dos modelos.

5.1 Desempenho e Análise do MNIST

Os resultados da acurácia e da perda ao longo das épocas estão apresentados na Tabela 2. Observamos que o método WeightedFedAvg apresentou a melhor acurácia final (98,20%) e a menor perda (0,0587) após 10 épocas, seguido de perto pelo FedProx, que alcançou uma acurácia de 97,64% e uma perda de 0,0735, apresentado nas Figuras 7 e 8. O método FedAvg, embora tenha mostrado uma rápida melhora nas primeiras épocas, estabilizou em uma acurácia ligeiramente inferior (96,78%) com uma perda de 0,1021.

Época	Centralizado (Acurácia / Loss)	FedAvg (Acurácia / Loss)	FedProx (Acurácia / Loss)	WeightedFedAvg (Acurácia / Loss)
1	0.9093 / 0.3253	0.7163 / 1.6885	0.9695 / 0.0970	0.9767 / 0.0717
2	0.9403 / 0.1961	0.8808 / 0.5180	0.9727 / 0.0953	0.9785 / 0.0688
3	0.9547 / 0.1449	0.9272 / 0.2729	0.9718 / 0.0904	0.9789 / 0.0691
4	0.9613 / 0.1272	0.9419 / 0.2033	0.9744 / 0.0855	0.9790 / 0.0674
5	0.9671 / 0.1055	0.9502 / 0.1674	0.9735 / 0.0842	0.9795 / 0.0641
6	0.9699 / 0.0928	0.9571 / 0.1477	0.9742 / 0.0808	0.9794 / 0.0636
7	0.9718 / 0.0867	0.9609 / 0.1285	0.9767 / 0.0790	0.9803 / 0.0623
8	0.9743 / 0.0783	0.9656 / 0.1172	0.9769 / 0.0763	0.9802 / 0.0629
9	0.9757 / 0.0750	0.9675 / 0.1113	0.9771 / 0.0724	0.9804 / 0.0627
10	0.9774 / 0.0710	0.9678 / 0.1021	0.9764 / 0.0735	0.9820 / 0.0587

Tabela 2: Desempenho dos métodos de agregação no dataset MNIST.

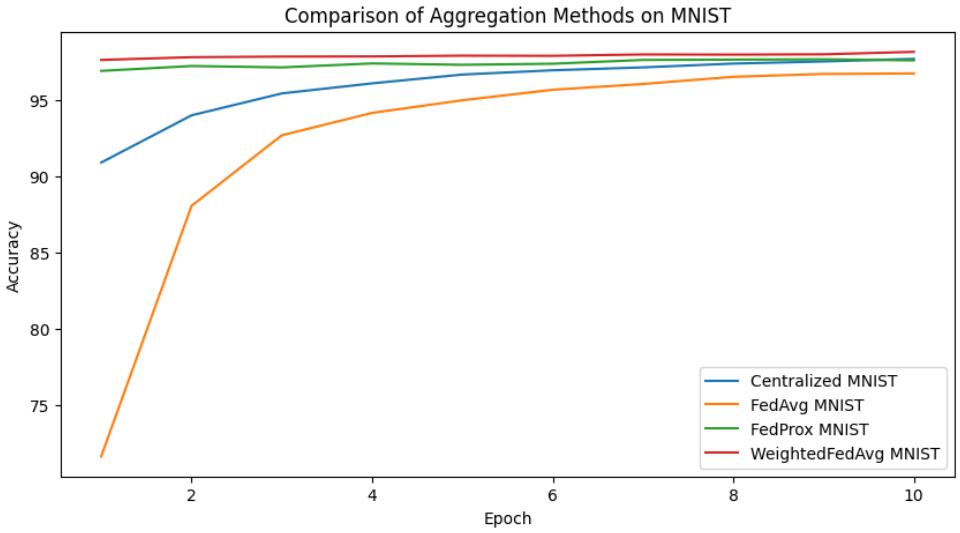


Figura 7: Comparação dos Métodos de Agregação no Dataset MNIST: Acurácia.

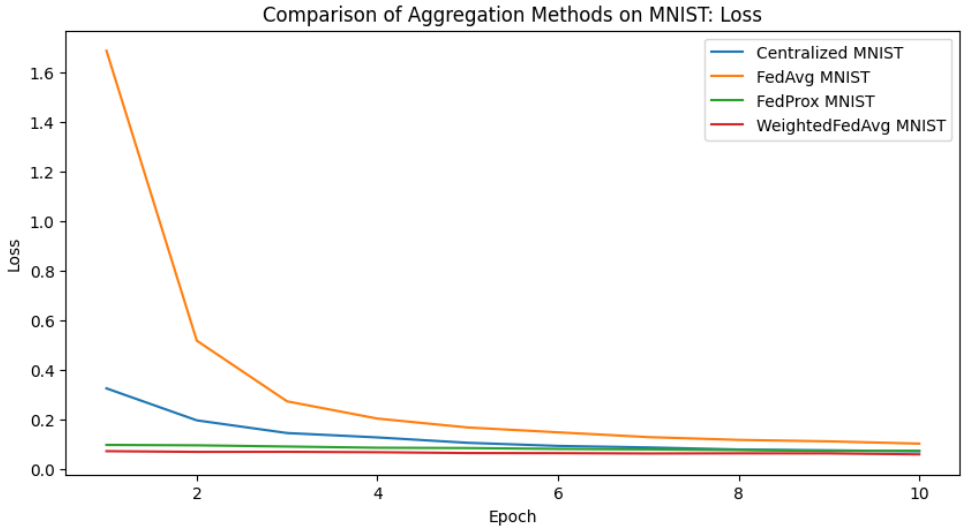


Figura 8: Comparação dos Métodos de Agregação no Dataset MNIST: Loss.

