Nearly-Optimal Private Selection via Gaussian Mechanism

Ethan Leeman
Google Research, New York, NY, USA

Pasin Manurangsi
Google Research, Bangkok, Thailand

Abstract

Steinke [9] recently asked the following intriguing open question: Can we solve the differentially private selection problem with nearly-optimal error by only (adaptively) invoking Gaussian mechanism on low-sensitivity queries? We resolve this question positively. In particular, for a candidate set $\mathcal{Y}$, we achieve error guarantee of $\tilde{O}(\log|\mathcal{Y}|)$, which is within a factor of $(\log\log|\mathcal{Y}|)^{O(1)}$ of the exponential mechanism [8]. This improves on Steinke's mechanism which achieves an error of $O(\log^{3/2}|\mathcal{Y}|)$.

2012 ACM Subject Classification Security and privacy; Theory of computation → Design and analysis of algorithms

Keywords and phrases Differentially Private Selection, Gaussian Mechanism

Digital Object Identifier 10.4230/LIPIcs.FORC.2026.4

Acknowledgements We thank Charlie Harrison for introducing us to this problem and for subsequent insightful discussions. We also thank Thomas Steinke for helpful discussions.

1 Introduction

Differential privacy (DP) [5] has become one of the most widely used notion of privacy, partly due to its mathematically rigorous protection for the user's data. In the original work that proposed DP [5], Dwork et al. also provided a mechanism that satisfies DP: If the desired output has low sensitivity, then we can simply add an appropriately-calibrated Laplace noise. Later on, other noise distribution – such as the Gaussian distribution [4] – has also been studied. These so-called Laplace and Gaussian mechanisms remain a staple of DP, both in theory and practice.

Meanwhile, another widely-used mechanism is the *Exponential Mechanism* [8], which can be easily described through the *Selection* problem, defined below.

▶ **Definition 1** (Selection). *In the Selection problem, there is a candidate set $\mathcal{Y}$, and for each candidate $y \in \mathcal{Y}$ there is a sensitivity-1 loss function $\ell_y : \mathcal{X}^* \to \mathbb{R}$. Given an input $X \in \mathcal{X}^*$, the goal of a selection algorithm is to output y_{out} that approximately minimizes the loss $\ell_y(X)$. We say that an algorithm $M : \mathcal{X}^* \to \mathcal{Y}$ solves Selection with expected error ν if*

$$\mathbb{E}_{y_{\text{out}}\sim M(X)}[\ell_{y_{\text{out}}}(X) - \min_{y\in\mathcal{Y}} \ell_y(X)] \le \nu.$$

Similarly, we say that an algorithm solves Selection with error ν with probability $1-\beta$ iff

$$\Pr_{y_{\text{out}}\sim M(X)}[\ell_{y_{\text{out}}}(X) - \min_{y\in\mathcal{Y}} \ell_y(X) \le \nu] \ge 1-\beta.$$

When there is no ambiguity, we simply write ℓ_y instead of $\ell_y(X)$ for brevity.

The ε-DP Exponential Mechanism solves Selection with expected error $O(\log|\mathcal{Y}|/\varepsilon)$. Since Selection can be used to model a variety of tasks, such as convex optimization [1], combinatorial optimization [7] and PAC learning [2]. Exponential mechanism immediately yields DP algorithms for these tasks, often with nearly tight error guarantees.

Given the importance of both types of mechanism, it is natural to ask whether we can implement the Exponential Mechanism using only noise addition mechanisms. Steinke [9] recently formalized this problem using the Gaussian mechanism. To further elaborate on this, it is best to focus on the notion of zero-concentrated differential privacy (zCDP) [6, 3]. Any ε-DP satisfies $\frac{\varepsilon^2}{2}$-zCDP; thus, the aforementioned Exponential Mechanism is an ρ-zCDP mechanism which solves Selection with expected error $O(\log|\mathcal{Y}|/\sqrt{\rho})$. Meanwhile, adding Gaussian noise $\mathcal{N}\left(0, \frac{1}{2\rho}\right)$ to any sensitivity-1 query satisfies ρ-zCDP. Furthermore, the composition theorem for zCDP is to simply add up to ρ values[1]. As a result, one may formalize the model as follows[2]:

