UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE CIÊNCIAS
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

ANTÔNIO ALVES MARREIRAS NETO

UMA AVALIAÇÃO DE ABORDAGENS LDP APLICADAS A CONJUNTOS DE DADOS LONGITUDINAIS

FORTALEZA
2025

ANTÔNIO ALVES MARREIRAS NETO

UMA AVALIAÇÃO DE ABORDAGENS LDP APLICADAS A CONJUNTOS DE DADOS LONGITUDINAIS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Ciência da Computação.

Orientador: Prof. Dr. Javam de Castro Machado.

FORTALEZA

2025

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

M324a Marreiras Neto, Antonio Alves.
Uma avaliação de abordagens LDP aplicadas a conjuntos de dados longitudinais / Antonio Alves Marreiras Neto. – 2025.
39 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Centro de Ciências, Curso de Computação, Fortaleza, 2025.
Orientação: Prof. Dr. Javam de Castro Machado.

1. Proteção de dados. 2. Privacidade diferencial local. 3. Bancos de dados longitudinais. I. Título.
CDD 005

ANTÔNIO AlVES MARREIRAS NETO

UMA AVALIAÇÃO DE ABORDAGENS LDP APLICADAS
A CONJUNTOS DE DADOS LONGITUDINAIS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Ciência da Computação.

Aprovada em: xx/02/2025.

BANCA EXAMINADORA

_____
Prof. Dr. Javam de Castro Machado (Orientador)
Universidade Federal do Ceará (UFC)

_____
Prof. Dr. José Maria da Silva Monteiro Filho
Universidade Federal do Ceará (UFC)

_____
Prof. Dr. Leonardo Oliveira Moreira
Universidade Federal do Ceará (UFC)

_____
Prof. Dr. Victor Aguiar Evangelista de Farias
Universidade Federal do Ceará (UFC)

À minha família.

À minha mãe, Priscilla.

AGRADECIMENTOS

A minha família, pelo encorajamento nos momentos em que enfrentei maiores dificuldades e dúvidas.

Ao Prof. Dr. Javam de Castro Machado, pela orientação e oportunidade de participar do Laboratório de Sistemas de Banco de Dados (LSBD).

Aos colegas do laboratório, por todo o apoio, neste e em trabalhos passados que formaram a base da trajetória de pesquisa da qual este é o mais recente produto. Em especial, a Eduardo Rodrigues, que foi co-autor em trabalhos passados e cuja colaborações foi fundamental para o desenvolvimento de minha produção acadêmica.

Aos professores participantes da banca examinadora Prof. Dr. José Maria da Silva Monteiro Filho, Prof. Dr. Leonardo Oliveira Moreira e Prof. Dr. Victor Aguiar Evangelista de Farias pelo tempo, pelas valiosas observações e sugestões.

RESUMO

Privacidade diferencial local (LDP) foi desenvolvida como uma versão mais rigorosa da privacidade diferencial (DP), o modelo state-of-the-art de garantia de anonimato para banco de dados. Devido ao requisito de anonimização de dados antes do envio ao servidor, um dos desafios na garantia de LDP é o risco de adição excessiva de ruído aos dados, o que pode ser especialmente difícil de se evitar quando se aplica LDP a dados longitudinais, que requerem sucessivas consultas ao longo do tempo, cada uma precisando garantir LDP. Neste trabalho, procuramos avaliar a performance de mecanismos LDP adaptados para a proteção de dados longitudinais, quando aplicados à tarefa de encontrar os k itens mais frequentes, e suas frequências, entre conjuntos de dados longitudinais. Para este fim, avaliamos o desempenho de uma série extensa de mecanismos LDP quando utilizados em conjunto da abordagem state-of-the-art SVIM, no processamento de quatro diferentes conjuntos de dados. Após exaustiva experimentação, comparamos os resultados encontrados e indicamos os mais promissores.

Palavras-chave: Proteção de dados; privacidade diferencial local; bancos de dados longitudinais

ABSTRACT

Local differential privacy (LDP) was developed as a more strict version of differential privacy (DP), the state-of-the-art model of anonymity guarantee for databases. Due to the requirement of data anonymity before it is sent to the server, a challenge in guaranteeing LDP is the risk of excessive addition of noise to the data, which can be especially hard to avoid when applying LDP to longitudinal data, that requires successive queries over time, with each one having to guarantee LDP. In this paper, we aim to evaluate the performance of LDP protocols adapted for the protection of longitudinal data, when applied to the task of finding the k most frequent items, and their frequencies, among longitudinal data sets. To this end, we evaluate the performance of a wide range of LDP mechanisms when used in conjunction with the state-of-the-art SVIM approach, in the processing of four different datasets. After exhaustive experimentation, we compared the results found and indicated the most promising ones.

Keywords: Data protection; local differential privacy; longitudinal databases.

LISTA DE GRÁFICOS

LISTA DE ABREVIATURAS E SIGLAS

CCPA	*California Consumer Privacy Act*
GDPR	*General Data Protection Regulation*
LGPD	Lei Geral de Proteção de Dados
DP	*Differential Privacy*
LDP	*Local Differential Privacy*
SVIM	*Set-Value Item Mining*
FO	*Frequency Oracle*
PSFO	*Padding-and-Sampling-based Frequency Oracle*
UE	*Unary Encoding*

LISTA DE SÍMBOLOS

ϵ Orçamento de privacidade

M Mecanismo

τ Conjunto de timestamps

SUMÁRIO

1 INTRODUÇÃO

No decorrer das últimas duas décadas, em especial após a adoção em massa de serviços e produtos dependentes da internet, se acelerou uma crescente coleta de dados de indivíduos por meio de diversos serviços fornecidos por ambos os setores públicos e privados. Em consequência deste crescente volume de dados coletados, uma maior preocupação da sociedade civil com a privacidade vem ganhando força em reação ao cenário atual. Por exemplo, existem preocupações relacionadas a coleta de dados como histórico de saúde por parte da indústria farmacêutica e de como isso pode possibilitar a descriminação com indivíduos com um histórico de certas condições de saúde, e como provedores de certos medicamento podem deixar de oferecer promoções, ou planos de saúde podem aumentar suas taxas a priori etc.

Assim, com a intenção de satisfazer as demandas da população geral, governos têm adotado legislações em favor da preservação do direito à privacidade de indivíduos, limitando como pode se dar a coleta de dados, em especial de dados sensíveis que possam ferir o anonimato, ou prejudicar indivíduos. Entre as mais influentes legislações adotadas sobre a questão estão a GDPR (General Data Protection Regulation) em 2016 pela União Européia (General Data Protection Regulation, 2016), CCPA (California Consumer Privacy Act) em 2018 (CALIFORNIA, 2018), e a LGPD (Lei Geral de Proteção de Dados) também em 2018 pelo Brasil (BRASIL, 2018). A GDPR em particular foi bastante influente e foi considerada base também para a LGPD.

Como resultado deste cenário, empresas e pesquisadores apresentaram demandas por novas e mais efetivas formas para realizar a coleta de dados de indivíduos sem a quebra de anonimato. Atualmente, o padrão state-of-the-art de garantia de anonimato é a privacidade diferencial(DWORK et al., 2006), porém mesmo este possui limitações: o modelo clássico da privacidade diferencial requer um curador ou servidor confiável, o que nem sempre é viável e também costuma causar ceticismo por parte dos usuários cujo dados estão sendo coletados. Assim, um novo modelo mais rigoroso, denominado privacidade diferencial local (ou LDP, do inglês Local differential Privacy) foi proposto(ERLINGSSON et al., 2014). Um modelo que não depende de um curador ou servidor confiável, pois os dados são anonimizados no momento da coleta, antes de serem enviados para o servidor.