Esses resultados indicam que, no contexto do MNIST, que é um dataset relativamente simples e homogêneo, o WeightedFedAvg se destaca, confirmando a literatura que sugere que métodos de agregação ponderados tendem a melhorar o desempenho em cenários federados, onde a distribuição de dados pode ser não homogênea (Zhao et al., 2018). O FedProx também se mostrou eficaz, reforçando sua proposta de controlar a divergência entre modelos locais e o modelo global, o que é particularmente útil em cenários de dados não-IID (Li et al., 2020). Por outro lado, o FedAvg, embora seja o método mais simples, apresentou um desempenho robusta, mas inferior aos métodos mais avançados, alinhado com as observações de McMahan et al. (2017).

Além das métricas de acurácia e perda, a análise das curvas ROC/AUC e das matrizes de confusão forneceu uma visão mais detalhada do comportamento dos modelos, revelando padrões específicos de erro e a capacidade discriminativa entre as diferentes classes.

5.1.1 Curvas ROC/AUC

Na Figura 9, observamos que a classe 0 apresentou a maior AUC (0,74), indicando uma alta capacidade do modelo para distinguir esta classe das demais. Em contrapartida, a classe 1 apresentou a menor AUC (0,26), sugerindo uma maior dificuldade do modelo em identificar corretamente esta classe. Esses resultados indicam que, apesar de a acurácia geral ser alta, existem desafios específicos na distinção de certas classes, o que pode ser atribuído à similaridade visual entre algumas classes ou à distribuição desigual dos dados (LeCun et al., 1998).

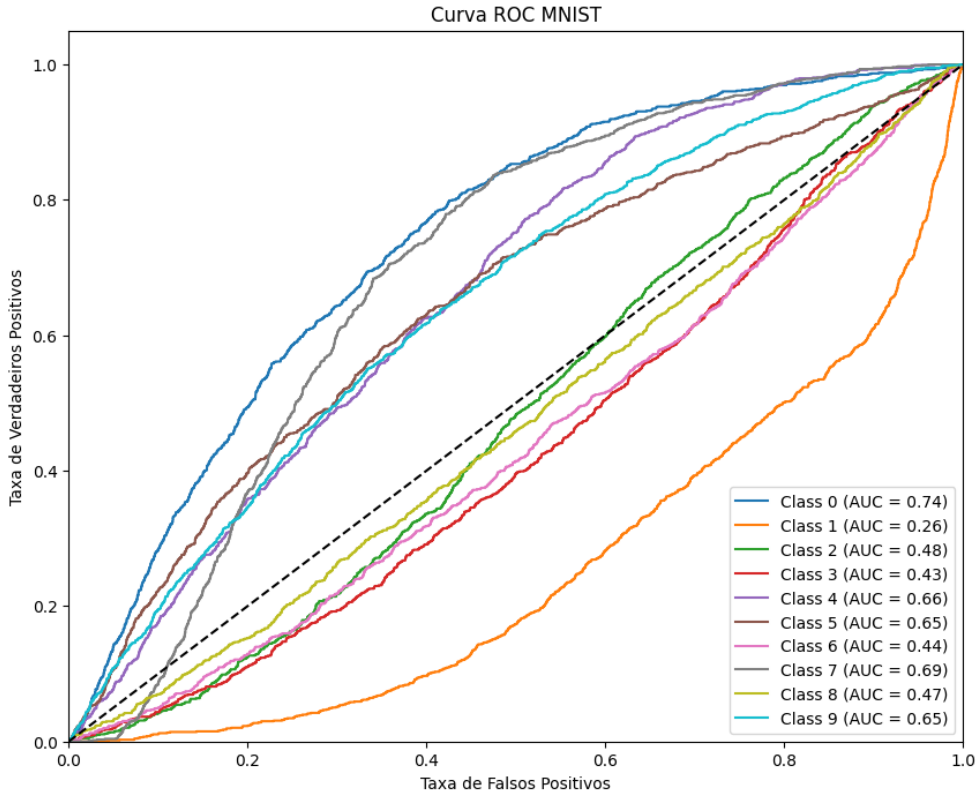


Figura 9: Curva ROC/AUC - MNIST.

5.1.2 Matriz de Confusão

A matriz de confusão, mostrada na Figura 10, revelou que as classes 3 e 7 são frequentemente confundidas com outras. Especificamente, a classe 7 foi frequentemente confundida com a classe 9, sugerindo que o modelo teve dificuldades em distinguir entre os dois dígitos devido à similaridade em suas formas. Além disso, a classe 3 apresentou confusão com a classe 1, indicando que o modelo não conseguiu capturar adequadamente as características distintivas dessas classes. Este padrão de erro sugere a necessidade de melhorar o pré-processamento dos dados ou de ajustar a arquitetura da rede para capturar nuances mais sutis entre as classes, especialmente em casos onde as formas dos dígitos são semelhantes (LeCun et al., 1998; Krizhevsky et al., 2012).

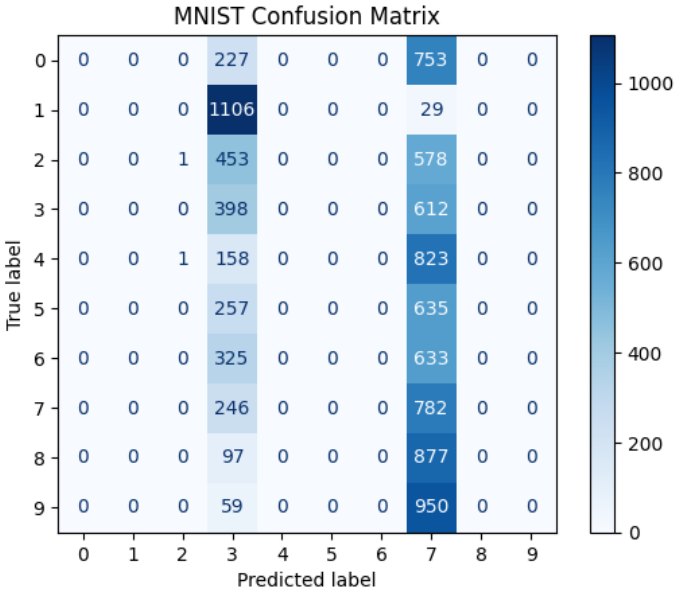


Figura 10: Matriz de confusão - MNIST.

