Reveal-or-Obscure: A Differentially Private Sampling Algorithm

Naima Tasnim [1] **Atefeh Gilani** [1] **Lalitha Sankar** [1] **Oliver Kosut** [1]

Abstract

We introduce a differentially private algorithm called reveal-or-obscure (ROO) to generate a single representative sample from a dataset of n observations drawn i.i.d. from an unknown discrete distribution P. Unlike methods that add explicit noise to the estimated empirical distribution, ROO achieves ϵ-differential privacy (DP) by randomly choosing whether to "reveal" or "obscure" the empirical distribution. While ROO is structurally identical to an algorithm proposed by (Cheu & Nayak, 2025), we prove a strictly better bound on the sampling complexity than that established in Theorem 12 of (Cheu & Nayak, 2025). To further improve the privacy-utility trade-off, we propose a novel generalized sampling algorithm called Data-Specific ROO (DS-ROO), where the probability of obscuring the empirical distribution of the dataset is chosen adaptively. We prove that DS-ROO satisfies ϵ-DP, and provide empirical evidence that DS-ROO can achieve better utility under the same privacy budget of vanilla ROO.

1. Introduction

The widespread use of sensitive data across various domains, including healthcare, finance, law enforcement, and social sciences, has heightened the importance of privacy-preserving data analysis. Consequently, there is a growing need for mechanisms that allow data analysis while minimizing individual privacy risks. One promising approach is the use of synthetic data that capture the statistical properties of the original data.

Differential Privacy (DP) (Dwork et al., 2006; 2014) has emerged as a sound framework for formalizing privacy guarantees across a range of applications, including data analysis. In essence, DP ensures that the output of an algorithm does not differ by much whether or not an individual's data is included in the input.

The task of synthetic data generation is closely related to the broader problem of learning probability distributions, which has been studied under both central and local DP; for example, see (Diakonikolas et al., 2015; Kamath et al., 2019; Acharya et al., 2020a;b). In the non-private setting, a learning algorithm approximates a distribution from which one can sample new data points that are representative of the original data. When privacy constraints are introduced, learning a distribution becomes significantly more challenging. In many practical cases, it may be sufficient to produce a small number of representative samples instead. The task of privately releasing one sample—known as DP sampling—is easier than full-fledged learning, since it requires less information from the underlying distribution. Motivated by this, we propose a novel DP sampling algorithm for discrete distributions on a finite alphabet. The key idea of our approach is to "obscure" the empirical distribution of the input dataset with a certain probability, or "reveal" it otherwise. Hence, we call our proposed algorithm reveal-or-obscure (ROO).

Main Contributions. Our main contributions are:

- We propose ROO—a sampling algorithm that achieves differential privacy without *explicitly* perturbing the empirical distribution of the input dataset. We incorporate uncertainty in our algorithm by sampling from the uniform distribution with some fixed probability q.
- We prove that our proposed algorithm reduces the sampling complexity while achieving better privacy-utility trade-off than the state-of-the art (Raskhodnikova et al., 2021), (Cheu & Nayak, 2025).
- We also propose DS-ROO (data-specific ROO) as a technique to generalize ROO by making q, i.e., the probability of sampling from the uniform distribution, a function of the empirical distribution of the dataset. We prove that it is possible to achieve the same privacy guarantee with a lower q value relative to the vanilla ROO algorithm for sufficiently large datasets.
- We demonstrate empirically that, for the same privacy guarantee, DS-ROO achieves significantly better utility than vanilla ROO as well as the state-of-the-art (Raskhodnikova et al., 2021), (Cheu & Nayak, 2025).

[1]School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, Arizona, United States. Correspondence to: Naima Tasnim <ntasnim2@asu.edu>.

Published at the ICML 2025 Workshop on Machine Unlearning for Generative AI. Copyright 2025 by the author(s).

Related Work. The problem of differentially private sampling from unknown distributions is first investigated in (Raskhodnikova et al., 2021). Raskhodnikova *et al.* (Raskhodnikova et al., 2021) provide the first known bounds with (ϵ, δ)-DP guarantees on the complexity of sampling from arbitrary distributions over a discrete alphabet. DP algorithms for sampling from higher dimensional distributions such as multivariate Gaussians are presented in (Ghazi et al., 2024). Husain *et al.* (Husain et al., 2020) considers DP sampling in the local setting, where the central aggregator cannot be trusted and each user must produce a single data record privately. Private sampling has also been studied in the local model; recent works propose minimax-optimal mechanisms for all f-divergences (Park et al., 2024), and leverage public data to improve utility under local DP (Zamanlooy et al., 2024). A key focus of these efforts has been to reduce the sample complexity of DP-assured private sampling. While a recent and concurrent work (Cheu & Nayak, 2025) independent from ours proposes an algorithm structurally identical to ROO, our analysis establishes a strictly better sampling complexity bound in the same setting. More generally, the problem of releasing a dataset in a differentially private manner has also been studied; for example, see (Bellovin et al., 2019; Hardt et al., 2012; Zhu et al., 2017; Majeed & Lee, 2020; Boedihardjo et al., 2022). Generating a single sample in a private manner is the first step towards releasing a larger synthetic dataset, and to this end, we focus on the former challenge in this paper.

2. Problem Setup

We begin by briefly reviewing some relevant definitions. We use uppercase letters, e.g., X, to denote random variables (RVs), and lowercase letters, e.g. x, for their instantiations. We assume that the dataset consists of n RVs sampled from a finite alphabet of k letters; without loss of generality, we take this alphabet to be $[k] = \{1, 2, \ldots, k\}$. Let $\mathcal{P}$ be the class of probability distributions on $[k]$. Given a dataset $X^n = (X_1, X_2, \ldots, X_n)$ of n i.i.d. observations from some unknown $P \in \mathcal{P}$, a randomized algorithm (privacy mechanism) $\mathcal{A} : \mathcal{X}^n \mapsto \mathcal{X}$ outputs a single sample from $\mathcal{X}$. Let $Y = \mathcal{A}(X^n)$ be the random variable corresponding to the output of algorithm $\mathcal{A}$ given input X^n. The output $\mathcal{A}(X^n)$ is drawn from a distribution Q such that

$$Q(y) = \sum_{x^n \in \mathcal{X}^n} \Pr\left\{\mathcal{A}(x^n) = y | x^n\right\} \Pr\left\{X^n = x^n\right\}. \quad (1)$$

The accuracy of $\mathcal{A}$ is measured by the *closeness* between Q and P. We use the *total variation distance*, defined as

$$d_{TV}(Q, P) = \frac{1}{2}\|Q - P\|_1 = \frac{1}{2}\sum_x |Q(x) - P(x)|. \quad (2)$$

We use the following definition of sampling accuracy, introduced in (Axelrod et al., 2020).

Definition 2.1 (Accuracy of Sampling)**.** An algorithm $\mathcal{A}$ is α-accurate on a distribution P if the total variation distance, d_{TV} between Q and P is bounded by some constant α, i.e.,

$$d_{TV}(Q, P) \leq \alpha. \quad (3)$$

An algorithm is α-accurate on a class $\mathcal{P}$ of distributions if it is α-accurate on every $P \in \mathcal{P}$.