Contudo, como resultado da necessidade de anonimização no momento da coleta de dados, a maioria dos mecanismos LDP (isto é, algoritmos que garantem os requisitos da

privacidade diferencial local) costumam ou correm o risco de adicionar uma quantidade excessiva de ruído aos dados, distorcendo significativamente as características do conjunto de dados anonimizados, o distanciando dos dados reais e reduzindo a sua utilidade para análises. Evitar a aplicação excessiva de ruído pode ser particularmente difícil dependendo do tipo de dado e da tarefa realizada. Por exemplo, estimativa de frequências, apesar de se tratar de uma tarefa que envolve a anonimização de grandes volumes de dados, possui mecanismos que realizam por meio de contagem, uma forma de garantir LDP com o mínimo de adição de ruído o possível, porém esta tarefa já se torna mais complexa quando deve ser realizada sob dados longitudinais, que exigem que conforme novas coletas de dados vão sendo realizadas ao longo do tempo, as estimativas têm de ser recalculadas, e assim, os dados necessitam de uma nova adição de ruído para cada nova amostra, assim diminuindo a utilidade dos resultados finais ao longo do tempo.

1.1 Propósito

Conduzimos uma série extensa de experimentos com o intuito de determinar quais os oráculos de frequência, que garantem privacidade diferencial local adaptada para dados longitudinais, mais adequados para o uso na tarefa de determinar os k itens mais frequentes entre conjuntos de dados longitudinais.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Dados longitudinais

Dados Longitudinais são definidos como dados que evoluem de acordo com o tempo, tal que são amostrados em diferentes intervalos de tempo, referidos no banco de dados como *timestamps*. Usuários têm seus dados enviados ao servidor a cada *timestamp*, de forma que cada registro correspondente a um usuário contém uma amostra de seus dados por *timestamp.*

2.2 Privacidade diferencial

Privacidade diferencial ou DP (do inglês, *Differential Privacy*) (DWORK et al., 2006) é uma definição formal de privacidade com garantias independente de um indivíduo possuir ou não um registro em um banco de dados. Considerada o modelo *state-of-the-art* de garantir privacidade, em seu *setting* original ou central, DP é garantida por meio de algoritmos, que adicionam ruído a uma resposta a consulta em um banco de dados de acordo com sensibilidade global e orçamento determinados, garantindo que a presença ou não de qualquer indivíduo não vai alterar a probabilidade de resposta de uma consulta. Estes algoritmos são denominados mecanismos diferencialmente privados.

O modelo central de DP pode ser adaptado tanto para consultas interativas, como para a disponibilidade de conjuntos de dados anonimizados, porém independente de como for utilizado, ainda há a necessidade de um servidor ou curador de dados confiável, responsável pelo manejo dos dados não-anonimizados e que é considerado uma limitação ou mesmo potencial falha de segurança para DP, dada a possibilidade de vazamento de dados e acesso por atores maliciosos. Assim, cumprindo demandas de pesquisa e do mercado, privacidade diferencial local foi desenvolvida como um modelo alternativo, mais rigoroso e sem a necessidade de curador ou servidor confiável.

2.3 Privacidade diferencial local

A privacidade diferencial local (ERLINGSSON et al., 2014) ou LDP (do inglês, *Local Differential Privacy*) garante a privacidade de indivíduos sem a necessidade de servidor

ou curador confiável fazendo uso de mecanismos DP que aplicam garantias de privacidade antes de enviar os dados ao servidor. De forma que os dados têm o ruído adicionado no momento da coleta, e já estão anonimizados antes de serem agregados por um servidor ou curador.

Uma definição formal de LDP pode ser dada por dado um algoritmo aleatório $\mathcal{Y}$ satisfaz ε-privacidade diferencial local (é um mecanismo LDP) se, para quaisquer duas possíveis entradas V e V′, e para qualquer saída possível Y $\subseteq$ Range($\mathcal{Y}$), temos: Pr[$\mathcal{Y}$(v) = y] $\leq$ exp(ε)*Pr[$\mathcal{Y}$(v') = y], onde Pr denomina probabilidade e ε é uma variável referida como "orçamento de privacidade" ou simplesmente *budget*, que por sua vez determina o grau de privacidade garantida pelo algoritmo de forma que quanto menor *budget*, maior privacidade é garantida e maior o ruído aos dados tende a ser aplicado,e quanto maior o *budget*, menos privacidade é garantida e menor o ruído aplicado de forma que os dados anonimizados vão tender a serem mais similares aos reais.

A relação de semelhança entre os dados anonimizados e reais costuma definir a utilidade dos dados, uma vez que análises feitas sob dados anonimizados com alta utilidade costumam melhor refletir os resultados das mesmas realizadas sob os dados reais. A utilidade dos dados e o ruído aplicado são portanto inversamente proporcionais, logo se dá preferência a mecanismos LDP que garantem o maior grau de privacidade possível (*budgets* menores) com menor adição de ruído (maior utilidade dos dados).

2.3.1 Resistência a pós-processamento

LDP tem entre as suas propriedades a proposição da resistência a pós-processamento (DWORK et al., 2014), na qual pós-processamento é definido como qualquer função que receba como entrada a saída de um mecanismo LDP, de forma que não importando qual seja a função, a sua saída ainda vai possuir garantias LDP. Assim a resistência a pós-processamento garante que uma vez que o dado tenha sido anonimizado com garantias LDP, quaisquer operações realizadas sob ele em seguida, mesmo que aumentem sua utilidade, ele ainda mantém o mesmo nível de garantia de privacidade.

2.3.2 Composição Sequencial

Segundo a proposição da composição sequencial (DWORK et al., 2014), se um mecanismo LDP com ϵ_1 de orçamento de privacidade, tem sua saída processada por outro mecanismo

LDP com *budget* ϵ_2 logo essa sequencial de mecanismos é equivalente a um mecanismo com garantias $\epsilon_1 + \epsilon_2$-LDP. Em uma definição formal generalizada: seja *Mt* um mecanismo ϵ_t -LDP, tal que t $\in$ [1,...,τ]. Então, a sequência de saídas [$M_1(v)$, ..., $M_\tau(v)$] tem garantias $\sum_{t=1}^{\tau} \epsilon_t$-LDP.

3 PROBLEMA

Em um cenário onde cada usuário possui uma sequência de valores $s = [v_1,..., v_t]$ pertencente a um domínio *D*, no qual *t* é o número de timestamps em um registro longitudinal correspondente a cada usuário. Um agregador é responsável por calcular a frequência de cada item sem ter acesso às frequências reais, garantindo LDP. Existem diversos FOs desenvolvidos para lidar com este cenário, porém no caso descrito em (QIN et. al, 2016), cada usuário possua não apenas um valor, mas um conjunto de valores, um conjunto por timestamp no caso de um banco de dados longitudinais.

Neste caso, o agregador tem de executar a tarefa de encontrar os *k* itens mais frequentes e suas frequências, sem saber ao certo o tamanho do domínio *D*. A frequência de um valor *v* é definida como o número de ocorrências do valor entre diferentes conjuntos e timestamps, dvidido pelo número de usuários. Com o objetivo de determinar as abordagens mais adequadas para a execução desta tarefa, avaliamos a performance de diferentes mecanismos LDP adaptados para a tarefa de estimar frequências entre dados longitudinais, quando usados em conjunto de um algoritmo state-of-the-art para a amostragem de *k* itens mais frequentes. Determinamos a melhor abordagem pela maior utilidade após aplicar LDP. Medimos a utilidade a partir pela média do erro quadrático médio ao longo de timestamps, dado por: $MSEavg = \frac{1}{\tau}\sum_{t=1}^{\tau}\frac{1}{|D|}\sum_{v\in D}(\overline{f}(v)_t - f(v)_t)^2$. De forma que quão menor o *MSEavg*, melhor a utilidade.

Um exemplo real desta tarefa se aplicaria ao histórico de navegação de um browser de internet: vários conjuntos de diferentes endereços web, divididos em timestamps de dias. O desenvolvedor do browser poderia querer aprender quais lojas são mais visitadas por ser usuários, para determinar melhor patrocínios na home page. Por outro lado a coleta de dados dos usuários, feita de forma a ferir a privacidade dos mesmos, poderia resultar em um boicote do produto e problemas judiciais em regiões que adotam legislações que garantem o direito dos cidadãos à privacidade.