▶ **Definition 2** (Gaussian Mechanism with Budget [9]). *In the* Gaussian mechanism with budget *model, the algorithm is given a finite budget $\rho > 0$ and can select the number of rounds M. At the i-th iteration, the algorithm can issue a sensitivity-1 query $q_i : \mathcal{X}^* \to \mathbb{R}$ and the budget ρ_i, and it will receive back $q_i(X) + Z_i$ where $Z_i \sim \mathcal{N}\left(0, \frac{1}{2\rho_i}\right)$. Here we enforce that $\sum_{i=1}^{M} \rho_i \le \rho$.*

[9] asked whether one can solve Selection with expected error $O(\log|\mathcal{Y}|/\sqrt{\rho})$ (similar to the Exponential Mechanism) in this model. Note that the trivial baseline – which queries $\ell_y(X)$ for all $y \in \mathcal{Y}$ using equal budget – results in an error of $\tilde{O}\left(\sqrt{|\mathcal{Y}|}/\sqrt{\rho}\right)$, significantly higher than that of the Exponential Mechanism. Perhaps surprisingly, Steinke [9] gave a simple algorithm that significantly improves upon this and achieves an expected error of $O\left(\frac{\log^{3/2}|\mathcal{Y}|}{\sqrt{\rho}}\right)$. (See Section 2.1 for more detail.) Nevertheless, there is still a gap of $O(\sqrt{\log|\mathcal{Y}|})$ between this bound and the error of the Exponential Mechanism. Thus, Steinke [9] asks whether this gap can be closed.

1.1 Our Contribution

The main contribution of our work is to answer this positively, up to a lower-order term. In particular, we give an algorithm with error $O\left(\frac{\log|\mathcal{Y}|\cdot(\log\log|\mathcal{Y}|)^{O(1)}}{\sqrt{\rho}}\right)$, as stated below.

▶ **Theorem 3** (Informal; See Theorem 5). *There is an algorithm in the Gaussian mechanism with budget model that solves Selection with expected error $O\left(\frac{\log|\mathcal{Y}|\cdot(\log\log|\mathcal{Y}|)^{11}}{\sqrt{\rho}}\right)$.*

At a high-level, our algorithm exploits the following observation: When there is a large *gap* between the smallest and second-smallest losses, the problem becomes significantly easier. In fact, when the gap is $\tilde{O}\left(\sqrt{\log|\mathcal{Y}|}/\sqrt{\rho}\right)$, Steinke's algorithm can already find the optimal index with high probability. Given this, our algorithm then proceeds in three steps. First, use subsampling to produce multiple instances (with different sampling rates). Secondly, recursively solve Selection on these instances but with the loss set in such a way that we prefer an instance with a large gap. Finally, we run Steinke's algorithm on the instance with large gap. The main challenge in the analysis is to ensure that, by carefully selecting the different sampling rates, at least one of the subsampled instances has both a large gap and a small optimal loss.

[1] In particular, this works even in the fully adaptive composition (aka privacy filter) setting [10].
[2] Strictly speaking, Steinke [9] proposed a model where the queries have equal budget. However, it is simple to see that the two models are equivalent; see Appendix A.

1.2 Discussion and Open Questions

An interesting open question here is to remove the lower order term (of $(\log\log|\mathcal{Y}|)^{O(1)}$) and completely achieve $O\left(\frac{1}{\sqrt{\rho}}\cdot\log|\mathcal{Y}|\right)$ error, which would exactly match the guarantee from the Exponential Mechanism. We remark that below we did not try to optimize the exponent of 11 in our $(\log\log|\mathcal{Y}|)^{O(1)}$ term, and indeed we suspect that it might be possible to make exponent arbitrarily close to two by slightly modifying the algorithm and the proof. Nevertheless, we do not see how the current technique can get rid of such a term completely.

Another interesting direction is to consider the model with *Laplace* mechanism, instead of Gaussian mechanism. Steinke's mechanism [9] gives an error of $O_\varepsilon(\log^2|\mathcal{Y}|)$. Unfortunately, our technique does not seem to achieve a tight bound in this setting since Laplace noise is not as concentrated as Gaussian; thus, one needs a much larger gap when working with Laplace noise[3].

Lastly, the algorithm we describe can only be implemented to solve Selection when the candidate set $|\mathcal{Y}|$ is finite. Meanwhile, the exponential mechanism can be well-defined on probability spaces with infinite sample spaces. It would be interesting to discover ways to use low-sensitivity queries to implement Selection in this setting. We believe it would require fundamentally different uses of the low-sensitivity queries, as opposed to the tree-based method used here and in [9].