Two datasets x^n and $\tilde{x}^n$ are considered *neighbors*, denoted $x^n \sim \tilde{x}^n$, if they differ by at most one entry. DP is defined with respect to all such neighboring datasets as follows.

Definition 2.2 (Differential Privacy (Dwork et al., 2006))**.** A randomized algorithm, or mechanism $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ is considered ϵ-differentially private (ϵ-DP) if, for every pair of neighboring datasets $x^n \sim \tilde{x}^n \in \mathcal{X}^n$, and for all $Y \subseteq \mathcal{Y}$,

$$\Pr\{\mathcal{A}(x^n) \in Y\} \leq e^{\epsilon} \Pr\{\mathcal{A}(\tilde{x}^n) \in Y\}. \quad (4)$$

In (Raskhodnikova et al., 2021), the authors present an achievable ϵ-DP sampler which does the following:
(i) computes, for each $j \in [k]$, the empirical probability distribution $\hat{p}_j$,
(ii) adds noise drawn from Lap$(\frac{2}{n\epsilon})$ to each count,
(iii) uses an L_1 projection to restrict the noisy distribution to be a probability vector $\tilde{P} = (\tilde{p}_1, \ldots, \tilde{p}_k)$, and
(iv) outputs an element of $[k]$ sampled from $\tilde{P}$.
They show that this algorithm is α-accurate with a sampling complexity

$$n' = \frac{2k}{\alpha\epsilon}. \quad (5)$$

Cheu and Nayak (Cheu & Nayak, 2025) propose Subsampled Randomized Response (SubRR), which selects a data point uniformly at random from the input dataset and applies k-ary randomized response (Warner, 1965; Dwork et al., 2006). In Theorem 12 of (Cheu & Nayak, 2025), they show that their algorithm is α-accurate with a reduced sampling complexity

$$n'' = \frac{1}{\alpha\epsilon}(k-1)(1-\alpha). \quad (6)$$

In the following section, we prove that it is possible to achieve an ϵ-DP and α-accurate sampler with fewer samples than both n' and n''. Notably, for higher values of ϵ, i.e., lower privacy, we gain an exponential reduction in the required number of samples. Moreover, (Raskhodnikova et al., 2021) also establishes a lower bound on the sampling complexity as $\Omega\left(\frac{k}{\alpha\epsilon}\right)$ for a restricted range of $\epsilon \in (0, 1]$.

3. Reveal-or-Obscure (ROO)

Algorithm 1 presents our proposed private sampler ROO. We implement the idea of obscuring the empirical distribution $\hat{P}_{x^n}$ by sampling from the uniform distribution on $[k]$.

However, we wish to do so with a small probability q, so that we do not deviate too much from the true distribution P. With probability $1-q$, we simply choose a sample from the given dataset, i.e., we reveal $\hat{P}_{x^n}$.

Algorithm 1 Reveal-or-Obscure (ROO)

Input: Dataset $x^n = (x_1, \ldots, x_n)$, alphabet size k, privacy budget ϵ, parameter q
Output: Sample y
With probability q, choose $y \sim \text{Unif}[1:k]$
Otherwise, pick $i \sim \text{Unif}[1:n]$ and choose $y = x_i$
return y

The privacy and utility guarantees provided by Algorithm 1 is given by the following theorem.

Theorem 3.1. *Given q, Algorithm 1 is ϵ-DP and α-accurate for*

$$\epsilon = \log\left(1 + \frac{k(1-q)}{nq}\right), \text{ and} \tag{7}$$

$$\alpha = q\left(1 - \frac{1}{k}\right), \tag{8}$$

from which we can solve for q to obtain the sampling complexity as

$$n = \frac{k(1-\alpha) - 1}{\alpha(e^{\epsilon} - 1)}. \tag{9}$$

Lemma 3.2. *For any $k \geq 2, \epsilon > 0$, and $\alpha \in \left(0, 1 - \frac{1}{k}\right)$, the sampling complexity of Algorithm 1 is lower than that of (Raskhodnikova et al., 2021) and (Cheu & Nayak, 2025).*

The proofs are provided in Appendix A.

In fact, the sampling complexity of Algorithm 1 is exponentially better in terms of ϵ compared to that of (Raskhodnikova et al., 2021) in (5), and (Cheu & Nayak, 2025) in (6). This would appear to violate the lower bound of (Raskhodnikova et al., 2021) on the sampling complexity; the reason it does not is that their lower bound only applies for $\epsilon < 1$.

4. Data-Specific Reveal-or-Obscure (DS-ROO)

For the ROO algorithm in Section 3, we fix q in order to achieve ϵ-DP for any possible dataset. From (19) in the privacy analysis of Appendix A, we observe that the supremum of ratio of probabilities is achieved by setting $p = 0$. This corresponds to the case where one of two neighboring datasets under consideration is entirely missing an element of the alphabet, but this element is present in the other dataset. Thus, ROO is inefficient on datasets where each element of the alphabet appears reasonably often. In this section, we show that by making q a function of the dataset—specifically, a function of the smallest empirical probability,

Algorithm 2 Data-specific ROO (DS-ROO)

Input: Dataset $x^n = (x_1, \ldots, x_n)$, alphabet size k, privacy budget ϵ
Output: Sample y
Compute

$$u' = -1 + \frac{1}{k} - \frac{1}{n}, v' = e^{\epsilon}\left(\frac{1}{k} - 1\right), w' = e^{\epsilon} - 1 - \frac{1}{n};$$
$$q_0 = \frac{1}{1 + \frac{n}{k}\left(e^{\epsilon} - 1\right)}; \tag{10}$$

for $m = 1, 2, \ldots, \left\lfloor \frac{n}{k} \right\rfloor$ **do**
Compute

$$u_m = -\frac{m}{n} + \frac{1}{k} - \frac{1}{n}, v_m = e^{\epsilon}\left(\frac{1}{k} - \frac{m}{n}\right),$$
$$w_m = -\frac{1}{n} - \frac{m}{n} + \frac{m}{n}e^{\epsilon};$$
$$q_m = \max\left\{0, \frac{u_m}{v_m}q_{m-1} - \frac{w_m}{v_m}, \frac{v'}{u'}q_{m-1} + \frac{w'}{u'}\right\}; \tag{11}$$

end for
Compute $m = n \cdot \min_x \hat{P}_{x^n}(x)$;
With probability q_m, choose $y \sim \text{Unif}[1:k]$;
Otherwise, pick $i \sim \text{Unif}[1:n]$ and choose $y = x_i$;
return y;

the accuracy can be improved for the same privacy. Let $m \in \left\{0, 1, \ldots, \left\lfloor \frac{n}{k} \right\rfloor\right\}$ denote the smallest number of times an element of $[k]$ appears in dataset x^n. Mathematically, m can be expressed as $m = n \cdot \min_x \hat{P}_{x^n}(x)$. The modified private sampler—which we call the data-specific reveal-or-obscure (DS-ROO)—is presented in Algorithm 2. In DS-ROO, the probability q_m is determined from the value of m associated with the given dataset.