4 ORÁCULOS DE FREQUÊNCIA

Um oráculo de frequência, ou FO (do inglês, *frequency oracle*) é um mecanismo LDP que pode ser utilizado para estimar a frequência de qualquer valor v em um domínio D com garantias LDP. Um FO consiste de dois algoritmos: um sanitizador, e um agregador.

O sanitizador é responsável pela coleta dos dados dos usuários e a aplicação de ruído por meio de um algoritmo de perturbação, realizando uma contagem ruidosa dos atributos da consulta. Após a perturbação, o dado é enviado ao agregador. O agregador é executado no servidor, e funciona primeiro agregando as respostas do sanitizador, e em seguida realiza um cálculo de estimativa da frequência dos valores recebidos levando em consideração o *budget* de privacidade utilizado pelo sanitizador com o intuito de corrigir em parte o erro adicionado pelo processo de anonimização. Os FOs listados nesta seção fazem uso de uma mesma função de estimativa de frequências: $\Phi f\ (v) :=\ \frac{C(v) - nq}{n(p-q)}$, onde p e q são probabilidades do algoritmo de perturbação.

A saída final de um FO é um vetor de frequências representando um histograma de frequências dos valores do atributo coletado pelo FO. Os FOs possuem várias aplicações, como censos, construção de *heatmaps* e outras formas de histogramas além de serem um componente em outros mecanismos LDP com aplicações em *Itemset Mining* (QIN et. al 2016), identificação de *Heavy Hitters* (ZHU et. al, 2023), e consultas em intervalo (DA COSTA FILHO et. al, 2023).

4.1 Generalized Randomized Response (GRR)

O algoritmo Randomized Response (WARNER, 1965) introduzido em 1965, é considerado um dos primeiros mecanismos LDP, uma vez que apesar de seu desenvolvimento ser anterior a formalização do conceito de privacidade diferencial, LDP por se tratar de uma definição matemática, qualquer algoritmo que garantir seus requisitos pode ser considerado um mecanismo.

Randomized Response foi extremamente influente no desenvolvimento dos oráculos de frequência, porém só pode lidar com consultas binárias. GRR foi desenvolvido como uma generalização do Randomized Response para lidar com consultas com um maior domínio de possíveis respostas (KAIROUZ et al., 2016), e é considerado um dos FOs mais simples.

Ao executar GRR, os usuários enviam seus dados para o sanitizador, que retorna o valor real privado $v \in D$ com probabilidade p, ou um valor aleatório $v' \in D$ com probabilidade $1 - p$. Mais formalmente:

$$\forall_{x \in D}\ Pr\left[\Psi_{GRR_{(\epsilon)}}(v) = x\right] = \begin{cases} p = \frac{e^{\epsilon}}{e^{\epsilon}+|D|-1} & \text{if x = v} \\ q = \frac{1-p}{|D|-1} = \frac{1}{e^{\epsilon}+|D|-1} & \text{if x} \neq \text{v} \end{cases}$$

Podemos garantir que GRR satisfaz LDP desde que $p/q = e^{\epsilon}$.

Em seguida, agregador recebe as saídas do sanitizador, que são então reunidas em um vetor $x = [x_1, \ldots, x_n]$ de tamanho $|x| = n$ onde $x_i \in D$ é o valor recebido do i-ésimo usuário. Seja $C(n)$ o número de vezes em que um valor v ocorre no vetor x, a saída do algoritmo é dada pela função de estimativa $\Phi f\ (v) := \frac{C(v) - nq}{n(p-q)}$.

4.2 Unary Encoding (UE)

Unary Encoding (do inglês, codificação unária), no contexto de oráculos de frequência pode se referir como uma classe de FOs que fazem uso de codificação unária para garantir LDP enquanto reduz o ruído aplicado a consultas, resultando em melhor utilidade.

FOs que utilizam codificação unária, fazem de modo que um valor $v \in D$, em que o domínio tem tamanho k é codificado em um vetor $B = [0,\ldots, 0, 1, 0,\ldots, 0]$ de tamanho k na qual somente a *v-éssima* posição é 1, e o restante é 0.

Apresentamos dois mecanismos UE de acordo com a proporção do ruído aplicado: SUE e OUE, ambos garantem LDP para $\epsilon = ln\left(\frac{p(1-q)}{q(1-p)}\right)$. Ambos seguem o mesmo algoritmo de pertubação:, porém com cálculos distintos para p e q:

$$Pr\left[\Psi_{UE_{(\epsilon)}} B'[i] = 1\right] = \begin{cases} p, if B[i] = 1 \\ q, if B[i] = 0 \end{cases}$$

4.2.1 Symmetrical Unary Encoding (SUE)

SUE é um FO da classe UE, equivalente ao mecanismo RAPPOR simples (ERLINGSSON et al., 2014), no qual as variáveis p e q são selecionadas de acordo com as

equações $p = \frac{e^{\epsilon/2}}{e^{\epsilon/2}+1}$ e $q = \frac{1}{e^{\epsilon/2}+1}$, de modo que $p + q = 1$, logo SUE tem probabilidades de perturbação simétricas.

4.2.2 Optimized Unary Encoding (OUE)

Em "Locally differentially private protocols for frequency estimation"(WANG et al., 2017) se determina que quando o domínio é extenso, selecionar p e q tal que $p = \frac{1}{2}$ e $q = \frac{1}{e^{\epsilon}+1}$ leva aos melhores resultados, pois devido a grande quantidade bits 0 no vetor codificado, se torna mais adequado otimizar o mecanismo de perturbação para transmitir o máximo de bits 0 o possível.

4.3 Local Hashing (LH)

Uma vez que GRR funciona melhor quando o domínio é menor, uma forma de melhorar sua performance e facilitar a transmissão do sinal em aplicações reais é fazer uso de funções hash para diminuir o tamanho do domínio a ser processado pelo mecanismo. Assim é desenvolvido o mecanismo LH(), que segue o mesmo algoritmo de perturbação que GRR, porém de acordo com *g*, o tamanho do domínio reduzido, ao invés de *k*. LH possui duas variações BLH e OLH.

4.3.1 Binary Local Hashing (BLH)

BLH simplesmente utiliza funções hash que diminuem o domínio para um tamanho mínimo de $g = 2$, de forma a garantir máxima privacidade e redução do custo da transmissão do sinal. Contudo, reduzir o tamanho para o mínimo nem sempre é o mais adequado em termos de garantia da utilidade dos dados, podendo resultar em uma perturbação excessiva.

4.3.2 Optimized Local Hashing (OLH)

OLH procura reduzir o tamanho *k* do domínio para um *g* otimizado, que garante um melhor equilíbrio entre privacidade e utilidade. Em alguns casos é possível também que o

g otimizado seja igual a 2, de forma que OLH e BLH sejam equivalentes, porém este nem sempre é o caso. Para a escolha de um g otimizado, OLH faz uso da seguinte fórmula: $g = e^{\epsilon} + 1$.

5 FOs ADAPTADOS A DADOS LONGITUDINAIS

Nem todos FOs são adequados para o processamento de dados longitudinais, uma vez que para que LDP seja garantida, os dados devem ser sanitizados a cada *timestamp*, que equivalem a uma amostra. Assim, FOs que não levam em consideração esta propriedade de LDP aplicada a dados longitudinais podem perturbar excessivamente os dados levando a perda de utilidade, ou privacidade.

Uma estratégia comum para adaptar FOs para o processamento de dados longitudinais é o uso de duas rodadas de sanitização com memoização, uma abordagem primeiro apresentada em (ERLINGSSON et al., 2014) com RAPPOR. Onde primeiro o dado ou é sanitizado por meio de um algoritmo de perturbação similar aos dos FOs anteriores, garantindo um *budget* de privacidade referido como ϵ_{perm}, e em seguida é memoizado, ou caso já tenha sido memoizado, tem o valor ruidoso memoizado enviado em seu lugar. O valor memoizado ruidoso então passa para a segunda rodada de sanitização, sendo processado por um algoritmo de perturbação otimizado, com budget ϵ_1 (uma fração α de ϵ_{perm}), e só então enviado ao agregador. No agregador, os FOs adaptados para o processamento de dados longitudinais fazem uso de uma função de estimativa adaptada, que leva em consideração as duas rodadas de sanitização utilizadas.