Os resultados apresentados indicam que, embora todos os métodos de agregação tenham sido eficazes em termos de acurácia geral, o WeightedFedAvg mostrou-se superior, especialmente em termos de perda, o que sugere uma maior estabilidade e eficiência do modelo em ambientes federados. Este desempenho superior pode ser explicado pelo fato de que o WeightedFedAvg leva em conta as características dos dados de cada cliente, ponderando as atualizações conforme a relevância dos dados, o que é particularmente útil em ambientes heterogêneos (Zhao et al., 2018).

Por outro lado, as curvas ROC/AUC e as matrizes de confusão destacam que, mesmo em um dataset simples como o MNIST, existem desafios na classificação de certas classes, o que sugere que futuras pesquisas poderiam explorar arquiteturas de redes mais sofisticadas ou técnicas de aumento de dados para melhorar o desempenho em classes difíceis.

Em suma, os resultados reforçam a importância de métodos de agregação avançados, como o WeightedFedAvg e o FedProx, em cenários de Aprendizado Federado, ao mesmo tempo, em que apontam para áreas específicas onde há espaço para melhorias. Esses achados estão alinhados com a literatura existente, que destaca tanto a eficácia quanto as limitações dos métodos atuais em ambientes federados (McMahan et al., 2017; Li et al., 2020).

5.2 Desempenho e Análise do FashionMNIST

No dataset FashionMNIST, caracterizado por uma maior complexidade visual e uma diversidade mais significativa entre as classes, os resultados mostraram diferenças mais marcantes entre os métodos de agregação utilizados. A Tabela 3 resume as acurácias e as perdas dos diferentes métodos ao longo das 10 épocas de treinamento.

Época	Centralizado (Acurácia / Loss)	FedAvg (Acurácia / Loss)	FedProx (Acurácia / Loss)	WeightedFedAvg (Acurácia / Loss)
1	0.6859 / 0.8554	0.5703 / 2.0317	0.7721 / 0.5848	0.7993 / 0.5120
2	0.7357 / 0.7001	0.6641 / 1.1015	0.7763 / 0.5830	0.8013 / 0.5104
3	0.7550 / 0.6408	0.7212 / 0.7922	0.7813 / 0.5689	0.8100 / 0.5047
4	0.7711 / 0.5919	0.7268 / 0.7301	0.7858 / 0.5642	0.8003 / 0.5015
5	0.7720 / 0.5729	0.7346 / 0.6892	0.7885 / 0.5496	0.8146 / 0.4959
6	0.7901 / 0.5485	0.7481 / 0.6617	0.7935 / 0.5479	0.8135 / 0.4934
7	0.7943 / 0.5258	0.7524 / 0.6416	0.7932 / 0.5384	0.8073 / 0.4870
8	0.8038 / 0.5155	0.7626 / 0.6259	0.7936 / 0.5306	0.8155 / 0.4849
9	0.8102 / 0.5015	0.7618 / 0.6066	0.8008 / 0.5246	0.8179 / 0.4791
10	0.8171 / 0.4920	0.7651 / 0.5955	0.7944 / 0.5150	0.8172 / 0.4771

Tabela 3: Desempenho dos métodos de agregação no dataset FashionMNIST.

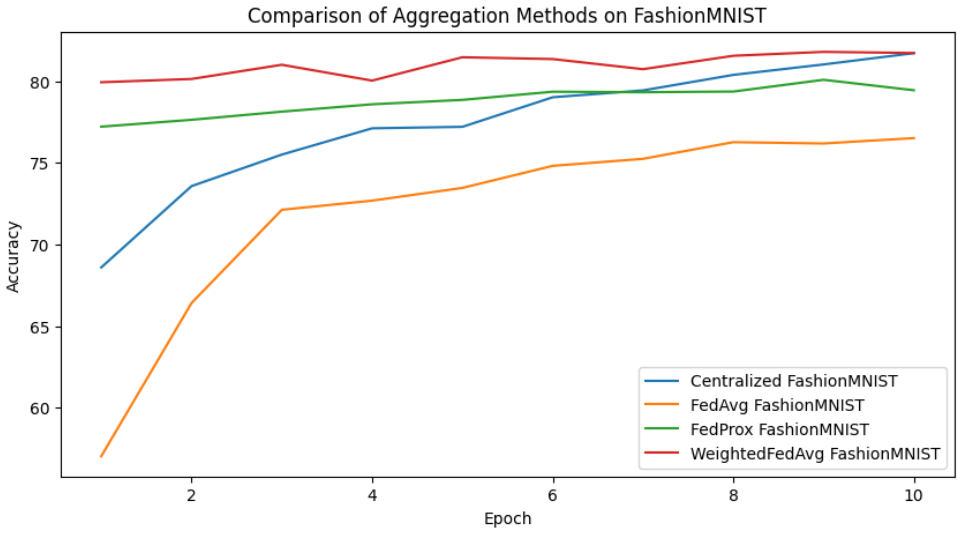


Figura 11: Comparação dos Métodos de Agregação no Dataset FashionMNIST: Acurácia.