Notably, when $m = 0$, the corresponding q_0 is equivalent to that of Algorithm 1. That is, the case of a dataset that is missing an element of the alphabet (so $m = 0$) is the worst-case scenario, and so in this case DS-ROO behaves identically to ROO. However, we will show empirically that in other cases, DS-ROO can do much better than vanilla ROO.

Fig. 1 shows the function q_m for $k = 10, n = 1000$, and several different values of the privacy parameter ϵ. We observe that q_m is non-increasing in m. As m increases, the empirical distribution of the dataset gets closer to the uniform distribution, and q_m approaches zero. Thus, DS-ROO is less likely to obscure the empirical distribution for larger m. In high privacy regimes, q_m tends to decrease much slower. On the other hand, for larger ϵ, i.e., low

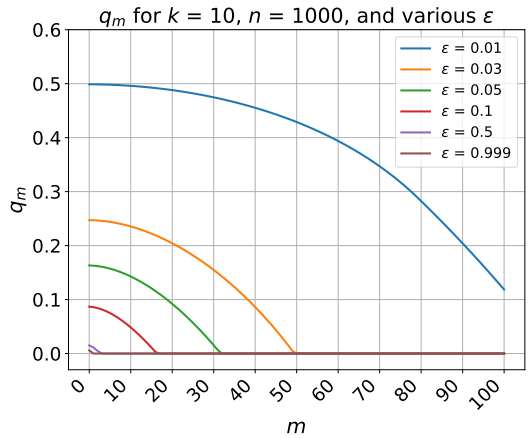

Figure 1. Plot of q_m as a function of m for fixed k and n, showing changes under different privacy budgets ϵ.

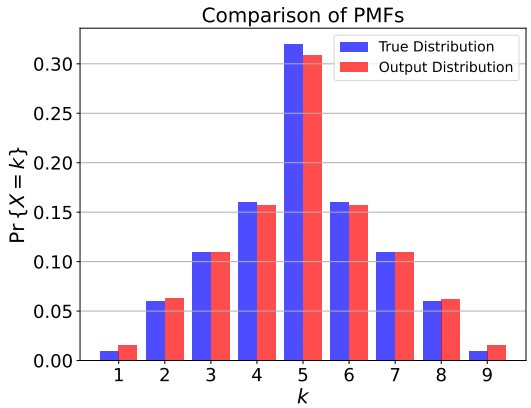

Figure 2. Comparison of true distribution and the estimated output distribution for DS-ROO.

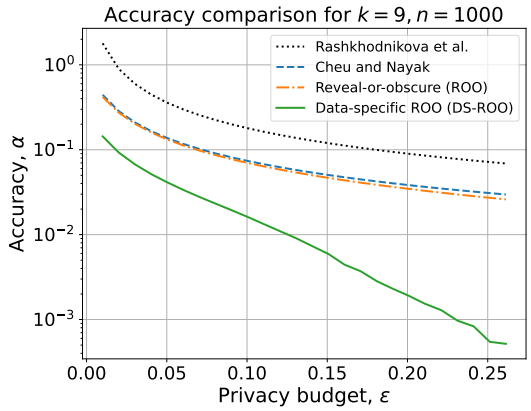

Figure 3. Comparison of accuracy (α) versus privacy (ϵ) curves of differentially private sampling algorithms.

privacy, q_m goes to zero much faster. The privacy guarantee provided by DS-ROO is given by the following theorem.

Theorem 4.1. *Algorithm 2 is ϵ-differentially private.*

The proof is provided in Appendix B.

4.1. Utility of DS-ROO

We do not have theoretical bounds on the utility of DS-ROO at this time. However, we provide empirical evidence that DS-ROO achieves better utility than vanilla ROO and the state-of-the-art samplers for the same privacy guarantee. In order to measure the utility of DS-ROO, we consider an input distribution, estimate the corresponding output distribution according to Algorithm 2, and compute the total variation distance. Fig. 2 shows an example case for a distribution on an alphabet of size $k = 9$, with dataset size $n = 1000$, and privacy parameter $\epsilon = 0.1$. We observe that mixing with the uniform distribution shifts some of the weight from the most probable central element to those with lower probabilities, resulting in reduced skewness in the output distribution.

Recall from Definition 2.1 that the total variation distance is upper bounded by α. From Theorem 3.1, we have the accuracy of ROO

$$\alpha = \frac{1}{1 + \frac{n}{k}(e^{\epsilon} - 1)} \left(1 - \frac{1}{k}\right). \tag{12}$$

For fixed values of k and n, we obtain the accuracy of DS-ROO empirically, and compare it with that of (Raskhodnikova et al., 2021; Cheu & Nayak, 2025) and ROO. Fig. 3 shows the α vs. ϵ curves for all three algorithms. We observe that DS-ROO achieves dramatically better accuracy than the other methods while providing the same privacy guarantee. Of course, the numerical results in Fig. 3 are for the particular distribution shown in Fig. 2. We anticipate that we will see similar improvements for distributions that are not too skewed—if the distribution is more skewed (such as if the probability of a letter is 0), then there will be no improvement over vanilla ROO.

5. Conclusion

In this work, we build on and improve a differentially private sampling algorithm for discrete distributions on a finite alphabet. We establish a strictly lower sampling complexity bound than prior methods. In addition, we propose a generalization—Data-Specific Reveal-or-Obscure (DS-ROO)—which adaptively selects the probability of obscuring the empirical distribution of the dataset to improve the privacy-utility trade-off. For future work, we aim to explore theoretical bounds on the utility of DS-ROO, and the practicality of our proposed method for more complex distribution classes.

References

Acharya, J., Canonne, C. L., and Tyagi, H. Inference under information constraints i: Lower bounds from chi-square contraction. *IEEE Transactions on Information Theory*, 66(12):7835–7855, 2020a.

Acharya, J., Canonne, C. L., and Tyagi, H. Inference under information constraints ii: Communication constraints and shared randomness. *IEEE Transactions on Information Theory*, 66(12):7856–7877, 2020b.

Axelrod, B., Garg, S., Sharan, V., and Valiant, G. Sample amplification: Increasing dataset size even when learning is impossible. In *International Conference on Machine Learning*, pp. 442–451. PMLR, 2020.

Bellovin, S. M., Dutta, P. K., and Reitinger, N. Privacy and synthetic datasets. *Stan. Tech. L. Rev.*, 22:1, 2019.

Boedihardjo, M., Strohmer, T., and Vershynin, R. Private sampling: a noiseless approach for generating differentially private synthetic data. *SIAM Journal on Mathematics of Data Science*, 4(3):1082–1115, 2022.

Cheu, A. and Nayak, D. Differentially private multi-sampling from distributions. In *International Conference on Algorithmic Learning Theory*, pp. 289–314. PMLR, 2025.

Diakonikolas, I., Hardt, M., and Schmidt, L. Differentially private learning of structured discrete distributions. *Advances in Neural Information Processing Systems*, 28, 2015.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. 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*, pp. 265–284. Springer, 2006.

Dwork, C., Roth, A., et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

Ghazi, B., Hu, X., Kumar, R., and Manurangsi, P. On differentially private sampling from gaussian and product distributions. *Advances in Neural Information Processing Systems*, 36, 2024.

Hardt, M., Ligett, K., and McSherry, F. A simple and practical algorithm for differentially private data release. *Advances in neural information processing systems*, 25, 2012.

