Desafios e Soluções para Privacidade em Aprendizado Federado: Abordagem de Privacidade Diferencial

José Augusto Nogueira Florentino[1], **Carlos Alberto Vieira Campos**[1]

[1]Centro de Ciências Exatas e Tecnologia
Programa de Pós-Graduação em Informática
Universidade Federal do Estado do Rio de Janeiro (UNIRIO)
Rio de Janeiro – RJ – Brazil

augusto.nogueira@edu.unirio.br, beto@uniriotec.br

Abstract. *This paper analyzes the application of differential privacy (DP) in federated learning (FL) using PyTorch, investigating the trade-off between privacy and performance on non-IID data. Experimentally, we demonstrate that DP impacts accuracy and loss, with higher accuracy (ϵ=0.5) resulting in greater degradation, but without rendering the system unviable. The research confirms the inverse relationship between privacy and model quality, highlighting the need for a balance. This work contributes to the practical implementation of robust data protection policies in FL.*
Index Terms*—privacidade diferencial, aprendizado federado, dados não-IID, PyTorch, Opacus.*

Resumo. *Este artigo analisa a aplicação de privacidade diferencial (DP) em aprendizado federado (FL) usando PyTorch, investigando o trade-off entre privacidade e desempenho em dados não-IID. Experimentalmente, demonstramos que a DP impacta a acurácia e a perda, com maior rigor (ϵ=0,5) resultando em maior degradação, mas sem inviabilizar o sistema. A pesquisa confirma a relação inversa entre privacidade e qualidade do modelo, ressaltando a necessidade de balanceamento. Este trabalho contribui para a implementação prática de políticas de proteção de dados robustas em FL.*
Index Terms*—privacidade diferencial, aprendizado federado, dados não-IID, PyTorch, Opacus.*

1. Introdução

As novas tecnologias de inteligência artificial (IA) e aprendizado de máquina (Machine Learning – ML) tem proporcionado significativas transformações em diversos setores da sociedade. Contudo, esse desenvolvimento também tem trazido desafios relevantes, especialmente relacionados à privacidade e segurança dos dados utilizados nos processos de treinamento de modelos preditivos [Dwork et al. 2006]. Tradicionalmente, o aprendizado de máquina segue uma abordagem centralizada, na qual grandes volumes de dados são coletados, armazenados e processados em servidores centralizados. Esse paradigma, entretanto, implica em riscos substanciais de vazamento de informações sensíveis e de violação de direitos individuais, especialmente em contextos regulados por legislações como o Regulamento Geral de Proteção de Dados (GDPR) na Europa e a Lei Geral de Proteção de Dados (LGPD) no Brasil [Li et al. 2020]. Entretanto, mesmo com a adoção

do FL, surgem novas preocupações relacionadas à privacidade. Pesquisas recentes demonstram que, a partir da análise de gradientes compartilhados ou de modelos agregados, ainda é possível realizar ataques de reconstrução ou inferência de dados privados dos clientes participantes [Geyer et al. 2017]. Para mitigar esses riscos, a Privacidade Diferencial (Differential Privacy - DP) tem sido incorporada ao contexto do aprendizado federado, como uma camada adicional de proteção. A DP oferece garantias matemáticas de que a inclusão ou exclusão de um único indivíduo no conjunto de dados não alterará significativamente a saída de uma análise, reduzindo, assim, a probabilidade de identificação de informações individuais [Abadi et al. 2016].

A combinação entre aprendizado federado e privacidade diferencial, apesar de conceitualmente atrativa, traz consigo desafios técnicos significativos. A adição de ruído aos gradientes, como requerido pelos mecanismos de DP, pode prejudicar a convergência dos modelos e afetar negativamente métricas de desempenho como acurácia e perda [Truex et al. 2020]. Assim, surge uma problemática central: como equilibrar a proteção da privacidade com a necessidade de manter a qualidade preditiva dos modelos em cenários de aprendizado federado? Esse questionamento orienta a presente pesquisa, que busca investigar de forma sistemática os impactos da aplicação de privacidade diferencial no desempenho de sistemas de aprendizado federado implementados com o framework PyTorch.