2 Preliminaries

Throughout, we use log to denote base-2 logarithm.

The sensitivity of a function $g:\mathcal{X}^*\to\mathbb{R}$ is $\Delta(g):=\max_{\mathbf{X},\mathbf{X}'}|g(\mathbf{X})-g(\mathbf{X}')|$ where the maximum is over all neighboring inputs $\mathbf{X},\mathbf{X}'$. All results stated below work for any neighboring notion.

2.1 The Binary Tree Algorithm

We recall the "binary tree" algorithm proposed in [9]. Roughly speaking, the algorithm constructs a balanced full binary tree with the candidates being the leaves. The algorithm then starts at the root and, at each step, uses a single low-sensitivity query to determine whether the minimum value of the left subtree or the right subtree is larger; it then traverses accordingly. The algorithm is described more formally in Algorithm 1. The guarantee shown in [9] is as follows:

▶ **Lemma 4** ([9]). *For any $\rho>0$, Algorithm 1 solves Selection with expected error $O\left(\frac{\log^{3/2}|\mathcal{Y}|}{\sqrt{\rho}}\right)$.*

3 Nearly Optimal Algorithm for Selection

The main result of our work is an improvement over Steinke's algorithm (Lemma 4). Namely, we devise an algorithm with expected error $\tilde{O}\left(\frac{1}{\sqrt{\rho}}\cdot\log|\mathcal{Y}|\right)$, as stated formally below.

[3] Specifically, the tail bound for Laplace only gives $\tau=\tilde{\Theta}(K)$ instead of $\tilde{\Theta}(\sqrt{K})$ in Lemma 8. To satisfy Lemmas 10 and 11, we then need $\xi\geq\tilde{\Omega}(\sqrt{K}),\gamma\geq\tilde{\Omega}(K^{1.5})$. In other words, the best bound one could hope for is $\tilde{O}\left(\log^{1.5}|\mathcal{Y}|\right)$.

Algorithm 1 BinTree$((\ell_y)_{y\in\mathcal{Y}};\rho)$ [9].

Input: Loss functions $(\ell_y)_{y\in\mathcal{Y}}$, **Parameter:** Privacy budget ρ
Output: Index $y_{\text{out}} \in \mathcal{Y}$

$\mathcal{C} \leftarrow \mathcal{Y}$ ▷ $\mathcal{C}$ is the current candidate set
$K \leftarrow \lceil \log |\mathcal{Y}| \rceil$
while $|\mathcal{C}| > 1$ **do**
  Partition $\mathcal{C}$ into $\mathcal{C}_1 \cup \mathcal{C}_2$ where the sizes of $\mathcal{C}_1, \mathcal{C}_2$ differ by at most one.
  Query $q := \frac{1}{2}\left(\min_{y_1\in\mathcal{C}_1} \ell_{y_1}(X) - \min_{y_2\in\mathcal{C}_2} \ell_{y_2}(X)\right)$ with budget $\frac{\rho}{K}$; let the answer be $\tilde{q}$
  if $\tilde{q} > 0$ **then**
    $\mathcal{C} \leftarrow \mathcal{C}_2$
  else
    $\mathcal{C} \leftarrow \mathcal{C}_1$
return the only index in $\mathcal{C}$

▶ **Theorem 5** (Main Theorem). *For any $\rho > 0$, Algorithm 3 solves Selection with expected error $O\left(\frac{1}{\sqrt{\rho}} \cdot \log|\mathcal{Y}| \cdot (\log\log|\mathcal{Y}|)^{11}\right)$.*

The remainder of the section is devoted to the proof of Theorem 5.

3.1 Binary Tree, Revisited: Gap Instances Are Easy

We start by observing that, in many cases, the binary tree algorithm is already doing much better than Lemma 4 is suggesting. In particular, when there is a "gap" between the smallest and second smallest loss values, the algorithm already does well. To state this, let us first formalize the "gap".