Husain, H., Balle, B., Cranko, Z., and Nock, R. Local differential privacy for sampling. In *International Conference on Artificial Intelligence and Statistics*, pp. 3404–3413. PMLR, 2020.

Kamath, G., Li, J., Singhal, V., and Ullman, J. Privately learning high-dimensional distributions. In *Conference on Learning Theory*, pp. 1853–1902. PMLR, 2019.

Majeed, A. and Lee, S. Anonymization techniques for privacy preserving data publishing: A comprehensive survey. *IEEE access*, 9:8512–8545, 2020.

Park, H.-Y., Asoodeh, S., and Lee, S.-H. Exactly minimax-optimal locally differentially private sampling. *Advances in Neural Information Processing Systems*, 37:10274–10319, 2024.

Raskhodnikova, S., Sivakumar, S., Smith, A., and Swanberg, M. Differentially private sampling from distributions. *Advances in Neural Information Processing Systems*, 34: 28983–28994, 2021.

Warner, S. L. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American statistical association*, 60(309):63–69, 1965.

Zamanlooy, B., Diaz, M., and Asoodeh, S. Locally private sampling with public data. *arXiv preprint arXiv:2411.08791*, 2024.

Zhu, T., Li, G., Zhou, W., and Philip, S. Y. Differentially private data publishing and analysis: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 29 (8):1619–1638, 2017.

A. Proof of Theorem 3.1

Privacy Analysis. Given x^n, Algorithm 1 chooses output $Y = y$ with probability

$$P(Y = y|x^n) = \frac{q}{k} + (1-q)\hat{P}_{x^n}(y), \tag{13}$$

where $\hat{P}_{x^n}(x)$ denotes the empirical probability of each $x \in [k]$,

$$\hat{P}_{x^n}(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}(x_i = x). \tag{14}$$

For neighboring datasets $x^n \sim \tilde{x}^n$, their corresponding empirical probabilities of one observation differ by at most $\frac{1}{n}$, i.e., for all $x \in \mathcal{X}$,

$$|\hat{P}_{x^n}(x) - \hat{P}_{\tilde{x}^n}(x)| \leq \frac{1}{n}. \tag{15}$$

By Definition 2.2, in order for ROO to satisfy ϵ-DP, the following condition must hold for all possible $x^n \sim \tilde{x}^n$,

$$\frac{P(Y = y|x^n)}{P(Y = y|\tilde{x}^n)} \leq e^{\epsilon}. \tag{16}$$

We can write the left side of (16) as

$$\frac{P(Y = y|x^n)}{P(Y = y|\tilde{x}^n)} = \frac{\frac{q}{k} + (1-q)\hat{P}_{x^n}(y)}{\frac{q}{k} + (1-q)\hat{P}_{\tilde{x}^n}(y)} \tag{17}$$

$$\leq \sup_p \frac{\frac{q}{k} + (1-q)\left(p + \frac{1}{n}\right)}{\frac{q}{k} + (1-q)p} \tag{18}$$

$$= \sup_p 1 + \frac{(1-q)\frac{1}{n}}{\frac{q}{k} + (1-q)p} \leq 1 + \frac{(1-q)\frac{1}{n}}{\frac{q}{k}}. \tag{19}$$

Here, in (17), we substitute (13). In (18), we use the property of the empirical probabilities stated in (15), denoting $\hat{P}_{\tilde{x}^n}(y)$ with p for notational simplicity. In (19), the supremum is obtained when $p = 0$. Therefore, ROO satisfies ϵ-DP guarantee if the right side of (19) is bounded by e^{ϵ}, i.e.,

$$1 + \frac{(1-q)\frac{1}{n}}{\frac{q}{k}} \leq e^{\epsilon}. \tag{20}$$

Rearranging (20), we obtain the privacy guarantee of Theorem 3.1 stated in (7).

Utility Analysis. For our proposed sampler ROO in Algorithm 1, the output distribution is

$$Q(y) = \sum_{x^n \in \mathcal{X}^n} \left(\frac{q}{k} + (1-q)\hat{P}_{x^n}(y)\right) \Pr\{X^n = x^n\} \tag{21}$$

$$= \frac{q}{k}\sum_{x^n \in \mathcal{X}^n} \Pr\{X^n = x^n\} + (1-q)\sum_{x^n \in \mathcal{X}^n} \hat{P}_{x^n}(y)\Pr\{X^n = x^n\} \tag{22}$$

$$= \frac{q}{k} + (1-q)\mathbb{E}_{X^n}[\hat{P}_{x^n}(y)] = \frac{q}{k} + (1-q)P(y), \tag{23}$$

where (23) follows from the fact that $\hat{P}_{x^n}$ is the empirical distribution of a dataset sampled from P. The total variation distance between the discrete distributions Q and P is thus

$$d_{TV}(Q, P) = \frac{1}{2}\sum_{y \in \mathcal{X}} |Q(y) - P(y)| \tag{24}$$

$$= \frac{q}{2}\sum_{y \in \mathcal{X}} \left|\frac{1}{k} - P(y)\right| \tag{25}$$

$$= q \times d_{TV}\left(\frac{1}{k}, P(y)\right) \tag{26}$$

$$\leq q \times \max_{P(y)} d_{TV}\left(\frac{1}{k}, P(y)\right), \tag{27}$$

where we obtain (25) by rearranging and substituting (23). Note that, for a convex objective function, the maximum is achieved at its corner points. Hence, the distribution P that maximizes the d_{TV} between the uniform distribution on $[k]$ and the input distribution P must be one of the corner points of the k-dimensional probability simplex $\mathcal{P}$, e.g. $P = \{1, 0, \ldots, 0\}$. The maximum d_{TV} is then computed as

$$\max_{P(y)} d_{TV}\left(\frac{1}{k}, P(y)\right) = \max_{P(y)} \frac{1}{2}\sum_{y \in \mathcal{X}} \left|\frac{1}{k} - P(y)\right| = 1 - \frac{1}{k}. \tag{28}$$

Substituting this maximum objective value into (27), we have

$$d_{TV}(Q, P) \leq q\left(1 - \frac{1}{k}\right). \tag{29}$$

Comparing this result with Definition 2.1, we obtain the utility of Theorem 3.1, stated in (8). We can then express q, i.e., the probability of sampling from the uniform distribution, in terms of α and k,

$$q = \frac{\alpha}{1 - \frac{1}{k}} = \frac{k\alpha}{k-1}. \tag{30}$$

Substituting the above into (20) and rearranging, we obtain sample complexity

$$n = \frac{k(1-\alpha) - 1}{\alpha(e^{\epsilon} - 1)}. \tag{31}$$

A.1. Proof of Lemma 3.2

We first compare the sampling complexity of ROO with that of (Raskhodnikova et al., 2021). From (9) in Theorem 3.1, we have the sampling complexity of ROO,

$$n = \frac{k(1-\alpha)-1}{\alpha(e^\epsilon - 1)} = \frac{k\left(1-\alpha-\frac{1}{k}\right)}{\alpha(e^\epsilon-1)} \tag{32}$$