A heterogeneidade estatística entre os conjuntos de dados dos participantes, uma característica fundamental do aprendizado federado conhecida como distribuição não-IID, representa um desafio particular. A literatura aponta que, justamente nesse cenário onde cada cliente possui dados com distribuições distintas, os impactos da aplicação de privacidade diferencial se tornam ainda mais pronunciados [Li et al. 2020]. Nesses cenários, os clientes possuem conjuntos de dados com características estatísticas distintas, o que torna o processo de agregação global ainda mais desafiador. A utilização de técnicas de normalização e ajustes nos parâmetros de privacidade, como o valor de ϵ (epsilon), aparece como uma estratégia necessária para mitigar tais impactos [Hsieh et al. 2020].

A relevância deste estudo justifica-se pela crescente demanda por soluções de aprendizado de máquina que conciliem desempenho e proteção de dados. Em um momento histórico no qual a segurança da informação tornou-se prioridade em organizações de todos os portes, desenvolver e validar técnicas que garantam privacidade diferencial em FL é uma contribuição significativa para a ciência aplicada e para o setor produtivo [Beutel et al. 2020].

2. Trabalhos Relacionados

O modelo de treinamento centralizado, que depende da consolidação de dados em um único local, entra em conflito com a realidade dos dados modernos, muitos dos quais são gerados e armazenados em dispositivos de usuários. Conforme argumentado por [McMahan et al. 2017], além dos riscos inerentes à privacidade ao agregar informações sensíveis, a própria transferência desses dados distribuídos representa um obstáculo fundamental em termos de latência e custo de comunicação. Esse duplo desafio — privacidade e eficiência — é um dos principais catalisadores para o desenvolvimento de abordagens descentralizadas. Neste contexto, surge o FL como uma solução promissora que permite o treinamento colaborativo de modelos sem a necessidade de centralizar os dados.

A aplicação de IA e ML em setores críticos como saúde e finanças intensificou o debate sobre a privacidade. Modelos de previsão, para serem eficazes, necessitam de dados detalhados, muitas vezes de natureza pessoal, o que cria um risco fundamental. [Dwork et al. 2006] atacaram a essência deste problema ao propor a Privacidade Diferencial, uma definição matemática de privacidade que permite o treinamento de modelos e a realização de análises estatísticas, ao mesmo tempo que oferece uma garantia formal de que a contribuição específica de um único indivíduo não pode ser inferida a partir dos resultados públicos do modelo. Tradicionalmente, o ML segue uma abordagem centralizada, na qual grandes volumes de dados são coletados, armazenados e processados em servidores centralizados.

O FL é uma abordagem inovadora que permite que múltiplos participantes treinem um modelo de aprendizado de máquina de forma colaborativa, mantendo seus dados localmente. Isso significa que, em vez de enviar os dados para um servidor central, apenas as atualizações do modelo (como gradientes ou pesos) são compartilhadas. Essa característica intrínseca do FL já oferece um nível de privacidade ao minimizar a exposição de dados sensíveis [Li et al. 2020].

Pesquisas recentes demonstram que, a partir da análise de gradientes compartilhados ou de modelos agregados, ainda é possível realizar ataques de reconstrução ou inferência de dados privados dos clientes participantes [Geyer et al. 2017]. Para mitigar esses riscos, a Privacidade Diferencial (Differential Privacy - DP) tem sido incorporada ao contexto do aprendizado federado, como uma camada adicional de proteção. A DP oferece garantias matemáticas de que a inclusão ou exclusão de um único indivíduo no conjunto de dados não alterará significativamente a saída de uma análise, reduzindo, assim, a probabilidade de identificação de informações individuais [Abadi et al. 2016].

A combinação entre aprendizado federado e privacidade diferencial, apesar de conceitualmente atrativa, traz consigo desafios técnicos significativos. A adição de ruído aos gradientes, como requerido pelos mecanismos de DP, pode prejudicar a convergência dos modelos e afetar negativamente métricas de desempenho como acurácia e perda [Truex et al. 2020]. Assim, surge uma problemática central: como equilibrar a proteção da privacidade com a necessidade de manter a qualidade preditiva dos modelos em cenários de aprendizado federado? Esse questionamento orienta a presente pesquisa, que busca investigar de forma sistemática os impactos da aplicação de privacidade diferencial no desempenho de sistemas de aprendizado federado implementados com o framework PyTorch.