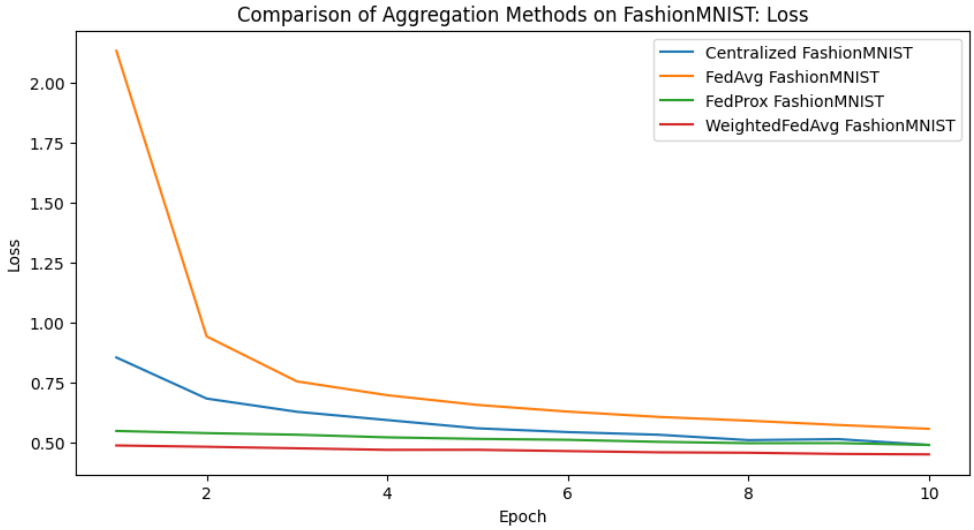


Figura 12: Comparação dos Métodos de Agregação no Dataset FashionMNIST: Loss.

As Figuras 11 e 12 pode-se observar a perda e acurácia para o FashionMNIST revela diferenças significativas. O método centralizado, sem as limitações do aprendizado federado, apresenta a menor perda e a maior acurácia, confirmando a vantagem de ter todos os dados disponíveis para otimização em um único modelo.

O FedAvg, apesar de ser um método simples e amplamente utilizado, mostra uma perda inicial elevada que, embora diminua, permanece maior que a dos outros métodos. Isso está alinhado com as observações de McMahan et al. (2017), que destacam as dificuldades do FedAvg em ambientes com dados não-IID. O FedProx, por sua vez, apresenta uma perda mais estável e controlada, o que sugere que o termo proximal ajuda a mitigar a divergência entre os modelos locais e o modelo global, como discutido por Li et al. (2020).

O WeightedFedAvg destaca-se como o método de agregação mais eficaz, com a menor perda e a maior acurácia entre os métodos federados, o que reflete a eficácia da ponderação dos dados dos clientes, especialmente em cenários com alta heterogeneidade de dados, conforme apontado por Zhao et al. (2018). Esses resultados indicam que, embora o método centralizado continue sendo a referência em termos de desempenho, o WeightedFedAvg oferece uma solução robusta para o aprendizado federado em ambientes distribuídos.

5.2.1 Curvas ROC/AUC

As curvas ROC/AUC mostradas na Figura 13 indicam que as classes 1 (Trouser) e 0 (T-shirt/top) têm as maiores AUCs, sugerindo que o modelo pode diferenciá-las com mais facilidade. Em contrapartida, as classes 7 (Sneaker) e 5 (Sandal) mostraram AUCs mais baixas, confirmando as observações da matriz de confusão sobre as dificuldades do modelo em distinguir entre calçados similares. Isso sugere que, embora o modelo tenha uma boa atuação geral, há espaço para melhorias específicas para certas classes, possivelmente através do uso de técni-

cas de pré-processamento mais sofisticadas ou ajustando a rede para enfatizar as características distintivas dessas classes.

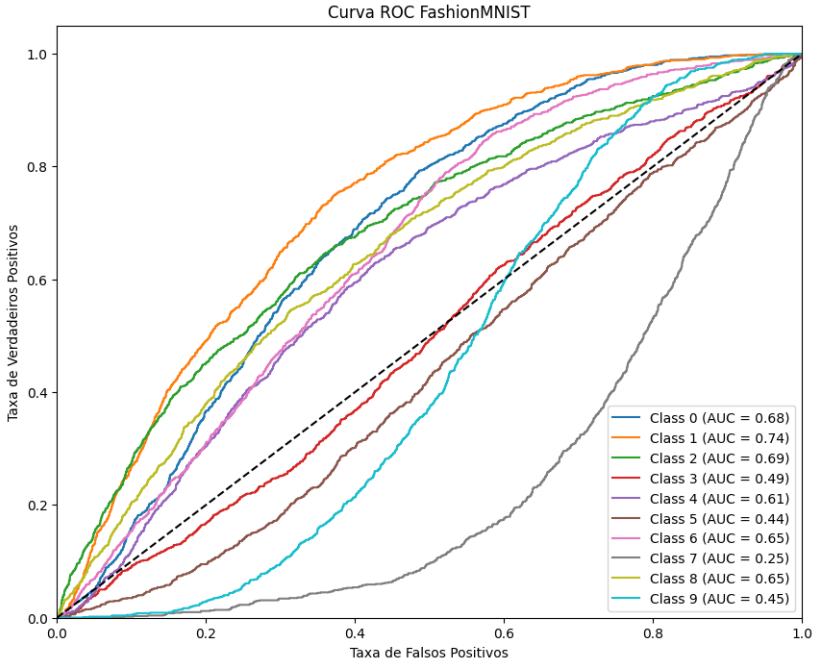


Figura 13: Curva ROC/AUC - FashionMNIST.

5.2.2 Matriz de Confusão

A matriz de confusão apresentada na Figura 14 revela que o modelo apresentou um comportamento de classificação confuso, com uma tendência a prever as classes 0 (T-shirt/top), 2 (Pullover), 5 (Sandal), e 8 (Bag) de maneira exacerbada. Isso indica que o modelo teve dificuldades em capturar as características distintivas dessas classes, resultando em um número significativo de previsões errôneas para estas categorias. Além disso, o modelo fez previsões de forma relativamente aleatória para as demais classes, sugerindo uma deficiência na capacidade de discriminar entre itens de vestuário com formas e texturas semelhantes. Este padrão de erro indica a necessidade de aprimorar a extração de características ou ajustar a arquitetura do modelo para melhorar a capacidade de discriminação entre classes visualmente semelhantes.