$$= \frac{2k}{\alpha\epsilon}\frac{\epsilon}{2(e^\epsilon-1)}\left(1-\frac{1}{k}-\alpha\right) \tag{33}$$

$$< \frac{2k}{\alpha\epsilon} = n'. \tag{34}$$

As ϵ increases, the term $e^\epsilon - 1$ in the denominator of (9) grows exponentially, resulting in significantly lower n compared to n'. Secondly, we compare the sampling complexity of ROO with that of (Cheu & Nayak, 2025). Subtracting (6) from (9), we have

$$\frac{k(1-\alpha)-1}{\alpha(e^\epsilon-1)} - \frac{(k-1)(1-\alpha)}{\alpha\epsilon}$$

$$= \frac{k(1-\alpha)-1+\alpha-\alpha}{\alpha(e^\epsilon-1)} - \frac{(k-1)(1-\alpha)}{\alpha\epsilon} \tag{35}$$

$$= \frac{(k-1)(1-\alpha)-\alpha}{\alpha(e^\epsilon-1)} - \frac{(k-1)(1-\alpha)}{\alpha\epsilon} \tag{36}$$

$$= \frac{\epsilon(k-1)(1-\alpha)-\alpha\epsilon-(e^\epsilon-1)(k-1)(1-\alpha)}{\alpha\epsilon(e^\epsilon-1)} \tag{37}$$

$$= \frac{(\epsilon-e^\epsilon+1)(k-1)(1-\alpha)-\alpha\epsilon}{\alpha\epsilon(e^\epsilon-1)} < 0, \tag{38}$$

where (38) follows from the fact that $1+\epsilon < e^\epsilon$ for $\epsilon > 0$. Therefore, despite the similar structure of ROO and SubRR, ROO requires fewer samples to achieve ϵ-DP.

B. Proof of Theorem 4.1

Since m denotes the smallest number of times an element in the alphabet $[k]$ appears in a dataset x^n, it is at most $\lfloor \frac{n}{k} \rfloor$ when the empirical distribution is uniform. For neighboring datasets $x^n \sim \tilde{x}^n$, there are three possible values of $\tilde{m}$: (1) $\tilde{m} = m$, when the different entry in $\tilde{x}^n$ is a different element in $[k]$, hence it does not affect the minimum count, (2) $\tilde{m} = m+1$, and (3) $\tilde{m} = m-1$. In order to understand how the behavior of q_m changes with m, we first analyze the likelihood ratio of the output distributions. Recall from Definition 2.2 that, for Algorithm 2 to achieve ϵ-DP, the following condition must hold for all possible $x^n \sim \tilde{x}^n$ pairs,

$$\max_x \frac{\frac{q_{\tilde{m}}}{k} + (1-q_{\tilde{m}})\hat{P}_{\tilde{x}^n}(x)}{\frac{q_m}{k} + (1-q_m)\hat{P}_{x^n}(x)} \le e^\epsilon. \tag{39}$$

Adding and subtracting $\hat{P}_{x^n}(x)$ from $\hat{P}_{\tilde{x}^n}(x)$ in the numerator, the condition becomes

$$\max_x \frac{\frac{q_{\tilde{m}}}{k} + (1-q_{\tilde{m}})(\hat{P}_{\tilde{x}^n}(x) - \hat{P}_{x^n}(x) + \hat{P}_{x^n}(x))}{\frac{q_m}{k} + (1-q_m)\hat{P}_{x^n}(x)} \le e^\epsilon. \tag{40}$$

The ratio is maximized when the numerator is maximized. Using (15), we can rewrite the condition as

$$\frac{\frac{q_{\tilde{m}}}{k} + (1-q_{\tilde{m}})(\frac{1}{n} + \hat{P}_{x^n}(x))}{\frac{q_m}{k} + (1-q_m)\hat{P}_{x^n}(x)} \le e^\epsilon \tag{41}$$

$$\Rightarrow \hat{P}_{x^n}(x)\left(1-q_{\tilde{m}} - e^\epsilon(1-q_m)\right) \le e^\epsilon\frac{q_m}{k} - \frac{q_{\tilde{m}}}{k} - (1-q_{\tilde{m}})\frac{1}{n}. \tag{42}$$

If (42) is satisfied at the two endpoints of $\hat{P}_{x^n}$, then the condition holds for all $\hat{P}_{x^n}$. Now, substituting $\min_x \hat{P}_{x^n}(x) = \frac{m}{n}$ into (42), we require

$$\frac{m}{n}\left(1-q_{\tilde{m}} - e^\epsilon(1-q_m)\right) \le e^\epsilon\frac{q_m}{k} - \frac{q_{\tilde{m}}}{k} - (1-q_{\tilde{m}})\frac{1}{n}. \tag{43}$$

Considering the three cases of $\tilde{m}$ and simplifying (43), we have the following conditions:

$$(u_m - v_m)q_m \le w_m, \text{ for all } m, \tag{44}$$

$$u_m q_{m+1} \le v_m q_m + w_m, \text{ for } m = 0, 1, \ldots, \left\lfloor \frac{n}{k} \right\rfloor - 1, \tag{45}$$

$$u_m q_{m-1} \le v_m q_m + w_m, \text{ for } m = 1, 2, \ldots, \left\lfloor \frac{n}{k} \right\rfloor. \tag{46}$$

Similarly, substituting $\max_x \hat{P}_{x^n}(x) = 1$ into (42), we require

$$1-q_{\tilde{m}} - e^\epsilon(1-q_m) \le e^\epsilon\frac{q_m}{k} - \frac{q_{\tilde{m}}}{k} - (1-q_{\tilde{m}})\frac{1}{n}. \tag{47}$$

Considering the three cases of $\tilde{m}$ and simplifying (47), we have three more conditions:

$$(u' - v')q_m \le w', \text{ for all } m, \tag{48}$$

$$u' q_{m+1} \le v' q_m + w', \text{ for } m = 0, 1, \ldots, \left\lfloor \frac{n}{k} \right\rfloor - 1, \tag{49}$$

$$u' q_{m-1} \le v' q_m + w', \text{ for } m = 1, 2, \ldots, \left\lfloor \frac{n}{k} \right\rfloor. \tag{50}$$

Therefore, for Algorithm 2 to achieve ϵ-DP, the function q_m must satisfy (44)–(46) and (48)–(50). Now, let q_m have the following expression:

$$q_m = \max\left\{0, \frac{u_m}{v_m}q_{m-1} - \frac{w_m}{v_m}, \frac{v'}{u'}q_{m-1} + \frac{w'}{u'}\right\} \tag{51}$$

$$= \max\left\{0, f_1(q_{m-1}), f_2(q_{m-1})\right\}. \tag{52}$$

Here, the initial value q_0 is defined as in (10). Note that, since q_m is a probability, we must have $0 \leq q_m \leq 1$ for all m. The remainder of the proof is structured as follows:

- Lemma B.1 shows that $f_1(q_{m-1})$ is non-increasing.
- Lemma B.2 shows that $f_2(q_{m-1})$ is non-increasing.
- Lemma B.3 shows that
$$q_m = \max\{0, f_1(q_{m-1}), f_2(q_{m-1})\} \tag{53}$$
is monotonic and non-increasing.
- Lemma B.4 shows that (51) satisfies the condition in (44).
- Lemma B.5 shows that (51) satisfies the condition in (45).
- Lemma B.6 shows that (51) satisfies the condition in (48).
- Lemma B.7 shows that (51) satisfies the condition in (50).