Desde então, a mesma ideia foi aplicada para o desenvolvimento de outros FOs, construídos a partir da composição sequencial de mecanismos anteriores, e são o foco de nossos experimentos. Todos os FOs avaliados em nossos experimentos usam a seguinte função de estimativa: $\Phi fL\,(v) := \frac{C(v) - nq1(p2 - q2) - nq2}{n(p1 - q1)(p2 - q2)}$, onde $p1$ e $q1$ correspondem às probabilidades usadas na primeira rodada de sanitização, e $p2$ e $q2$ são utilizadas na segunda.

5.1 L-GRR (Longitudinal Generalized Randomized Response)

Uma adaptação do mecanismo GRR para o processamento de dados longitudinais (ARCOLEZI, 2022a). Utilizando de duas rodadas de sanitização com memoização, construído pela composição sequencial de duas instâncias do mecanismo original. O algoritmo de perturbação é equivalente ao do GRR para a primeira rodada de sanitização:

$$\forall_{x\in D}\ Pr\left[\Psi_{L-GRR_{(\epsilon\infty)}}(v)=x\right]=\begin{cases}p_1=\frac{e^{\epsilon}}{e^{\epsilon}+|D|-1} & \text{if x = v}\\ q_2=\frac{1-p}{|D|-1}=\frac{1}{e^{\epsilon}+|D|-1} & \text{if x} \neq \text{v}\end{cases}$$

e uma segunda rodada:

$$\forall_{x'\in D}\ Pr\left[\Psi_{L-GRR_{(\epsilon_1)}}(x)=x'\right]=\begin{cases}p_2 & \text{if x' = x}\\ q_2=\frac{1-p_2}{|D|-1} & \text{if x'} \neq \text{x}\end{cases}$$

na qual: $p_2=\frac{q_1-e^{\epsilon_1}p_1}{(-p_1e^{\epsilon_1})+|D|q_1e^{\epsilon_1}-q_1-p_1(|D|-1)+q_1}$,

uma vez que $\epsilon_1=ln\left(\frac{p_1p_2+q_1q_2}{p_1q_2+q_1p_2}\right)$ para L-GRR.

5.2 RAPPOR and L-SUE (Longitudinal Symmetric Unary Encoding)

A implementação de RAPPOR (ERLINGSSON et al., 2014) apresentada é a orientada a utilidade, equivalente ao protocolo L-SUE: se utiiza o protocolo SUE para as duas rodadas, os dados são codificados apenas uma vez antes da primeira rodada de sanitização. O algoritmo de pertubação para RAPPOR e outros mecanismos que usam de Unary Encoding pode ser formalmente descrito para a primeira rodada:

$$Pr\left[\Psi_{UE_{(\epsilon)}}B'[i]=1\right]=\begin{cases}p1, if B[i]=1\\ q1, if B[i]=0\end{cases}$$

para a segunda rodada:

$$Pr\left[\Psi_{UE_{(\epsilon)}}B'[i]=1\right]=\begin{cases}p2, if B[i]=1\\ q2, if B[i]=0\end{cases}$$

Por fim, todos mecanismos UE adaptados a dados longitudinais garantem que a seguinte equação seja satifeita:

$$\epsilon_1=ln(\frac{(p_1p_2-q_2(p_1-1))(p_2q_1-q_2(q_1-1)-1)}{(p_2q_1-q_2(q_1-1))(p_1p_2-q_2(p_1-1)-1)})$$

5.3 L-OUE (Longitudinal Optimized Unary Encoding)

Semelhante ao protocolo L-SUE, porém faz uso de OUE para as duas rodadas. L-OUE é propenso a adicionar ruído excessivo (ARCOLEZI et al. 2022a),), levando a uma perda significativa de utilidade ao longo do tempo. Segue o mesmo algoritmo de RAPPOR para a perturbação, porém com $p_1 = p_2 = 0.5$, $q_1 = \frac{1}{e^{\epsilon_{perm}}+1}$, e um q_2 que satisfaça o cálculo de ϵ_1.

5.4 L-OSUE (Longitudinal Optimized-Symmetric Unary Encoding)

L-OSUE, proposto em "Improving the utility of locally differentially private protocols for longitudinal and multidimensional frequency estimates"(ARCOLEZI et al. 2022a), é uma solução híbrida que usa a OUE para a primeira rodada e a SUE para a segunda rodada, evitando assim a adição excessiva de ruído ao longo do tempo, como acontece com os dados processados sob L-OUE. Probabilidades de pertubação dadas por $p_1 = 0.5$, $q_1 = \frac{1}{e^{\epsilon_{perm}}+1}$, e $p_2 + q_2 = 1$ que satisfaçam o cálculo de ϵ_1.

5.5 LOLOHA (Longitudinal Local Hashing)

LOLOHA (ARCOLEZI, 2022b) pode ser entedido como uma adaptação de Local Hashing para o processamento de dados longitudinais. Como não há sentido em fazer uso de uma função hash para reduzir o domínio mais de uma vez, após este primeiro passo, o mecanismo é construído a partir de duas instâncias de GRR configuradas para o domínio de tamanho reduzido *g*. Uma distinção entre LH e LOLOHA também está presente no cálculo de um *g* otimizado:

$$g = 1 + \max\left(1, \left\lfloor \frac{1 - a^2 + \sqrt{a^4 - 14a^2 + 12ab(1-ab) + 12a^3b + 1}}{6(a-b)} \right\rceil\right)$$

Um cálculo distinto do de OLH, que leva em consideração as duas rodadas de santização. g também pode ser definido simplesmente como $g = 2$, obtendo BiLOLOHA,

equivalente ao mecnaismo BLH. Uma instância de LOLOHA que faz uso de um valor otimizado é referida como OLOLOHA, equivalente a OLH. OLOLOHA procura balancear privacidade e utilidade, enquanto BiLOLOHA oferece maior privacidade e menor custo de transmissão do sinal de rede em aplicações reais.

6 SVIM

Para solucionar o problema descrito em (QIN et. al, 2016), uma série de abordagens foram propostas como LDPMiner, SVIM, e SVSM (uma abordagem que expande o problema original e procura encontrar os *k* conjuntos mais frequentes, e não apenas os itens). Contudo, neste trabalho decidimos focar na análise de SVIM (Set-Value Item Mining)(WANG et. al, 2018), que se mantém no escopo do problema original, e tem melhor performance que LDPMiner.

Para lidar com o desafio de não saber ao certo o tamanho dos conjuntos, SVIM faz uso da abordagem PSFO. Em PSFO , primeiramente se realiza os passos de *padding* e *sampling* (preenchimento e amostragem), onde cada usuário primeiro preenche seu conjunto de valores até atingir um tamanho *l* com valores fictícios, um valor aleatório é amostrado, e enviado para um FO, que é utilizado como uma caixa negra independente dos passos anteriores, e tem a sua frequência estimada com garantias LDP. Sem o passo de preenchimento, a probabilidade de um item ser amostrado é difícil de calcular e variável, diminuindo a utilidade do resultado final.

O algoritmo de SVIM inicia com a partição dos usuários em três grupos, cada grupo tendo um papel na execução do algoritmo. O primeiro grupo é utilizado para encontrar *2k* itens candidatos a *k* mais frequentes, com cada usuário reportando 1 item por conjunto para um PSFO configurado para $l = 1$. O segundo grupo é utilizado para estimar o tamanho do domínio dos itens candidatos, primeiro estimando a frequência dos *2k* itens candidatos no segundo grupo, e descartando valores a baixo do 90º percentil. O último grupo, recebendo o domínio ajustado *l* do segundo passo e os *2k* itens candidatos do primeiro, então realiza o cálculo de estimativa de frequência de forma privada por meio de um PSFO uma última vez. Essas estimativas são ajustadas em um último passo para considerar valores que possam ter sido subestimados no primeiro e no segundo passo. Por fim, SVIM retorna os *k* mais frequentes valores e suas estimativas, dentre os resultados do último passo.