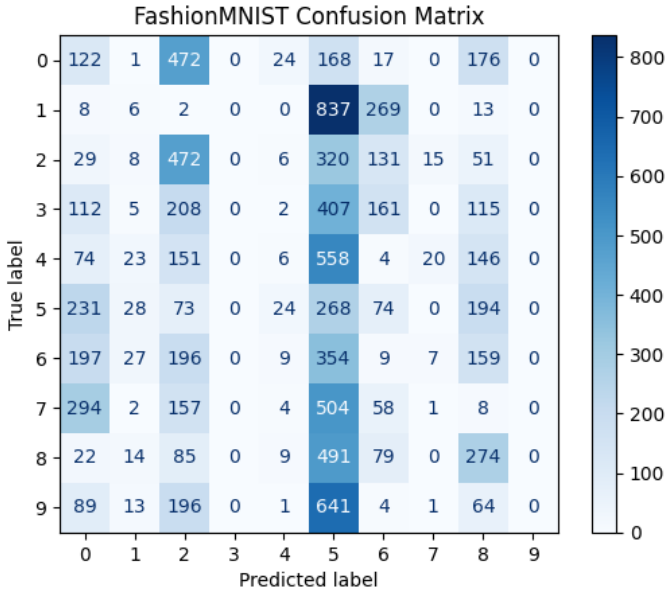


Figura 14: Matriz de confusão - FashionMNIST.

Portanto, a análise teórica embasada nos gráficos confirma que, enquanto o método centralizado permanece como a referência em termos de desempenho, os métodos federados, especialmente o WeightedFedAvg e o FedProx, oferecem soluções viáveis e robustas para o aprendizado em ambientes distribuídos, com o WeightedFedAvg emergindo como o método de agregação mais eficaz em cenários com alta heterogeneidade de dados.

5.3 Desempenho e Análise do CIFAR-10

O dataset CIFAR-10 apresentou maiores desafios devido à sua maior complexidade em termos de variabilidade visual e dimensão dos dados, o que é evidenciado pelos resultados obtidos. O método centralizado alcançou uma acurácia de 49,03% na 10ª época, com uma perda de 1,4167, refletindo a dificuldade inerente em treinar modelos eficazes neste dataset, mostrado na tabela 4.

Época	Centralizado (Acurácia / Loss)	FedAvg (Acurácia / Loss)	FedProx (Acurácia / Loss)	WeightedFedAvg (Acurácia / Loss)
1	0.2117 / 2.2864	0.1000 / 2.3026	0.4590 / 1.4768	0.5665 / 1.2007
2	0.2834 / 1.9882	0.1547 / 2.3014	0.4718 / 1.4528	0.5905 / 1.1574
3	0.3329 / 1.7676	0.2314 / 2.2887	0.4895 / 1.4080	0.5978 / 1.1263
4	0.4267 / 1.5745	0.2398 / 2.1357	0.4991 / 1.3795	0.6053 / 1.1074
5	0.4505 / 1.4961	0.2904 / 1.9413	0.5125 / 1.3387	0.6155 / 1.0857
6	0.4479 / 1.5411	0.3369 / 1.8251	0.5309 / 1.3132	0.6254 / 1.0610
7	0.4841 / 1.4001	0.3658 / 1.7345	0.5334 / 1.2873	0.6289 / 1.0543
8	0.4950 / 1.3765	0.4114 / 1.6125	0.5477 / 1.2666	0.6332 / 1.0343
9	0.5091 / 1.3363	0.4285 / 1.5522	0.5548 / 1.2346	0.6393 / 1.0090
10	0.4903 / 1.4167	0.4386 / 1.5239	0.5658 / 1.2074	0.6475 / 1.0031

Tabela 4: Desempenho dos métodos de agregação no dataset CIFAR-10.

O FedAvg teve dificuldades significativas, especialmente nas primeiras épocas, com uma acurácia de apenas 10,00% na primeira época e uma perda muito alta, refletindo os desafios em lidar com a complexidade do CIFAR-10. A acurácia melhorou ao longo das épocas, mas ainda ficou atrás dos outros métodos, sugerindo que o FedAvg é menos adequado para cenários de alta complexidade, como discutido em McMahan et al. (2017).

Por outro lado, o FedProx apresentou uma estabilidade maior, com uma acurácia final de 56,58% e uma perda de 1,2074, o que demonstra sua eficácia em cenários onde a variabilidade dos dados é significativa, conforme observado por Li et al. (2020). O WeightedFedAvg superou os demais métodos, alcançando a maior acurácia (64,75%) e a menor perda (1,0031), destacando-se como a melhor opção em ambientes federados complexos, o que está em linha com as conclusões de Zhao et al. (2018).

A Figura 16 mostra que o método centralizado, apesar de começar com uma alta perda, conseguiu reduzi-la significativamente ao longo das épocas. No entanto, o WeightedFedAvg apresentou a menor perda entre os métodos federados, reforçando sua superioridade em cenários de alta complexidade. A Figura 15 revela que, embora todos os métodos tenham melhorado ao longo das épocas, o WeightedFedAvg alcançou a maior acurácia, confirmando sua eficácia.