Therefore, Algorithm 2 is ϵ-differentially private.

B.1. Additional Proofs

Lemma B.1. *For any $k \geq 2, \epsilon > 0$, and datasets of size n where $n > k$, the function $f_1(q_{m-1}) = \frac{u_m}{v_m} q_{m-1} - \frac{w_m}{v_m}$ is non-increasing in m.*

Proof. Since the function $f_1(q_{m-1}) : \left\{0, 1, \ldots, \left\lfloor \frac{n}{k} \right\rfloor\right\} \mapsto [0, 1]$ is discrete, we need to show that

$$f_1(q_{m-1}) \leq f_1(q_{m-2}), \tag{54}$$

for all $m \in \left\{0, 1, \ldots, \left\lfloor \frac{n}{k} \right\rfloor\right\}$. Note that, we obtain the recursive expression $q_m = f_1(q_{m-1})$ by considering (46) as an equality:

$$v_m q_m = u_m q_{m-1} - w_m \tag{55}$$
$$\Rightarrow v_m q_m - v_m q_{m-1} = u_m q_{m-1} - w_m - v_m q_{m-1} \tag{56}$$
$$\Rightarrow q_m - q_{m-1} = \frac{u_m - v_m}{v_m} q_{m-1} - \frac{w_m}{v_m} \tag{57}$$
$$\Rightarrow q_m - q_{m-1} = \left(\frac{u_m}{v_m} - 1\right) q_{m-1} - \frac{w_m}{v_m}. \tag{58}$$

It suffices to show that the right side of (58) is negative. Substituting the expressions for u_m and v_m into the first coefficient, we have

$$\frac{u_m}{v_m} - 1 = \frac{-\frac{m}{n} + \frac{1}{k} - \frac{1}{n}}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} - 1 \tag{59}$$
$$= \frac{\left(\frac{1}{k} - \frac{m}{n}\right)\frac{1}{n} - e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} \tag{60}$$
$$= \frac{-\frac{1}{n} - \left(\frac{1}{k} - \frac{m}{n}\right)(e^\epsilon - 1)}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} \tag{61}$$
$$< 0. \tag{62}$$

Similarly, substituting the expression for w_m into the second coefficient, we have

$$\frac{w_m}{v_m} = \frac{-\frac{1}{n} - \frac{m}{n} + \frac{m}{n}e^\epsilon}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} \tag{63}$$
$$= \frac{-\frac{1}{n} + \frac{m}{n}(e^\epsilon - 1)}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} \tag{64}$$
$$> 0, \tag{65}$$

for $m > \left\lfloor \frac{1}{e^\epsilon - 1} \right\rfloor$. Since q_m has a lower bound 0 in this range, the right side of (58) is

$$\left(\frac{u_m}{v_m} - 1\right) q_{m-1} - \frac{w_m}{v_m} < 0. \tag{66}$$

It still remains to show that $q_m - q_{m-1} < 0$ for $m = 1, 2, \ldots, \left\lfloor \frac{1}{e^\epsilon - 1} \right\rfloor$. We observe from Lemma B.4 (proved independently) that in this range of m, q_m has a non-zero, positive lower bound. Substituting this bound to the right side of (58), we have

$$\left(\frac{u_m}{v_m} - 1\right) \frac{w_{m-1}}{u_{m-1} - v_{m-1}} - \frac{w_m}{v_m}$$
$$= \frac{-\frac{1}{n} - \left(\frac{1}{k} - \frac{m}{n}\right)(e^\epsilon - 1)}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} \frac{\frac{m-1}{n}(e^\epsilon - 1) - \frac{1}{n}}{\left(\frac{m-1}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} - \frac{w_m}{v_m} \tag{67}$$
$$= \frac{-\frac{1}{n} - \left(\frac{1}{k} - \frac{m}{n}\right)(e^\epsilon - 1)}{\left(\frac{m-1}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \frac{\frac{m-1}{n}(e^\epsilon - 1) - \frac{1}{n}}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} - \frac{w_m}{v_m} \tag{68}$$
$$= C\frac{-\frac{1}{n} + \frac{m}{n}(e^\epsilon - 1) - \frac{1}{n}(e^\epsilon - 1)}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} - \frac{w_m}{v_m} \tag{69}$$
$$= C\left[\frac{-\frac{1}{n} + \frac{m}{n}(e^\epsilon - 1)}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)} - \frac{\frac{1}{n}(e^\epsilon - 1)}{e^\epsilon\left(\frac{1}{k} - \frac{m}{n}\right)}\right] - \frac{w_m}{v_m} \tag{70}$$
$$= C\left[\frac{w_m}{v_m} - \frac{e^\epsilon - 1}{e^\epsilon\left(\frac{n}{k} - m\right)}\right] - \frac{w_m}{v_m}. \tag{71}$$

Here, $\frac{w_m}{v_m} < 0$ for the relevant range of m. Moreover, since $\frac{n}{k} > m$, $\frac{e^\epsilon - 1}{e^\epsilon\left(\frac{n}{k} - m\right)}$ is positive and less than 1. Finally, the

coefficient,

$$C = \frac{-\frac{1}{n} - \left(\frac{1}{k} - \frac{m}{n}\right)(e^\epsilon - 1)}{\left(\frac{m-1}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \tag{72}$$

$$= \frac{-\frac{1}{n} - \left(\frac{1}{k} - \frac{m-1+1}{n}\right)(e^\epsilon - 1)}{\left(\frac{m-1}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \tag{73}$$

$$= \frac{\left(\frac{m-1}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n} + \frac{1}{n}(e^\epsilon - 1)}{\left(\frac{m-1}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \tag{74}$$

$$= 1 + \frac{\frac{1}{n}(e^\epsilon - 1)}{\left(\frac{m-1}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \tag{75}$$

$$= 1 - \frac{e^\epsilon - 1}{1 + n(e^\epsilon - 1)\left(\frac{1}{k} - \frac{m-1}{n}\right)} \tag{76}$$

$$< 1. \tag{77}$$

Therefore, we have

$$C\left[\frac{w_m}{v_m} - \frac{e^\epsilon - 1}{e^\epsilon\left(\frac{n}{k} - m\right)}\right] < \frac{w_m}{v_m} \tag{78}$$

$$\Rightarrow C\left[\frac{w_m}{v_m} - \frac{e^\epsilon - 1}{e^\epsilon\left(\frac{n}{k} - m\right)}\right] - \frac{w_m}{v_m} < 0 \tag{79}$$

Thus, for all m,

$$q_m \leq q_{m-1}, \tag{80}$$

i.e., $f_1(q_{m-1}) \leq f_1(q_{m-2})$. Therefore, $f_1(q_{m-1})$ is non-increasing. □

Lemma B.2. *For any $k \geq 2, \epsilon > 0$, and datasets of size n where $n > k$, the function $f_2(q_{m-1}) = \frac{v'}{u'}q_{m-1} + \frac{w'}{u'}$ is non-increasing in m.*