Uma aplicação ainda não testada de SVIM, seria a dados longitudinais. Os experimentos com SVIM apresentados na literatura até então se utilizam de mecanismos não adaptados ao processamento de dados longitudinais. Nossa contribuição se dá em uma análise dos resultados de SVIM processando dados longitudinais com FOs adaptados para a tarefa em questão.

7 ANÁLISE EXPERIMENTAL

Em nossos experimentos, quatro conjuntos de dados distintos foram usados para analisar o desempenho dos mecanismos usados de acordo com diferentes cenários. Implementamos nosso código em Python 3.12. Todos os experimentos foram realizados em um servidor com Ubuntu 20.04, Intel Core i7-7820X e 128 GB de memória.

7.1 Datasets

Os datasets utilizados e suas propriedades estão listadas a seguir:

- Bfive: Um conjunto de dados que apresenta resultados de testes de personalidade. Cada resultado foi tratado com um conjunto de valores, e interpolamos os conjuntos para obter 10 timestamps. Os conjuntos possuem um tamanho fixo de 5;
- BMS-POS: Dataset de transações comerciais de meio milhão de usuários e 1657 categorias. Interpolamos os conjuntos para obter 15 timestamps. Os conjuntos possuem tamanhos de 1 a mais de 20.
- Loan: Mostramos aleatoriamente e interpolamos valores de um quinto do dataset de empréstimos lending club. Resultando em 5 timestamps, conjuntos que variam de 1 a 3 itens.
- Kosarak, um dataset de cliques em streams de um site húngaro com cerca de um milhão de usuários e 42 mil categorias. Interpolamos os conjuntos para obter 5 timestamps. Os conjuntos possuem tamanhos variados e o tamanho do domínio é o maior de todos datasets usados.

7.2 Configuração dos experimentos

Selecionamos um intervalo de valores para o orçamento de privacidade iniciando em 0.5 até 4, em incrementos 0.5, resultando em 8 valores de ϵ_{perm}, e $\alpha = 0.6$ (relativamente alto, levando em consideração a adição de ruído a mais pelo passo de *padding* e *sampling* de SVIM) para $\epsilon_1 = 0.6\epsilon_{perm}$ a serem utilizados na segunda rodada de sanitização.

8 RESULTADOS

Fazemos uso do erro quadrático médio (MSE, do inglês *mean squared error*) para medir o quanto diferem os resultados anonimizados dos reais: isto é, quão maior o MSE, menor a utilidade e pior os resultados. Primeiro identificamos se os *k* itens mais frequentes retornados por SVIM correspondem ao resultado da consulta sob dados não-anonimizados. Itens frequentes não identificados, e itens erroneamente classificados como entre os *top-k* ambos tem suas frequências adicionadas ao cálculo do erro. O restante do erro é calculado pela diferença entre as estimativas e as frequências reais dos *top k* itens identificados corretamente.

De forma mais formal: $MSE = \frac{1}{|D|}(f(x)^2 + f(y)^2 + (f(v) - f'(v))^2)$, onde *f(x)* são as frequências de itens não identificados, *f(y)* incorretamente classificados como *k* mais frequentes, e dentre as frequências dos *top k* itens identificados corretamente *f(v)* e *f'(v)* são o valor real e a saída de SVIM respectivamente. Por fim, a soma dos quadrados de todos erros é dividida por |*D*|, onde o *D* é o conjunto de todos valores de x, y, e v.

Para calcular o *MSE* para dados longitudinais, simplesmente calculamos o *MSE* médio entre consultas para cada timestamp $t \in \tau$ onde τ é o conjunto de timestamps. Formalmente: $MSE_{avg} = \frac{1}{\tau}\sum_{t=1}^{\tau} MSE_t$.

8.1 Bfive

Em nossos testes com o dataset Bfive, analisamos primeiro os protocolos L-GRR, OLOLOHA, e BiLOLOHA. Encontramos que l-grr possui utilidade significativamente menor que ambas instâncias de LOLOHA. OLOLOHA, apresentou melhor ou igual desempenho que BiLOLOHA para maioria dos budgets.

Entre os protocolos UE, RAPPOR e L-OUE apresentaram performance similar, enquanto L-OSUE superou o desempenho de ambos. Entre os dois melhores, OLOLOHA e L-OSUE, o segundo apresentou resultados no geral melhores.

Gráfico 1 – Comparativo do MSE_{avg} das abordagens L-GRR, OLOLOHA, e BiLOLOHA para o dataset Bfive.

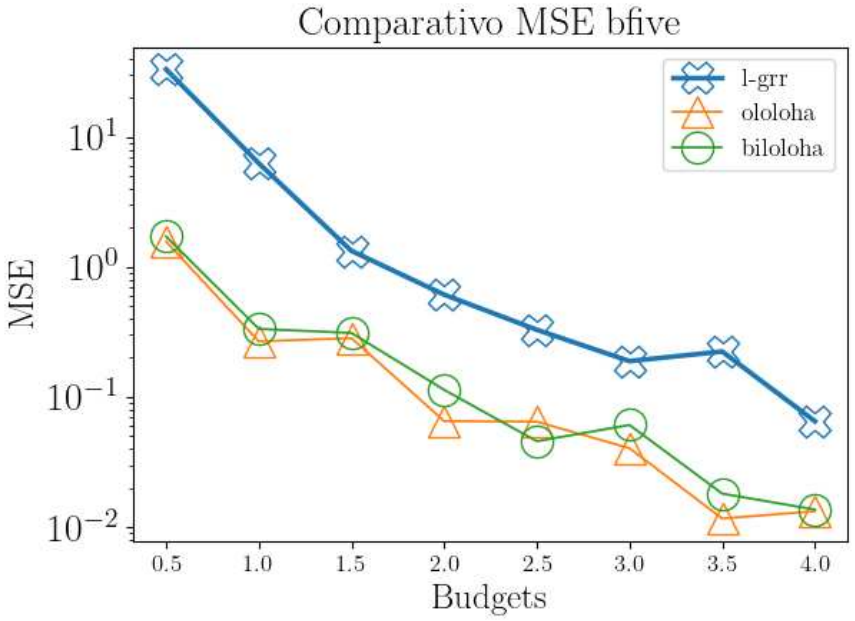


Gráfico 2 – Comparativo do MSE_{avg} das abordagens RAPPOR, L-OSUE, e L-OUE para o dataset Bfive.