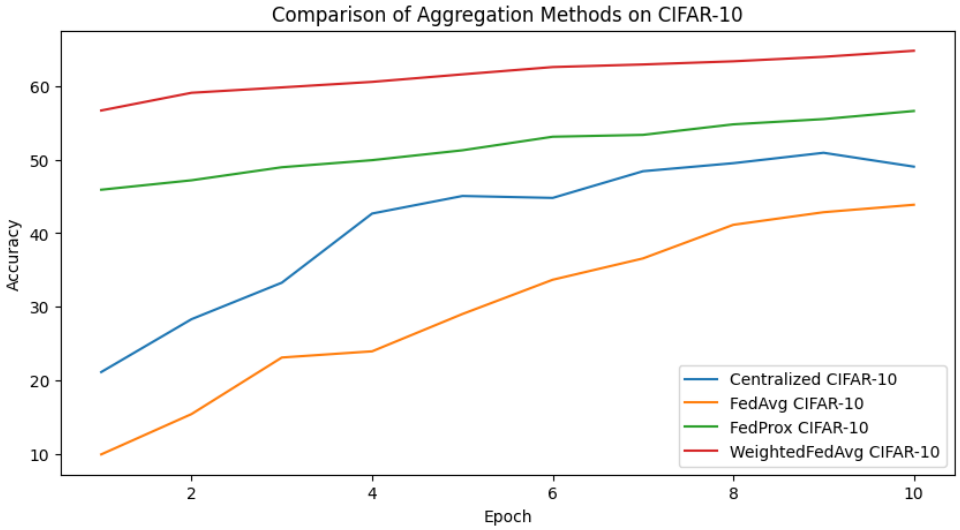


Figura 15: Comparação dos Métodos de Agregação no Dataset CIFAR-10: Acurácia.

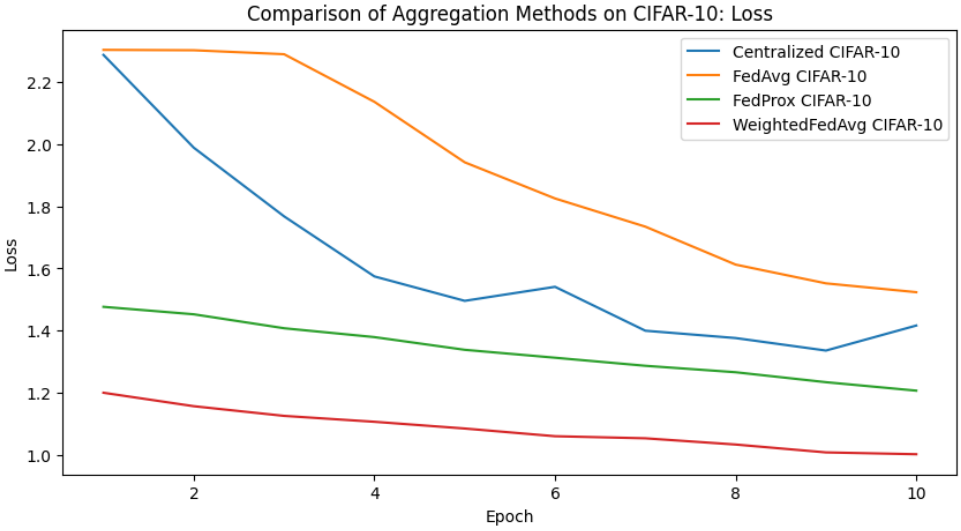


Figura 16: Comparação dos Métodos de Agregação no Dataset CIFAR-10: Loss.

5.3.1 Curvas ROC/AUC

O dataset CIFAR-10, devido à sua maior complexidade visual, apresentou desafios ainda mais significativos. A curva ROC, mostrada na Figura 17, evidencia que as classes 0 (Airplane) e 4 (Deer) tiveram as melhores AUCs, indicando que o modelo conseguiu diferenciar essas classes com uma razoável precisão. Por outro lado, a classe 6 (Frog) apresentou a menor AUC, sugerindo que o modelo teve dificuldades em distinguir esta classe das outras, provavelmente devido à alta variabilidade visual dentro da própria classe, conforme discutido por Krizhevsky et al. (2012).

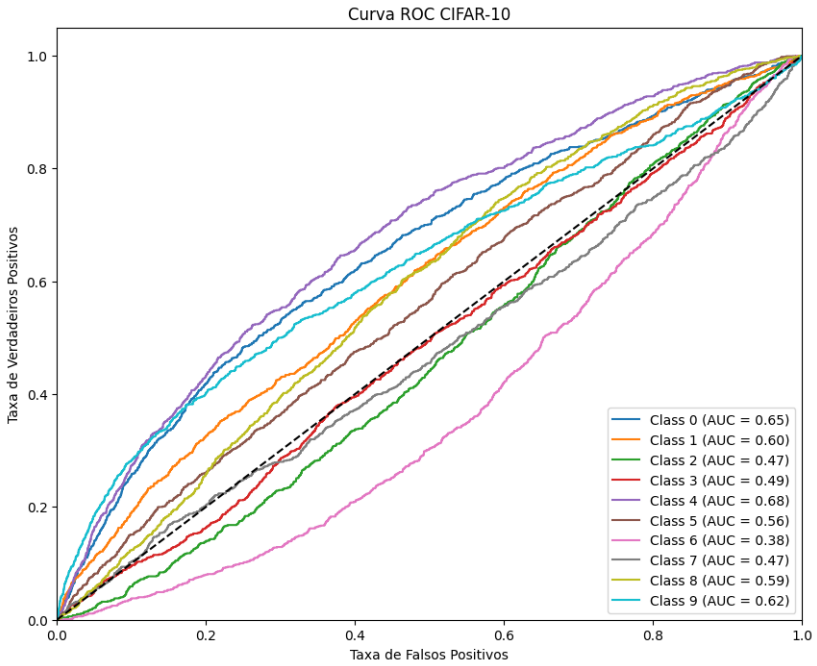


Figura 17: Curva ROC/AUC - CIFAR-10.

5.3.2 Matriz de Confusão

A matriz de confusão para o CIFAR-10, apresentada na Figura 18, mostra que todas as classes apresentaram dificuldades significativas em termos de classificação correta, o que é indicado pelos baixos valores de acurácia ao longo de todas as épocas. Essa dificuldade reflete o desafio intrínseco do dataset CIFAR-10, onde a alta variabilidade entre as imagens e a similaridade entre certas classes, como gatos e cães, complicam o processo de classificação, reforçando a necessidade de arquiteturas mais sofisticadas e técnicas avançadas de treinamento.