A integração de FL e DP tem sido um tópico de pesquisa ativo. Por exemplo, trabalhos como os de [Wei et al. 2020] propõem frameworks baseados em DP que adicionam ruído artificial aos parâmetros no lado do cliente antes da agregação, buscando um equilíbrio entre a proteção da privacidade e o desempenho do modelo. Outros estudos, como o de [Truex et al. 2020], exploram a Privacidade Diferencial Local (LDP) para garantir uma proteção de privacidade ainda mais forte, onde o ruído é adicionado nos dados do cliente antes mesmo de qualquer processamento local.

A padronização de pesos, ou Weight Standardization (WS), é uma técnica que tem ganhado atenção no campo do aprendizado profundo para melhorar a estabilidade do treinamento e a generalização dos modelos. Diferente de outras técnicas de normalização,

como Batch Normalization (BN) ou Layer Normalization (LN), que operam nas ativações das camadas, a WS atua diretamente nos pesos das camadas convolucionais. A abordagem de [Vieira and Campos 2024], atua diretamente nos pesos convolucionais, para resolver os desafios do Aprendizado Federado em dados altamente não-IID. Nesses cenários, caracterizados por uma grande distância entre as distribuições de dados dos clientes (alto EMD), a performance do modelo costuma degradar e os custos de comunicação aumentam devido à lenta convergência. O método proposto, FedWS, demonstrou reduzir os efeitos da divergência de pesos em níveis elevados de heterogeneidade. Isso ocorre porque a WS suaviza os gradientes no treinamento local, o que diminui o impacto da distribuição de dados heterogênea e resulta em uma convergência mais rápida e estável , levando à redução considerável de custos de comunicação.

No contexto do Aprendizado Federado, a aplicação de técnicas de padronização de pesos tem-se mostrado promissora para mitigar os desafios impostos pela heterogeneidade dos dados. A ideia é que, ao padronizar os pesos localmente em cada cliente, é possível reduzir a variabilidade entre os modelos treinados individualmente, facilitando a agregação e melhorando a qualidade do modelo global. Trabalhos como o de [Kim et al. 2025] propõem o FedWSQ, um framework que integra a padronização de pesos e a quantização para um FL mais eficiente, destacando como a WS realiza um filtro de gradientes implícito que remove componentes alinhados aos pesos locais já enviesados e também o componente médio dos gradientes, que pode ser igualmente influenciado pelos dados locais. Essa filtragem, somada a um efeito de regularização, melhora a estabilidade da convergência e a robustez do modelo.

Para lidar com a degradação de performance em dados não-IID, o trabalho proposto por [Vieira and Campos 2024], introduz a técnica FedWS (Federated Learning with Weight Standardization). A abordagem atua diretamente na causa do problema: a divergência de pesos entre os clientes, que é agravada em cenários com alta heterogeneidade estatística. Para quantificar essa heterogeneidade, o autor utiliza a EMD (Earth Mover's Distance), uma métrica que calcula a distância entre a distribuição de classes nos dados de um cliente e a distribuição de classes global. [Vieira and Campos 2024] identifica a EMD como fator determinante da divergência de pesos, onde valores maiores de EMD indicam um desequilíbrio mais drástico e, consequentemente, um maior impacto negativo na acurácia do modelo. O diferencial do FedWS ocorre durante o treinamento local, onde a padronização dos pesos das camadas convolucionais é aplicada para suavizar os gradientes e as perdas (losses).Essa suavização reduz o impacto multiplicativo do EMD na divergência de pesos, estabilizando-a desde a primeira rodada e fazendo com que as rodadas subsequentes sejam menos afetadas pela distribuição dos dados. Empiricamente, o FedWS não apenas alcançou uma acurácia superior, mas também demonstrou uma convergência mais rápida, o que se traduz em uma redução drástica dos custos de comunicação. Essa eficiência o torna uma solução particularmente adequada para ambientes federados realistas, compostos por dispositivos com capacidade computacional restrita.

3. Método Proposto

A estrutura proposta foi baseada em um sistema de aprendizado federado com suporte à simulação de múltiplos clientes em ambiente controlado, todos operando com a biblioteca PyTorch. O código original foi projetado para investigar o impacto de técnicas de

normalização em cenários distribuídos. Para a realização deste trabalho, foram implementadas extensões ao código para suportar a adição de privacidade diferencial, sem prejuízo das funcionalidades existentes.