Proof. We observe that for some n, k and ϵ, the function

$$f_2(q_{m-1}) = \frac{v'}{u'}q_{m-1} + \frac{w'}{u'} \tag{81}$$

is linear in q_m, with slope $\frac{v'}{u'}$ and intercept $\frac{w'}{u'}$. We can show that the $f_2(q_{m-1})$ is non-increasing by using the fixed-point iteration method. The slope

$$\frac{v'}{u'} = \frac{e^\epsilon\left(\frac{1}{k} - 1\right)}{-1 + \frac{1}{k} - \frac{1}{n}} = \frac{e^\epsilon\left(\frac{1}{k} - 1\right)}{\left(\frac{1}{k} - 1\right) - \frac{1}{n}} \tag{82}$$

is positive and less than 1 for $0 < \epsilon < \log(1 + \frac{1}{n})$. For $\epsilon > \log(1 + \frac{1}{n})$, the slope is positive and greater than 1. The intercept

$$\frac{w'}{u'} = \frac{e^\epsilon - 1 - \frac{1}{n}}{-1 + \frac{1}{k} - \frac{1}{n}} \tag{83}$$

is positive for $\epsilon < \log(1 + \frac{1}{n})$ and negative for $\epsilon > \log(1 + \frac{1}{n})$. Thus, we can consider two regions of ϵ values and apply fixed-point iteration to each case.

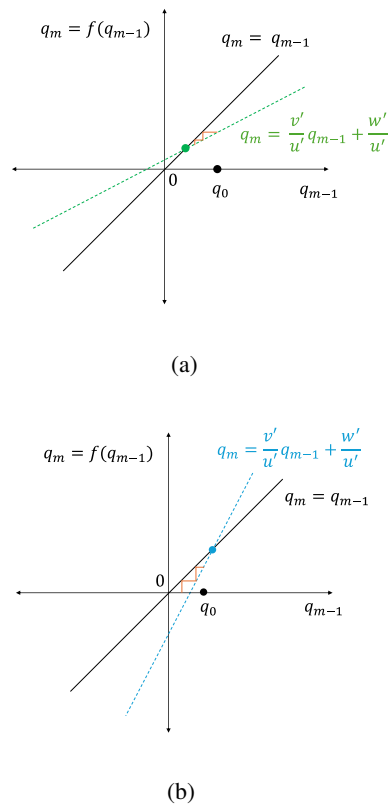

Figure 4. Fixed-point iteration.

Case I: $0 < \epsilon < \log(1 + \frac{1}{n})$. In this region, we have $0 < \frac{v'}{u'} < 1$ and a positive intercept $\frac{w'}{u'}$. As illustrated in Figure 4(a), the initial value q_0 is greater than the intersection point of $q_m = \frac{v'}{u'}q_{m-1} + \frac{w'}{u'}$ and $q_m = q_{m-1}$. In this case, the fixed-point iteration is contractive, and as a result, the sequence q_m converges monotonically to the unique fixed point of $f(q_{m-1})$.

As the slope approaches 1, the fixed point still remains smaller than q_0. When crossover happens and the slope becomes greater than 1, the fixed point moves to a value larger than q_0, preserving the decreasing nature of q_m.

Case II: $\epsilon > \log(1 + \frac{1}{n})$. In this region, we have $\frac{v'}{u'} > 1$ and a negative intercept $\frac{w'}{u'}$. As illustrated in Figure 4(b), the initial value q_0 is less than the intersection point of $q_m = \frac{v'}{u'}q_{m-1} + \frac{w'}{u'}$ and $q_m = q_{m-1}$. In this case, the sequence q_m decreases monotonically until it reaches 0. □

Lemma B.3. $q_m = \max\{0, f_1(q_{m-1}), f_2(q_{m-1})\}$ *is non-increasing in m.*

Proof. We need to show that

$$\max\{0, f_1(q_m), f_2(q_m)\} \leq \max\{0, f_1(q_{m-1}), f_2(q_{m-1})\}. \tag{84}$$

By Lemmas B.1 and B.2, we have

$$f_1(q_m) \leq f_1(q_{m-1}) \tag{85}$$
$$f_2(q_m) \leq f_2(q_{m-1}). \tag{86}$$

Since $f_1(q_m)$ and $f_2(q_m)$ are both non-increasing in m, they can intersect at most once in the domain of m. The monotonicity of both functions is preserved at the intersection m^*, i.e.,

$$f_1(q_{m^*+1}) \leq f_1(q_{m^*}) = f_2(q_{m^*}) \leq f_2(q_{m^*-1}) \tag{87}$$

Thus, the pointwise maximum of $0, f_1(q_m)$ and $f_2(q_m)$ preserves the non-increasing property of the individual functions. Therefore, the final expression of q_m is non-increasing. □

Lemma B.4. *For $m \in \left\{0, 1, 2, \ldots, \left\lfloor \frac{1}{e^\epsilon - 1} \right\rfloor \right\}$, and u_m, v_m, the function q_m satisfies the condition*

$$q_m \geq \frac{w_m}{u_m - v_m}. \tag{88}$$

Proof. Recall the inequality condition in (44),

$$(u_m - v_m) q_m \leq w_m, \tag{89}$$

for all m. Note that, $u_m - v_m$ is negative for all m. Rearranging (89), we need to show that q_m satisfies

$$q_m \geq \frac{w_m}{u_m - v_m}. \tag{90}$$

Now, since $0 \leq q_m \leq 1$, the inequality in (90) is automatically satisfied whenever the right side is negative. Therefore, the only case we need to consider is when the right side is positive. Since the denominator $u_m - v_m$ is always negative, this happens only when the numerator w_m is also negative, i.e.,

$$w_m = -\frac{1}{n} - \frac{m}{n} + \frac{m}{n} e^\epsilon < 0 \Rightarrow m < \frac{1}{e^\epsilon - 1}. \tag{91}$$

Let,

$$t_m = \frac{w_m}{u_m - v_m} = \frac{\frac{w_m}{v_m}}{\frac{u_m}{v_m} - 1} \tag{92}$$
$$\Rightarrow \left(\frac{u_m}{v_m} - 1\right) t_m = \frac{w_m}{v_m}. \tag{93}$$

We observe that

$$t_0 = \frac{w_0}{u_0 - v_0} = \frac{-\frac{1}{n}}{\frac{1}{k} - \frac{1}{n} - \frac{e^\epsilon}{k}} = \frac{1}{1 + \frac{n}{k}(e^\epsilon - 1)} = q_0. \tag{94}$$

Moreover,

$$t_m = \frac{\frac{m}{n}(e^\epsilon - 1) - \frac{1}{n}}{\left(\frac{m}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \tag{95}$$
$$= \frac{\frac{m}{n}(e^\epsilon - 1) - \frac{1}{n} - \frac{1}{k}(e^\epsilon - 1) + \frac{1}{k}(e^\epsilon - 1)}{\left(\frac{m}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \tag{96}$$
$$= \frac{\left(\frac{m}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n} + \frac{1}{k}(e^\epsilon - 1)}{\left(\frac{m}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \tag{97}$$
$$= 1 + \frac{\frac{1}{k}(e^\epsilon - 1)}{\left(\frac{m}{n} - \frac{1}{k}\right)(e^\epsilon - 1) - \frac{1}{n}} \tag{98}$$
$$= 1 - \frac{\frac{1}{k}(e^\epsilon - 1)}{\left(\frac{1}{k} - \frac{m}{n}\right)(e^\epsilon - 1) + \frac{1}{n}}. \tag{99}$$

As m increases, (99) becomes smaller due to a larger fraction being subtracted from 1. Thus, t_m is decreasing in m. Now, we prove the lemma for the base case and apply induction. *Base case:* For $m = 1$, we have

$$q_1 = \frac{u_1}{v_1} q_0 - \frac{w_1}{v_1} \tag{100}$$
$$= \frac{u_1}{v_1} t_0 - \left(\frac{u_1}{v_1} - 1\right) t_1 \tag{101}$$
$$= \frac{u_1}{v_1}(t_0 - t_1) + t_1 \tag{102}$$
$$> t_1, \tag{103}$$

since $\frac{u_1}{v_1} > 0$, and t_m is strictly decreasing in m.