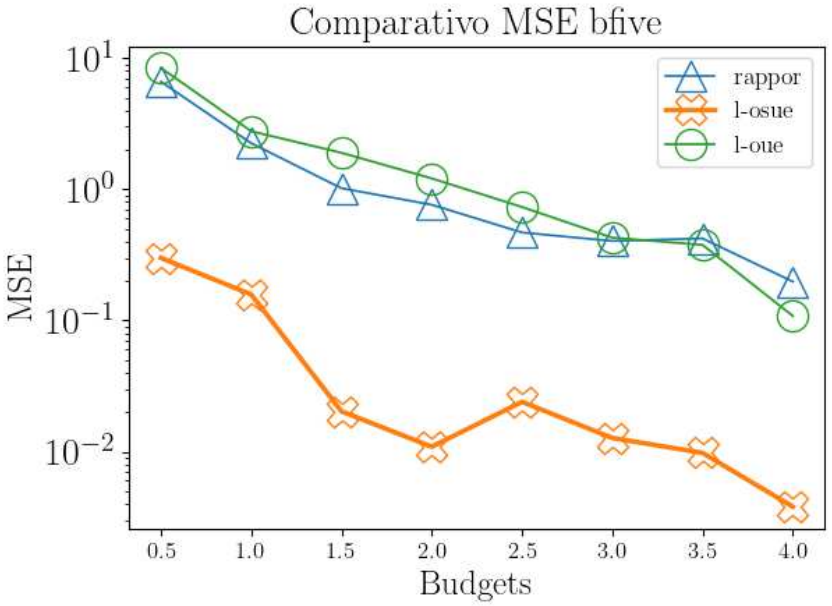
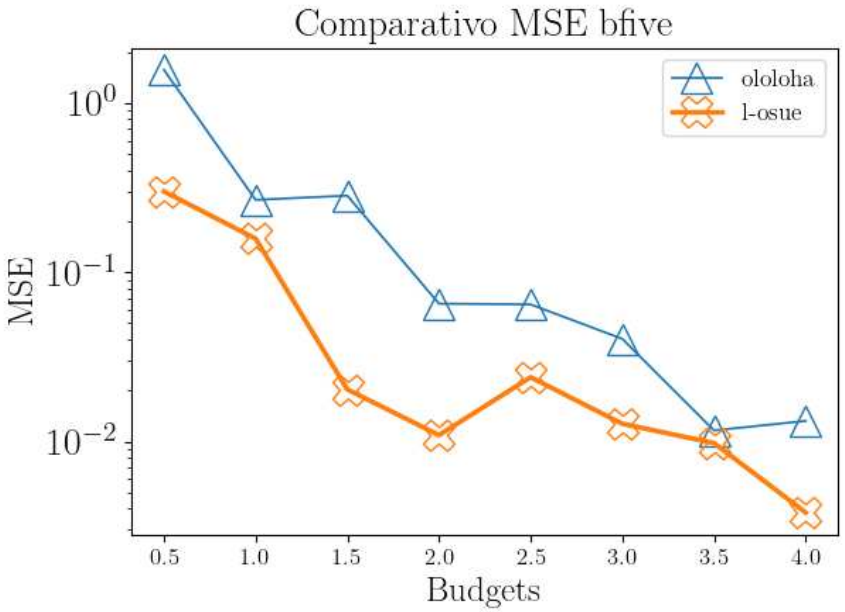


Gráfico 3 – Comparativo do MSE_{avg} das abordagens OLOLOHA e L-OSUE para o dataset Bfive.



8.2 BMS-POS

Para o dataset BMS-POS, encontramos entre os protocolos derivados de GRR, encontramos uma maior variação de utilidade em relação ao budget para L-GRR. Porém como é interessante garantir maior utilidade para um budget pequeno, consideramos os resultados apresentados por OLOLOHA mais interessantes, apresentando desempenho similar porém na maioria dos casos melhor que BiLOLOHA.

Com relação aos protocolos UE, RAPPOR apresentou maior variação de utilidade de acordo com o budget, L-OSUE e L-OUE pouco variaram o MSE para o intervalo de ϵ_{perm} escolhido, com L-OSUE mantendo o melhor desempenho entre os dois no geral. Pelo mesmo raciocínio da análise anterior, consideramos que L-OSUE apresentou os melhores resultados. Entre os dois melhores, OLOLOHA apresentou os menor MSE.

Gráfico 4 – Comparativo do MSE_{avg} das abordagens L-GRR, OLOLOHA, e BiLOLOHA para o dataset BMS-POS.

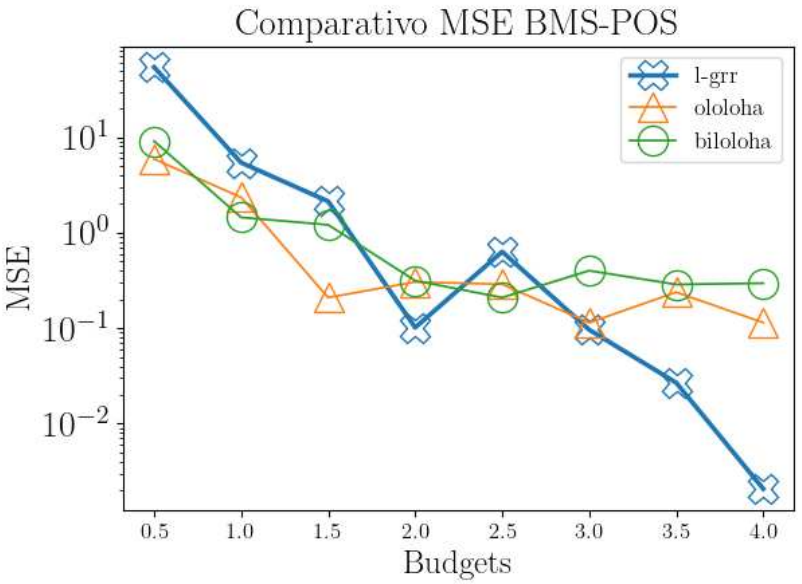


Gráfico 5 – Comparativo do MSE_{avg} das abordagens RAPPOR, L-OSUE, e L-OUE para o dataset BMS-POS.

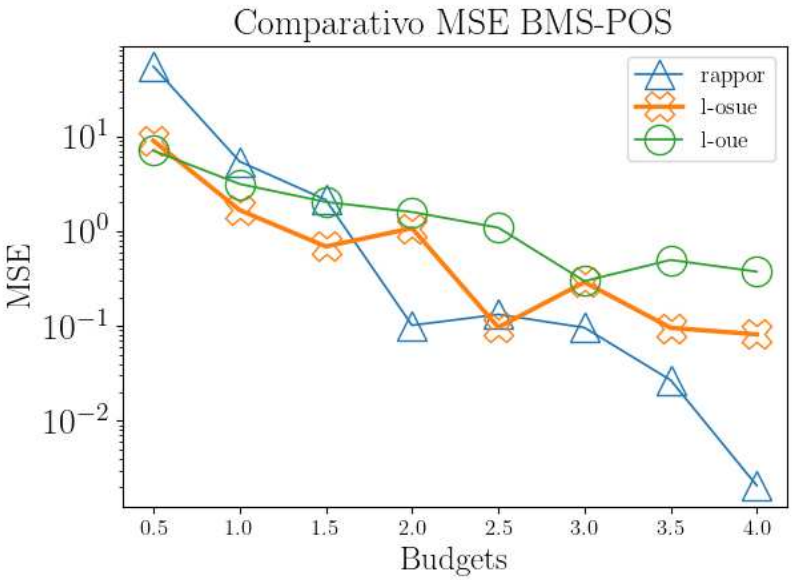
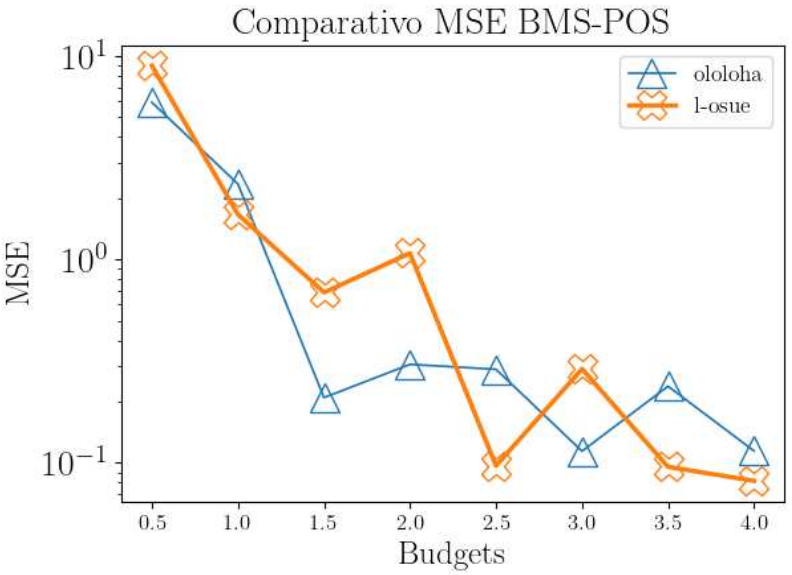


Gráfico 6 – Comparativo do MSE_{avg} das abordagens OLOLOHA e L-OSUE para o dataset BMS-POS.