O projeto foi organizado em módulos responsáveis pelas seguintes funções: geração e distribuição dos dados de entrada, definição da arquitetura do modelo, execução do treinamento local por cliente, agregação dos parâmetros globais e avaliação dos modelos a cada rodada de treinamento. Como resultado da aplicação de diferentes níveis de privacidade, observou-se uma variação na consistência do desempenho entre os clientes. A Tabela 1 sumariza o desvio padrão da acurácia final, evidenciando que garantias de privacidade mais rigorosas levaram a uma maior dispersão nos resultados.

Tabela 1. Desvio padrão da acurácia

Cenário	**Desvio Padrão da Acurácia (%)**
Cenário 1 – Sem DP	3,5
Cenário 2 – DP Moderada (ϵ=1,0)	6,2
Cenário 3 – DP Rigorosa (ϵ=0,5)	9,8

A principal modificação metodológica do projeto consistiu na incorporação da biblioteca Opacus, uma extensão do PyTorch especializada na implementação de privacidade diferencial [Andrew et al. 2021]. Essa integração exigiu a reestruturação do pipeline de treinamento para incluir as etapas de clipping de gradientes e adição de ruído gaussiano calibrado.

Inicialmente, para aplicar a Privacidade Diferencial, a influência de cada amostra de dado no processo de treinamento foi limitada através do recorte (clipping) da norma L2 dos gradientes individuais, sendo o valor de recorte fixado em 1,0. Esta é uma prática padrão em implementações de DP para controlar a sensibilidade da função. Em seguida, foram configurados os parâmetros de privacidade (ϵ e δ). Foram selecionados dois níveis para o orçamento de privacidade ϵ: 1,0, representando uma garantia de privacidade moderada, e 0,5, para uma privacidade mais rigorosa. O parâmetro δ, que representa a probabilidade de a garantia formal de privacidade não ser mantida, foi fixado em 1^{-5}. A escolha de um δ criptograficamente pequeno, idealmente menor que o inverso do tamanho do conjunto de dados, é um pré-requisito comum para assegurar uma proteção robusta. Por fim, ruído Gaussiano, com magnitude calibrada de acordo com o valor de recorte e os parâmetros de privacidade, é adicionado para satisfazer a definição de (ϵ e δ)-Privacidade Diferencial.

Foram definidos três cenários experimentais distintos para fins comparativos:

• Cenário 1: Aprendizado Federado sem Privacidade Diferencial - Modelo de referência utilizado como baseline para comparação dos impactos da privacidade.

• Cenário 2: Aprendizado Federado com Privacidade Diferencial (ϵ=1,0) - Configuração intermediária, com nível moderado de privacidade.

• Cenário 3: Aprendizado Federado com Privacidade Diferencial (ϵ=0,5) - Configuração com nível mais alto de privacidade, representando um ambiente mais restritivo.

Para cada cenário, foram simulados dez clientes com dados distribuídos de

forma não-IID, seguindo uma partição heterogênea inspirada no método proposto por [Li et al. 2020], com o objetivo de refletir a diversidade estatística típica de ambientes reais.

As principais variáveis dependentes consideradas na análise foram a acurácia global do modelo, a função de perda agregada, a variabilidade de desempenho entre os clientes e o consumo acumulado do orçamento de privacidade. Como variáveis independentes, consideraram-se os níveis de privacidade adotados e o número de rodadas de treinamento.

A métrica de acurácia foi calculada a partir do percentual de classificações corretas em um conjunto de dados de validação global. A perda foi avaliada utilizando a função CrossEntropyLoss, padrão para problemas de classificação multicategorias em PyTorch [Paszke et al. 2019]. Para a análise da variabilidade entre clientes, foi utilizado o desvio padrão das acurácias locais por rodada, permitindo observar o grau de dispersão dos resultados individuais, conforme metodologia aplicada por [Hsieh et al. 2020].

Os resultados finais consolidados para cada um dos três cenários experimentais são apresentados na Tabela 2. Os dados demonstram o claro trade-off entre a aplicação de privacidade e o desempenho do modelo. Observa-se que, à medida que a garantia de privacidade se torna mais rigorosa (de Sem DP para DP Rigorosa), a acurácia final média do modelo global diminui, enquanto a variabilidade entre os clientes, indicada pelo desvio padrão, aumenta consideravelmente.