Induction step: Let us assume $q_m > t_m$ for any m. We have

$$q_{m+1} = \frac{u_{m+1}}{v_{m+1}} q_m - \frac{w_{m+1}}{v_{m+1}} \tag{104}$$
$$= \frac{u_{m+1}}{v_{m+1}} q_m - \left(\frac{u_{m+1}}{v_{m+1}}\right) t_{m+1} \tag{105}$$
$$> \frac{u_{m+1}}{v_{m+1}} t_m - \left(\frac{u_{m+1}}{v_{m+1}}\right) t_{m+1} \tag{106}$$

Rearranging the right side of (106), we have

$$q_{m+1} > \frac{u_{m+1}}{v_{m+1}}(t_m - t_{m+1}) + t_{m+1}. \tag{107}$$

Since t_m is strictly decreasing in m and the coefficient of $t_m - t_{m+1}$ is positive, $q_{m+1} > t_{m+1}$. Thus, for $m \in \left\{0, 1, 2, \ldots, \left\lfloor \frac{1}{e^\epsilon - 1} \right\rfloor \right\}$, q_m satisfies (88). □

Lemma B.5. *For $m \in \left\{0, 1, \ldots, \left\lfloor \frac{n}{k} \right\rfloor - 1\right\}$, the function q_m satisfies the inequality condition*

$$u_m q_{m+1} < v_m q_m + w_m. \tag{108}$$

Proof. Using the expressions for u_m, v_m, and w_m as defined in Algorithm 2, we have, for $m = 0$,

$$u_0 = \frac{1}{k} - \frac{1}{n}, v_0 = \frac{e^\epsilon}{k}, w_0 = -\frac{1}{n}. \tag{109}$$

Then, we can write

$$
\begin{aligned}
u_0q_1 - v_0q_0 &= u_0q_1 - u_0q_0 + u_0q_0 - v_0q_0 \\
&= u_0(q_1 - q_0) + \left(\frac{1}{k} - \frac{1}{n}\right) q_0 - \frac{e^\epsilon}{k} q_0 && (110)\\
&= u_0(q_1 - q_0) + \left(\frac{1}{k} + w_0\right) q_0 - \frac{e^\epsilon}{k} q_0 && (111)\\
&= u_0(q_1 - q_0) + \frac{1}{k} q_0 + w_0q_0 - \frac{e^\epsilon}{k} q_0 && (112)\\
&= w_0q_0 - \frac{e^\epsilon - 1}{k} q_0 - u_0(q_0 - q_1) && (113)\\
&< w_0. && (114)
\end{aligned}
$$

Here, (114) follows from the fact that $0 \le q_0 \le 1$ and q_m is non-increasing.

For $m = 1, 2, \ldots, \lfloor \frac{n}{k} \rfloor - 1$, we can write

$$
\begin{aligned}
u_mq_{m+1} - v_mq_m &= u_mq_{m+1} - u_mq_{m-1} + w_m && (115)\\
&= w_m + u_m(q_{m+1} - q_{m-1}) && (116)\\
&= w_m + \left(-\frac{m}{n} + \frac{1}{k} - \frac{1}{n}\right)(q_{m+1} - q_{m-1}) && (117)\\
&= w_m - \left(\frac{1}{k} - \frac{m+1}{n}\right)(q_{m-1} - q_{m+1}) && (118)\\
&< w_m. && (119)
\end{aligned}
$$

Here, (119) follows from the fact that q_m is non-increasing and $\frac{1}{k} - \frac{m+1}{n}$ is non-negative for $m = 1, 2, \ldots, \lfloor \frac{n}{k} \rfloor - 1$. Finally, for $q_{m+1} = 0$, we have

$$
\begin{aligned}
-v_mq_m &= w_m - \left(\frac{1}{k} - \frac{m+1}{n}\right) q_{m-1} && (120)\\
&< w_m && (121)\\
\Rightarrow q_m &> -\frac{w_m}{v_m}. && (122)
\end{aligned}
$$

It follows from (65) that the right side of (122) is negative. Thus, (122) is a valid inequality, and we can conclude that q_m satisfies (108) for the relevant range of m. □

Lemma B.6. *For $m = 0, 1, 2, \ldots, \lfloor \frac{n}{k} \rfloor$, and u', v', and w' as defined in Algorithm 2, the function q_m satisfies the inequality condition*

$$(u' - v')q_m < w'. \tag{123}$$

Proof.

$$
\begin{aligned}
(u' - v')q_m &= \left(-1 + \frac{1}{k} - \frac{1}{n} - \frac{e^\epsilon}{k} + e^\epsilon\right) q_m && (124)\\
&= \left(e^\epsilon - 1 - \frac{1}{n} + \frac{1}{k}(1 - e^\epsilon)\right) q_m && (125)\\
&= w'q_m + \frac{q_m}{k}(1 - e^\epsilon) && (126)\\
&< w'. && (127)
\end{aligned}
$$

Here, (139) follows from the fact that $0 \le q_m \le 1$ and $\frac{q_m}{k}(1 - e^\epsilon) < 0$. □

Lemma B.7. *For $m = 1, 2, \ldots, \lfloor \frac{n}{k} \rfloor$, and u', v', and w' as defined in Algorithm 2, the function q_m satisfies the inequality condition*

$$u'q_{m-1} < v'q_m + w'. \tag{128}$$

Proof.

$$
\begin{aligned}
u'q_{m-1} - v'q_m &= u'q_{m-1} - v'q_{m-1} + v'q_{m-1} - v'q_m && (129)\\
&= (u' - v')q_{m-1} + v'(q_{m-1} - q_m) && (130)\\
&= (u' - v')q_{m-1} + e^\epsilon\left(\frac{1}{k} - 1\right)(q_{m-1} - q_m) && (131)\\
&< (u' - v')q_{m-1} && (132)\\
&< w'. && (133)
\end{aligned}
$$

Here, (144) follows from the fact that $e^\epsilon\left(\frac{1}{k} - 1\right) < 0$ and q_m is non-increasing, and (145) follows from Lemma B.6. Therefore,

$$u'q_{m-1} < v'q_m + w'. \tag{134}$$

□