8.3 Loan

Nos experimentos com o dataset Loan, L-GRR, OLOLOHA, e BiLOLOHA apresentaram MSE significativamente mais alto do que em experimentos anteriores. BiLOLOHA apresentou resultados no geral um pouco melhores que OLOLOHA, e ambos muito melhores que L-GRR. RAPPOR apresentou uma grande variação para o intervalo de budgets, enquanto L-OUE permaneceu relativamente estável. L-OSUE manteve uma melhor performance que L-OUE ao longo de incrementos no budget, com maior MSE somente para o menor. Em comparação com BiLOLOHA, L-OSUE foi o mais promissor, especialmente quando comparado com resultados para budgets menores.

Gráfico 7 – Comparativo do MSE_{avg} das abordagens L-GRR, OLOLOHA, e BiLOLOHA para o dataset Loan.

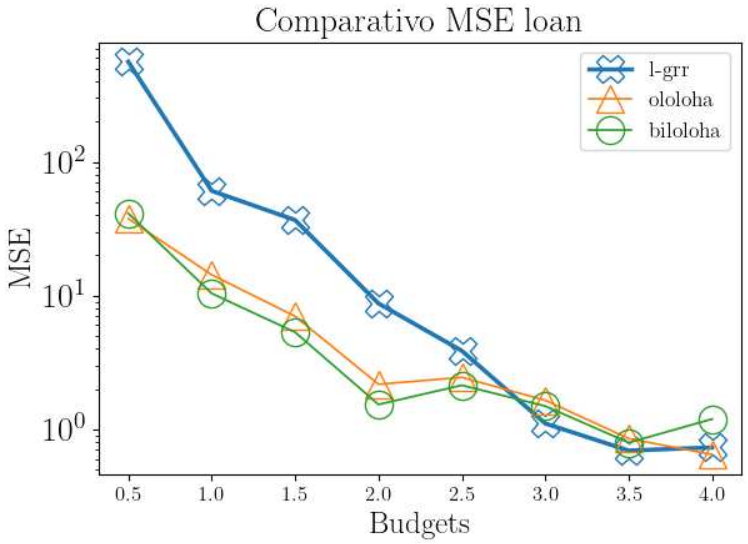


Gráfico 8 – Comparativo do MSE_{avg} das abordagens RAPPOR, L-OSUE, e L-OUE para o dataset Loan.

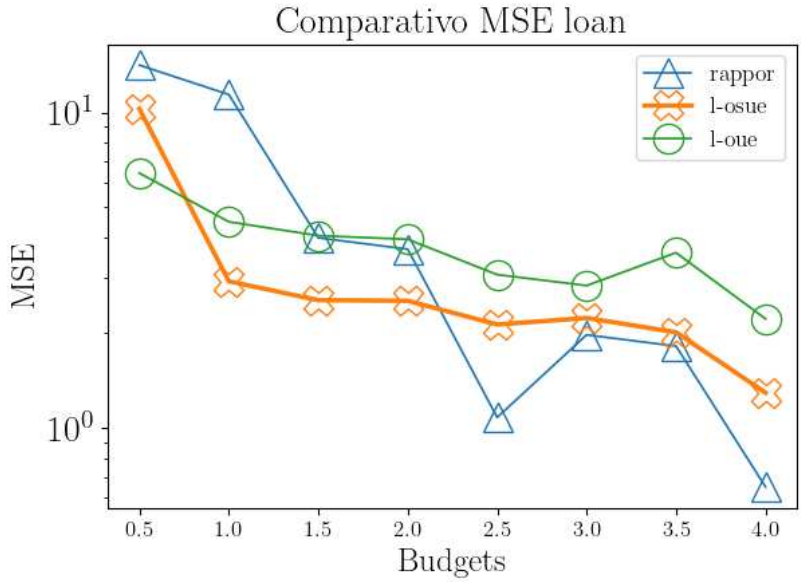
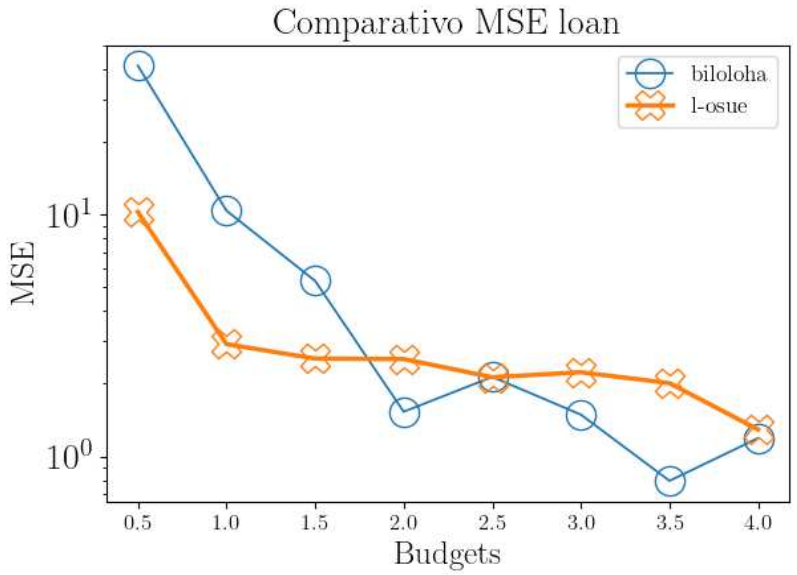


Gráfico 9 – Comparativo do MSE_{avg} das abordagens BiLOLOHA e L-OSUE para o dataset Loan.



8.4 Kosarak

Em Kosarak, encontramos resultados similares aos experimentos com o dataset Loan, com BiLOLOHA apresentando desempenho melhor que OLOLOHA. No geral, L-GRR só apresentou boa performance para os maiores valores de ϵ_{perm}, e OLOLOHA teve melhor performance quando o tamanho do domínio era menor.

Os protocolos UE apresentaram resultados similares para os menores valores do budget, este também foi o único experimento que L-OUE e RAPPOR tiveram melhorm desempenho que L-OSUE. O domínio mais esparso levou a melhores resultados de L-OUE, enquanto L-OSUE só teve melhor desempenho dentre budgets menores que RAPPOR qando processando Bfive, um dataset com dominío pequeno. L-OUE também foi um pouco melhor que BiLOLOHA para os menores budgets.

Gráfico 10 – Comparativo do MSE_{avg} das abordagens L-GRR, OLOLOHA, e BiLOLOHA para o dataset Kosarak.

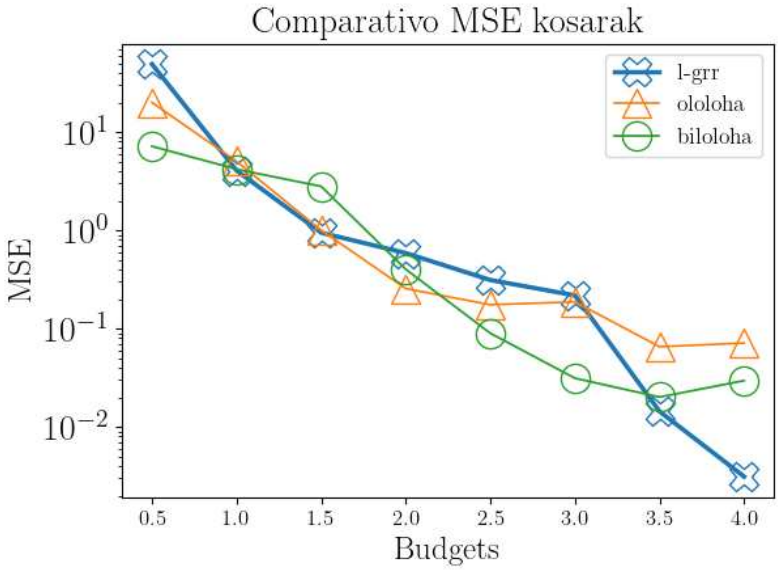


Gráfico 11 – Comparativo do MSE_{avg} das abordagens RAPPOR, L-OSUE, e L-OUE para o dataset Kosarak.