Tabela 2. Desvio padrão por cenário de Privacidade Diferencial

Cenário	**Acurácia Final (%)**	**Desvio Padrão (%)**
Cenário 1 – Sem DP	88	3,5
Cenário 2 – DP Moderada (ϵ=1,0)	82	6,2
Cenário 3 – DP Rigorosa (ϵ=0,5)	76	9,8

4. Experimentos

Cada cenário experimental foi executado por cinco rodadas globais de treinamento, com avaliação ao final de cada rodada. Os treinamentos foram realizados em ambiente local com suporte a GPU, utilizando os recursos de paralelismo do PyTorch para simular a execução concorrente dos múltiplos clientes [Beutel et al. 2020].

Em conformidade com as boas práticas para garantir a validade científica e a reprodutibilidade dos resultados, todos os experimentos foram executados com seeds fixos. Um seed é um valor inicial que alimenta o gerador de números pseudoaleatórios, garantindo que a mesma sequência de eventos aleatórios ocorra a cada nova execução do código. No contexto deste trabalho, a fixação do seed controla fontes críticas de variabilidade, incluindo: (i) a inicialização dos pesos dos modelos, (ii) a seleção do subconjunto de clientes participantes em cada rodada de comunicação e (iii) o embaralhamento dos lotes de dados (mini-batches) durante o treinamento local. Essa abordagem é fundamental para assegurar que as diferenças de desempenho observadas sejam atribuíveis exclusivamente às variações nas técnicas testadas, e não a uma ”sorte”estatística. Para complementar a análise, os logs de treinamento foram registrados de forma contínua, permitindo uma avaliação estatística detalhada dos resultados.

Durante as execuções dos cenários com privacidade diferencial, o PrivacyEngine foi configurado para monitorar o consumo de ϵ, interrompendo o treinamento caso o orçamento de privacidade fosse atingido antes do número previsto de rodadas.

A análise dos dados seguiu abordagem quantitativa, com geração de gráficos de linha e boxplots para representar as evoluções de acurácia, perda e variabilidade entre os clientes. As curvas de consumo de ϵ também foram traçadas para cada cenário com DP, permitindo observar o comportamento do orçamento de privacidade ao longo das iterações.

Os resultados foram comparados entre os cenários para validar a hipótese central do trabalho: a existência de um trade-off mensurável entre o nível de privacidade diferencial aplicado e o desempenho global do modelo federado, um princípio validado em trabalhos como o de [Truex et al. 2020].

A interpretação dos resultados considerou também aspectos qualitativos, como a estabilidade das curvas de perda e a tendência de convergência dos modelos, mesmo em cenários com elevada inserção de ruído nos gradientes. Além da análise de desempenho, é fundamental monitorar o custo de privacidade ao longo do tempo. A Tabela 3 detalha o consumo acumulado do orçamento de privacidade ϵ para os dois cenários protegidos. Conforme as propriedades de composição da Privacidade Diferencial, o gasto do orçamento aumenta linearmente a cada rodada, refletindo o custo total da privacidade aplicada até aquele ponto do treinamento.

Tabela 3. Consumo acumulado

Rodada	Cenário 2 - ϵ=1,0 (%)	Cenário 3 ϵ=0,5 (%)
1	0,2	0,4
2	0,4	0,8
3	0,6	1,2
4	0,8	1,6
5	1	2

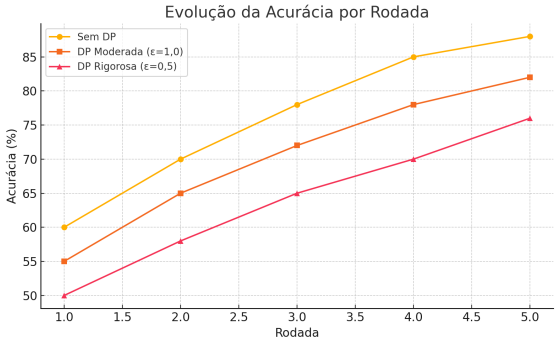
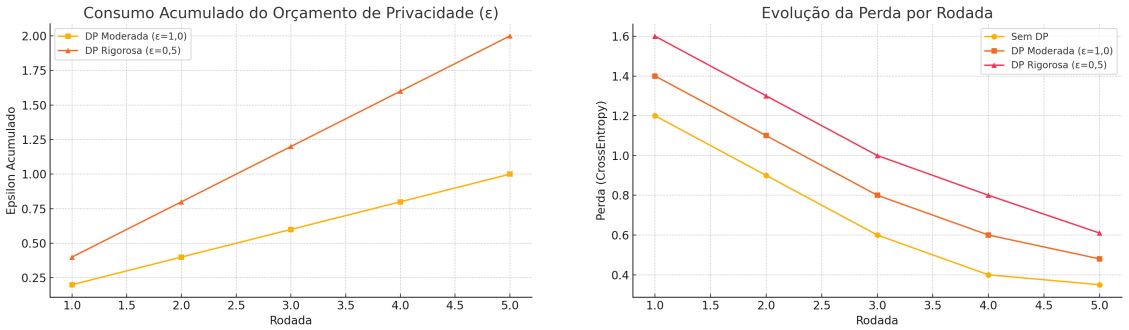