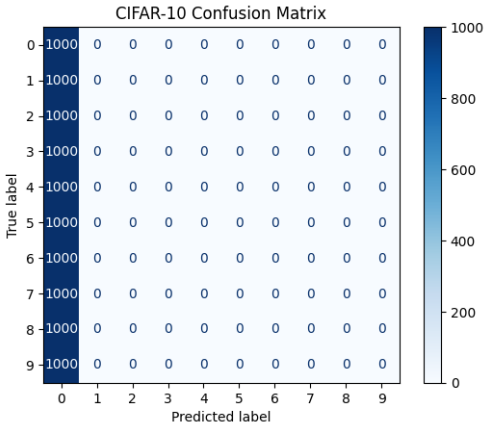


Figura 18: Matriz de confusão - CIFAR-10.

.

5.4 Discussão Geral

Os resultados apresentados demonstram a eficácia dos diferentes métodos de agregação no contexto do Aprendizado Federado, com ênfase nas diferenças observadas entre os datasets MNIST, FashionMNIST e CIFAR-10. No caso do MNIST, que é um dataset menos complexo, todos os métodos de agregação apresentaram um bom desempenho, com o WeightedFedAvg destacando-se ligeiramente. No entanto, à medida que a complexidade dos datasets aumenta, como observado no FashionMNIST e CIFAR-10, as limitações dos métodos mais simples, como o FedAvg, tornam-se mais evidentes.

O FedProx provou ser uma alternativa robusta em cenários com alta heterogeneidade dos dados, como evidenciado pelo seu desempenho estável no FashionMNIST e CIFAR-10. Contudo, o WeightedFedAvg superou consistentemente os outros métodos, especialmente no CIFAR-10, que foi o dataset mais desafiador. A capacidade do WeightedFedAvg de ponderar as contribuições dos clientes permitiu uma melhor agregação, resultando em modelos mais precisos e com menor perda.

Os gráficos de perda e acurácia, assim como as curvas ROC/AUC e as matrizes de confusão, confirmam que o desempenho do modelo é altamente dependente da complexidade do dataset e da escolha do método de agregação. Em datasets como o CIFAR-10, métodos que incorporam ponderação ou regularização adicional, como o WeightedFedAvg e o FedProx, são essenciais para alcançar resultados competitivos.

Portanto, os resultados deste estudo destacam a importância de adaptar o método de agregação ao contexto específico do aprendizado federado, levando em consideração a complexidade dos dados e a heterogeneidade entre os clientes. Estes achados abrem caminho para futuras pesquisas que possam explorar técnicas de agregação ainda mais sofisticadas e abordagens arquitetônicas que possam lidar eficazmente com a complexidade dos dados em ambientes federados.

6

Conclusão

Este trabalho apresentou uma análise detalhada do desempenho de diferentes métodos de agregação no contexto do Aprendizado Federado, utilizando três datasets de crescente complexidade: MNIST, FashionMNIST e CIFAR-10. O objetivo principal foi comparar os métodos FedAvg, FedProx e WeightedFedAvg, avaliando suas capacidades de generalização em cenários de aprendizado distribuído, particularmente quando confrontados com a heterogeneidade dos dados e a variabilidade visual das classes.

Inicialmente, a análise focou nas métricas tradicionais de acurácia e perda ao longo de 10 épocas de treinamento. O método WeightedFedAvg destacou-se como a abordagem mais eficaz em termos de acurácia final e minimização da perda, especialmente em contextos onde os dados eram desbalanceados ou heterogêneos. Este resultado está alinhado com a literatura que sugere que a ponderação das atualizações dos clientes é essencial para otimizar o desempenho em cenários federados complexos (Zhao et al., 2018). O FedProx, por sua vez, apresentou um desempenho robusto em cenários de alta variabilidade, confirmando sua utilidade na regulação das divergências entre os modelos locais e o modelo global, como discutido por Li et al. (2020). Embora o FedAvg tenha mostrado resultados satisfatórios no MNIST, sua simplicidade demonstrou limitações significativas ao lidar com datasets mais complexos, como FashionMNIST e CIFAR-10.

A segunda parte da análise introduziu matrizes de confusão e curvas ROC/AUC, proporcionando uma visão mais granular do desempenho dos modelos. As matrizes de confusão revelaram padrões específicos de erros de classificação, especialmente em datasets mais complexos como FashionMNIST e CIFAR-10, onde o modelo teve dificuldades em distinguir entre classes visualmente semelhantes, como sapatos e sandálias, ou gatos e cães. Esses padrões indicam que, apesar dos métodos de agregação avançados, há necessidade de melhorias adicionais no pré-processamento dos dados e na arquitetura dos modelos para lidar eficazmente com as sutilezas entre as classes, conforme sugerido por LeCun et al. (1998) e Krizhevsky et al. (2012).

As curvas ROC/AUC também revelaram importantes percepções sobre a capacidade discriminativa dos modelos. No dataset CIFAR-10, por exemplo, as classes 2 (bird) e 6 (frog) apresentaram as menores AUCs, refletindo a dificuldade do modelo em diferenciar essas classes devido à alta similaridade visual. Este resultado sublinha a necessidade de arquiteturas mais sofisticadas e técnicas de extração de características que possam capturar nuances mais finas entre classes complexas.

Em suma, este estudo contribuiu significativamente para o entendimento das dinâmicas do Aprendizado Federado, demonstrando a importância de escolher o método de agregação apropriado de acordo com a complexidade e a heterogeneidade dos dados. Os métodos avançados, como WeightedFedAvg e FedProx, mostraram-se superiores, especialmente em cenários desafiadores como o CIFAR-10, onde a capacidade de adaptação às peculiaridades dos dados é crucial para o sucesso do modelo. No entanto, também ficou claro que há espaço para avanços adicionais, particularmente no que diz respeito à otimização das redes para classes visualmente semelhantes e ao tratamento de dados altamente variáveis.