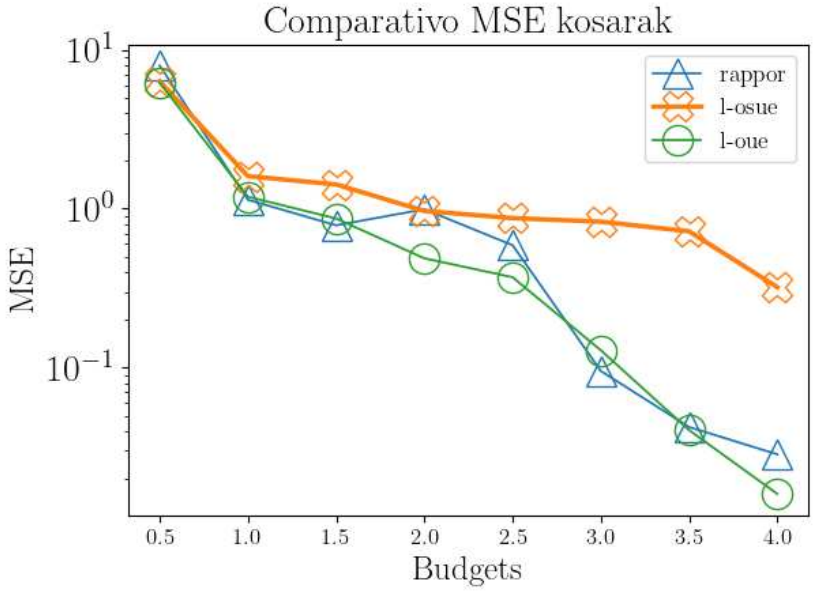
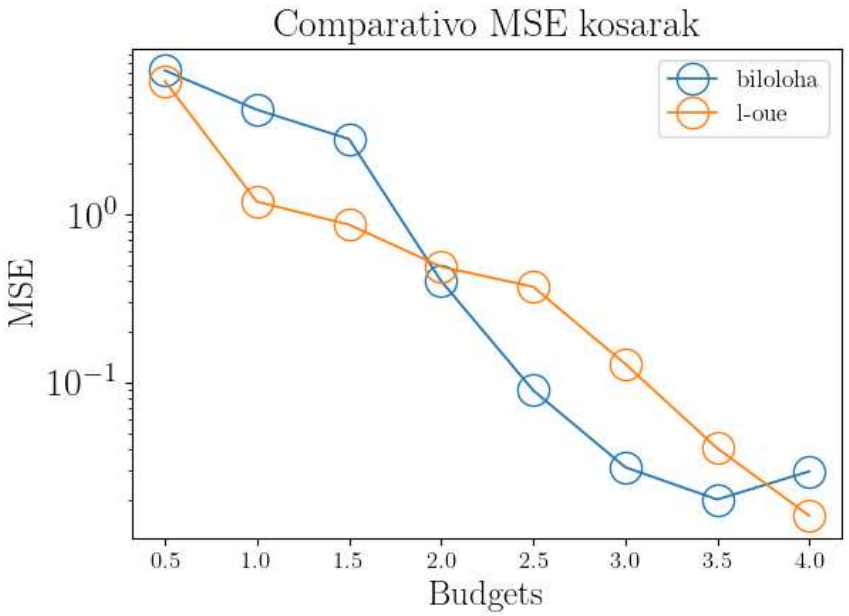


Gráfico 12 – Comparativo do MSE_{avg} das abordagens BiLOLOHA e L-OUE para o dataset Kosarak.



9 CONCLUSÃO

Neste trabalho, conduzimos uma extensa série de experimentos com óraculos de frequência adaptados ao processamento de dados longitudinais, em conjunto com a abordagem state-of-the-art SVIM com o objetivo de determinar quais mecanismos eram os mais promissores na execução da tarefa de encontrar os *top k* itens mais frequentes em um banco de dados longitudinal.

Encontramos desempenho similar entre os mecanismos OLOLOHA e BiLOLOHA, especialmente quando processando datasets em que o domínio dos valores era grande, quando BiLOLOHA teve desempenho melhor. L-OUE teve boa performance para datasets mais esparsos, mas no geral L-OSUE foi a abordagem mais promissora dentre os FOs que usavam de codificação unária, apesar de ter a pior performance em experimentos com o dataset Kosarak. Com a exceção dos resultados de BMS-POS, FOs UE apresentaram menor MSE que as duas versões de LOLOHA. Por fim, L-GRR foi consistentemente o pior dentre os FOs testados.

Concluímos que SVIM em conjunto com L-OSUE foi a melhor abordagem no geral, porém OLOLOHA e BiLOLOHA apresentaram resultados mais consistentes e desempenho próximo do melhor dentre todos os datasets. Em um cenário em que o agregador não tenha qualquer conhecimento prévio de atributos como tamanho do domínio dos dados a serem analisados, OLOLOHA ou BiLOLOHA podem ser abordagens mais adequadas que L-OSUE. Entre OLOLOHA e BiLOLOHA, a performance similar de ambos nos leva a considerar que a nível de garantias LDP e utilidade ambos são efetivamente equivalentes para esta tarefa, com BiLOLOHA sendo a preferência devido a questões de custo computacional, uma vez que não há necessidade de calcular um g otimizado, e $g=2$ torna a transmissão do sinal do santizador até o agregador mais barata.

Para trabalhos futuros, consideramos estudar o comportamento de protocolos com duas rodadas de sanitização quando utilizados em conjunto com SVSM, a extensão de SVIM para a mineração de itemsets, e outras aplicações de LDP.

REFERÊNCIAS

ARCOLEZI, Héber H. et al. Improving the utility of locally differentially private protocols for longitudinal and multidimensional frequency estimates. Digital Communications and Networks (2022) ,2022.

ARCOLEZI, Héber H. et al. Frequency estimation of evolving data under local differential privacy. arXiv preprint arXiv:2210.00262, 2023.

BRASIL. Lei Nº 13.709 - Lei Geral de Proteção de Dados Pessoais (LGPD). 2018. http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm. Acessado em: 09-02-2025.

CALIFORNIA. California Consumer Privacy Act (CCPA). 2018. https://www.caprivacy.org/. Acessado em: 09-02-2025.

DA COSTA FILHO, José Serafim; MACHADO, Javam C. FELIP: A local Differentially Private approach to frequency estimation on multidimensional datasets. In: EDBT. 2023. p. 671-683

DWORK, Cynthia et al. Calibrating noise to sensitivity in private data analysis. In: Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3. Springer Berlin Heidelberg, 2006. p. 265-284.

DWORK, Cynthia et al. The algorithmic foundations of differential privacy. Foundations and Trends® in Theoretical Computer Science, v. 9, n. 3–4, p. 211-407, 2014.

ERLINGSSON, Úlfar; PIHUR, Vasyl; KOROLOVA, Aleksandra. Rappor: Randomized aggregatable privacy-preserving ordinal response. In: Proceedings of the 2014 ACM SIGSAC conference on computer and communications security. 2014. p. 1054-1067.

General Data Protection Regulation. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46. Official Journal of the European Union, v. 59, p. 1–88, 2016.

KAIROUZ, Peter; BONAWITZ, Keith; RAMAGE, Daniel. Discrete distribution estimation under local privacy. In: International Conference on Machine Learning. PMLR, 2016. p. 2436-2444.

QIN, Zhan et al. Heavy hitter estimation over set-valued data with local differential privacy. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security. 2016. p. 192-203.

WANG, Tianhao et al. Locally differentially private protocols for frequency estimation. In: 26th USENIX Security Symposium (USENIX Security 17). 2017. p. 729-745.

WANG, Tianhao et al. Continuous release of data streams under both centralized and local

differential privacy. In: Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. 2021. p. 1237-1253.

WANG, Tianhao; LI, Ninghui; JHA, Somesh. Locally differentially private frequent itemset mining. In: 2018 IEEE Symposium on Security and Privacy (SP). IEEE, 2018. p. 127-143.

WARNER, Stanley L. Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American statistical association, v. 60, n. 309, p. 63-69, 1965.

ZHU, Youwen et al. Heavy hitter identification over large-domain set-valued data with local differential privacy. IEEE Transactions on Information Forensics and Security, 2023.