Figura 1. Evolução da Acurácia

A Figura 1 ilustra a evolução da acurácia do modelo global ao longo de cinco rodadas de treinamento federado. Observa-se que todos os cenários apresentam uma curva de aprendizado positiva, com a acurácia aumentando a cada rodada. O cenário Sem DP atinge o maior desempenho, servindo como linha de base. A introdução da privacidade

diferencial resulta em uma queda na acurácia final, confirmando o conhecido trade-off entre privacidade e utilidade, um princípio também validado em trabalhos como o de [Truex et al. 2020]. O impacto é proporcional à força da privacidade: o cenário de DP Rigorosa (ϵ=0,5), que adiciona mais ruído para uma maior proteção, apresenta uma acurácia inferior ao de DP Moderada (ϵ=1,0).



(a) Orçamento de Privacidade **(b) Perda por Rodada**

Figura 2. Gráficos de análise do treinamento com Privacidade Diferencial.

A Figura 2a demonstra como o orçamento de privacidade (ϵ) é consumido de forma acumulada a cada rodada de treinamento. Conforme esperado pelas propriedades de composição da Privacidade Diferencial, o gasto do orçamento aumenta linearmente com o número de interações com os dados. Este gráfico visualiza o custo total de privacidade ao longo do tempo para os dois cenários privados.

A Figura 2b apresenta a evolução da função de perda (loss) do modelo, medida por Cross-Entropy, ao longo das rodadas. Os resultados são consistentes e inversamente correlacionados com os da acurácia. Todos os cenários mostram uma redução na perda, indicando que o treinamento está convergindo. O cenário Sem DP alcança o menor valor de perda, enquanto os cenários com privacidade diferencial convergem para valores de perda mais elevados. O ruído injetado para garantir a privacidade torna a tarefa de otimização do modelo mais desafiadora, resultando em um erro final maior. Novamente, a DP Rigorosa (ϵ=0,5) apresenta a maior perda, alinhando-se com os resultados de acurácia mais baixos e reforçando o trade-off entre a força da garantia de privacidade e a performance do modelo.

Para minimizar a influência de fatores externos, as execuções foram repetidas três vezes para cada cenário, com posterior cálculo da média dos resultados obtidos. Essa prática buscou reduzir o impacto de flutuações causadas por instabilidades do hardware ou variações estocásticas do processo de inicialização de pesos [Abadi et al. 2016].

As limitações inerentes ao ambiente simulado, como a ausência de comunicação real em rede e a execução local dos clientes, foram reconhecidas, mas consideradas adequadas para os objetivos desta etapa de pesquisa, conforme abordagens metodológicas similares descritas na literatura [Li et al. 2020].

Por fim, a metodologia empregada neste estudo foi validada por meio da comparação com experimentos de referência na área de FL com DP, buscando garantir que as configurações, métricas e procedimentos estivessem alinhados às melhores práticas científicas descritas em publicações recentes [Truex et al. 2020].

5. Conclusão

A análise dos resultados obtidos na implementação de privacidade diferencial em um sistema de aprendizado federado com PyTorch, permitiu validar as hipóteses formuladas na etapa inicial deste trabalho. Os dados experimentais demonstraram que a adição de mecanismos de privacidade diferencial impacta diretamente o desempenho do modelo global, principalmente nas métricas de acurácia e função de perda, sem, contudo, inviabilizar a aplicação do sistema em cenários reais.