Este trabalho abre caminho para futuras pesquisas que possam explorar técnicas de agregação ainda mais refinadas, arquiteturas de redes neurais especializadas e abordagens inovadoras para lidar com a heterogeneidade dos dados em ambientes federados. Além disso, a integração de ferramentas como matrizes de confusão e curvas ROC/AUC na avaliação de modelos de Aprendizado Federado oferece uma nova perspectiva que combina abordagens quantitativas e qualitativas para uma compreensão mais completa do desempenho dos modelos.

6.1 Trabalhos Futuros

Com base nas limitações identificadas e nos resultados obtidos, sugerem-se algumas direções promissoras para futuras pesquisas, visando melhorar ainda mais o estado da arte no Aprendizado Federado:

- **Desenvolvimento de Novos Métodos de Agregação:** Embora o WeightedFedAvg tenha mostrado superioridade, há espaço para inovação no desenvolvimento de novos métodos de agregação que possam lidar ainda melhor com a complexidade e heterogeneidade dos dados federados. Métodos que incorporem técnicas de aprendizado adaptativo ou que utilizem clustering dinâmico de clientes podem oferecer melhorias significativas.

- **Aprimoramento das Arquiteturas de Redes Neurais:** A dificuldade em distinguir entre classes visualmente semelhantes, especialmente no CIFAR-10, sugere a necessidade de redes neurais mais sofisticadas, como aquelas baseadas em atenção (transformers) ou redes convolucionais mais profundas, que possam capturar características mais detalhadas e melhorar a capacidade discriminativa do modelo.

- **Otimização para Cenários de Dados Desbalanceados:** Investigar técnicas de balanceamento de dados específicas para ambientes federados, como o uso de aprendizado com dados sintéticos ou ajustamento de métodos de agregação para compensar a distribuição desigual dos dados entre os clientes, pode ser uma área frutífera para futuras pesquisas.

- **Exploração de Dados Multimodais:** Expandir a aplicação do Aprendizado Federado para dados multimodais, como texto, imagem e áudio combinados, pode oferecer visões valiosas sobre a interoperabilidade entre diferentes tipos de dados e sobre como os métodos de agregação podem ser otimizados para lidar com essa complexidade.

Em conclusão, este estudo forneceu uma análise detalhada dos métodos de agregação no contexto do Aprendizado Federado, destacando suas vantagens e limitações em cenários de crescente complexidade. As investigações realizadas revelaram áreas críticas que exigem avanços adicionais, como o desenvolvimento de técnicas de agregação mais adaptativas e o aprimoramento das arquiteturas de redes neurais para lidar com a heterogeneidade dos dados. Abordar essas questões não só tem o potencial de melhorar significativamente a precisão e a robustez dos modelos em ambientes federados, mas também de expandir a aplicabilidade do Aprendizado Federado para domínios mais desafiadores, como aqueles que envolvem dados multimodais e distribuídos de forma desigual. Com isso, espera-se que o Aprendizado Federado evolua para atender com maior eficácia às demandas de aplicações reais em escala global, assegurando tanto a privacidade quanto o desempenho dos modelos treinados.

Referências bibliográficas

Almodóvar, A., Parras, J., and Zazo, S. (2024). Propensity weighted federated learning for treatment effect estimation in distributed imbalanced environments. *Computers in Biology and Medicine*, 178:108779.

Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečnỳ, J., Mazzocchi, S., McMahan, B., et al. (2019). Towards federated learning at scale: System design. *Proceedings of machine learning and systems*, 1:374–388.

Brisimi, T. S., Chen, R., Mela, T., Olshevsky, A., Paschalidis, I. C., and Shi, W. (2018). Federated learning of predictive models from federated electronic health records. *International journal of medical informatics*, 112:59–67.

Chen, H.-Y. and Chao, W.-L. (2020). Fedbe: Making bayesian model ensemble applicable to federated learning. *arXiv preprint arXiv:2009.01974*.

Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V. (2020). Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703.

Deng, L. (2012). The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine*, 29(6):141–142.

Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., and Ramage, D. (2018). Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*.

Hayat, A., Morgado-Dias, F., Bhuyan, B. P., and Tomar, R. (2022). Human activity recognition for elderly people using machine and deep learning approaches. *Information*, 13(6).

Hsu, T.-M. H., Qi, H., and Brown, M. (2019). Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. (2021). Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210.

Konečnỳ, J., McMahan, H. B., Ramage, D., and Richtárik, P. (2016). Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, T., Sahu, A. K., Sanjabi, M., Zaheer, M., Talwalkar, A., and Smith, V. (2018). On the convergence of federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*.

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. (2020). Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450.

Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. (2019). On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

Mohri, M., Sivek, G., and Suresh, A. T. (2019). Agnostic federated learning. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4615–4625. PMLR.

Qi, P., Chiaro, D., Guzzo, A., Ianni, M., Fortino, G., and Piccialli, F. (2023). Model aggregation techniques in federated learning: A comprehensive survey. *Future Generation Computer Systems*.

Reyes, J., Di Jorio, L., Low-Kam, C., and Kersten-Oertel, M. (2021). Precision-weighted federated learning. *arXiv preprint arXiv:2107.09627*.

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., Bakas, S., Galtier, M. N., Landman, B. A., Maier-Hein, K., et al. (2020). The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7.

Sheller, M. J., Edwards, B., Reina, G. A., Martin, J., Pati, S., Kotrotsou, A., Milchenko, M., Xu, W., Marcus, D., Colen, R. R., et al. (2020). Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific reports*, 10(1):12598.

Smith, V., Chiang, C.-K., Sanjabi, M., and Talwalkar, A. S. (2017). Federated multi-task learning. *Advances in neural information processing systems*, 30.

Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D., and Khazaeni, Y. (2020). Federated learning with matched averaging. *arXiv preprint arXiv:2002.06440*.

Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

Yang, Q., Liu, Y., Chen, T., and Tong, Y. (2019). Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19.

Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., and Chandra, V. (2018). Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*.