A primeira hipótese testada, que previa uma degradação controlada da acurácia em função do nível de privacidade imposto, foi amplamente confirmada. Conforme verificado, a redução da acurácia nos cenários com ϵ=1,0 e ϵ=0,5 ocorreu de maneira previsível e em conformidade com as curvas de aprendizado observadas. O comportamento de queda progressiva na performance, mais acentuado no cenário de maior rigor (ϵ=0,5), está alinhado aos estudos de [Abadi et al. 2016], que evidenciam a relação inversa entre o nível de privacidade e a qualidade do modelo treinado.

Outro aspecto relevante foi o aumento observado na função de perda global. Tal elevação é resultado direto da inserção de ruído gaussiano aos gradientes durante o processo de agregação, mecanismo central das estratégias de privacidade diferencial [Dwork et al. 2006]. Embora a perda tenha aumentado de forma consistente com o nível de privacidade adotado, a tendência de queda ao longo das rodadas sugere que o modelo manteve sua capacidade de aprendizado, confirmando a robustez do método FedAvg e das técnicas de normalização aplicadas no pré-processamento dos dados [Li et al. 2020].

A análise da variabilidade de desempenho entre os clientes também trouxe insights significativos. Observou-se que o aumento da heterogeneidade nas acurácias locais, principalmente nos cenários com privacidade diferencial, é um efeito esperado da aplicação de ruído estocástico não uniformemente distribuído [Truex et al. 2020]. Esse fenômeno reforça a necessidade de futuros ajustes em estratégias de balanceamento, como o uso de clipping adaptativo, conforme proposto por [Andrew et al. 2021].

O desenvolvimento desta pesquisa teve como foco central a implementação e a análise dos impactos da privacidade diferencial no desempenho de um sistema de aprendizado federado, utilizando como base o framework PyTorch.

Em termos de contribuições para a área de aprendizado federado com privacidade diferencial, este trabalho demonstrou, de maneira prática e mensurável, que é possível implementar políticas de proteção de dados robustas sem perda catastrófica de desempenho. Os resultados também sugerem que níveis moderados de privacidade (como ϵ=1,0) podem oferecer um equilíbrio aceitável entre segurança e acurácia, viabilizando a adoção dessa abordagem em cenários de aplicação real, como na área da saúde, finanças e dispositivos móveis [Beutel et al. 2020].

Como perspectiva para trabalhos futuros, recomenda-se a exploração de novas estratégias de otimização, como o uso de algoritmos de agregação robusta e a investigação do impacto da privacidade diferencial em arquiteturas mais complexas de redes neurais. Além disso, sugere-se a realização de experimentos com diferentes conjuntos de dados, bem como a análise da escalabilidade da solução para ambientes com número elevado de clientes, ampliando o escopo de aplicação dos conhecimentos aqui produzidos.

Referências

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. (2016). Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318.

Andrew, G., Thakkar, O., McMahan, B., and Ramaswamy, S. (2021). Differentially private learning with adaptive clipping. *Advances in Neural Information Processing Systems*, 34:17455–17466.

Beutel, D. J., Topal, T., Mathur, A., Qiu, X., Fernandez-Marques, J., Gao, Y., Sani, L., Li, K. H., Parcollet, T., De Gusmão, P. P. B., et al. (2020). Flower: A friendly federated learning research framework. *arXiv preprint arXiv:2007.14390*.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer.

Geyer, R. C., Klein, T., and Nabi, M. (2017). Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*.

Hsieh, K., Phanishayee, A., Mutlu, O., and Gibbons, P. (2020). The non-iid data quagmire of decentralized machine learning. In *International Conference on Machine Learning*, pages 4387–4398. PMLR.

Kim, S.-W., Kim, S., Kim, J., Ji, S., and Lee, S.-H. (2025). Fedwsq: Efficient federated learning with weight standardization and distribution-aware non-uniform quantization. *arXiv preprint arXiv:2506.23516*.

Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. (2020). Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60.

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.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Truex, S., Liu, L., Chow, K.-H., Gursoy, M. E., and Wei, W. (2020). Ldp-fed: Federated learning with local differential privacy. In *Proceedings of the third ACM international workshop on edge systems, analytics and networking*, pages 61–66.

Vieira, F. and Campos, C. A. V. (2024). Reducing costs using normalization in federated learning in heterogeneous data distributions.

Wei, K., Li, J., Ding, M., Ma, C., Yang, H. H., Farokhi, F., Jin, S., Quek, T. Q., and Poor, H. V. (2020). Federated learning with differential privacy: Algorithms and performance analysis. *IEEE transactions on information forensics and security*, 15:3454–3469.