▶ **Definition 6** (Minimum Index). *We say that y^* is a* minimum index *of $(\ell_y)_{y\in\mathcal{Y}}$ if $\ell_{y^*}(X) = \min_{y\in\mathcal{Y}} \ell_y(X)$.*

▶ **Definition 7** (Gap). *For any non-empty set $\mathcal{Y}$, the* gap *of an instance $(\ell_y)_{y\in\mathcal{Y}}$ is defined as*

$$\text{gap}\left((\ell_y)_{y\in\mathcal{Y}}\right) := \min_{y\in\mathcal{Y}\setminus\{y^*\}} \ell_y(X) - \ell_{y^*}(X),$$

where y^* is any minimum index of $\mathcal{Y}$. When $|\mathcal{Y}| = 1$, we define the gap to be ∞.

Our observation here is that if an instance has gap at least $\tilde{\Theta}\left(\sqrt{\log|\mathcal{Y}|}/\sqrt{\rho}\right)$, then we output the minimum index w.h.p., as stated more formally below.

▶ **Lemma 8** (Binary Tree for Gap Instances). *For any $\rho > 0, \beta \in (0, 0.5)$, let $\tau(|\mathcal{Y}|, \rho, \beta) := \frac{2\sqrt{\lceil\log|\mathcal{Y}|\rceil \cdot \left(\log\left(\frac{\lceil\log|\mathcal{Y}|\rceil}{\beta}\right)\right)}}{\sqrt{\rho}}$. If an instance $(\ell_y)_{y\in\mathcal{Y}}$ has gap at least $\tau(|\mathcal{Y}|, \rho, \beta)$, then Algorithm 1 outputs the minimum index y^* with probability at least $1 - \beta$.*

Proof. By standard concentration of Gaussian and the union bound, we have that $|\tilde{q} - q| < \frac{1}{\sqrt{(\rho/K)}} \cdot \sqrt{\log(K/\beta)} = \frac{\tau(|\mathcal{Y}|,\rho,\beta)}{2}$ for all rounds with probability at least $1 - \beta$. When this holds, since the instance has gap at least $\tau(|\mathcal{Y}|, \rho, \beta)$, we will always pick $\mathcal{C}_i$ that contains the minimum index y^*; that is, we output y^* in the end. ◀

3.2 Our New Recursive Algorithm and Its Guarantee

In this subsection, we present our main contribution, which is an algorithm that with high probability achieves an error bound of $\tilde{O}\left(\frac{1}{\sqrt{\rho}} \cdot \log |\mathcal{Y}|\right)$. The exact bound is stated below.

▶ **Theorem 9** (Main Theorem). *For any $\rho > 0, \beta \in (0, 0.001)$, Algorithm 2 solves Selection with error $O\left(\frac{1}{\sqrt{\rho}} \cdot \log |\mathcal{Y}| \cdot (\log\log |\mathcal{Y}|)^{10} \cdot \log\left(\frac{\log |\mathcal{Y}|}{\beta}\right)\right)$ with probability at least $1 - \beta$.*

Before we present the algorithm, let us describe the intuition behind the algorithm. The high-level description of the algorithm is very simple: (i) subsampling "many" subsets and (ii) recursively call the algorithm to select the subset with (approximately) largest gap, and then (iii) call BinTree (using the guarantee in Lemma 8). The main argument is regarding how should we sample and how "many" subsets do we need to get a gap of $\tilde{O}\left(\sqrt{\log |\mathcal{Y}|}\right)$ as required in Lemma 8. For convenience of discussion, suppose that $|\mathcal{Y}| = 2^K$ where $K \in \mathbb{N}$ is a perfect square. Intuitively, a "worst" case here is the following: For all $i \in [K]$, there are 2^i candidates whose losses are within $\tilde{O}(i)$ of the optimal. The reason here is that, if the losses are "more packed", then all candidates would result in an error less than $\tilde{O}(K)$, which would satisfy our desired bound. On the other hand, if the losses are "less packed", then the gaps are larger and thus it is easier for our steps (i) and (ii). Now, coming back to this instance, if we sample $2^{K-\sqrt{K}}$ candidates from $\mathcal{Y}$, then with probability $\Theta(2^{-\sqrt{K}})$, we will select the optimal candidate and none of the other $2^{\sqrt{K}}$ candidates that are within $\tilde{O}\left(\sqrt{K}\right)$-close to it. Such a set yields $\tilde{O}\left(\sqrt{K}\right)$ gap required for Lemma 8. Moreover, since the probability of sampling such a subset is $\Theta(2^{-\sqrt{K}})$, it suffices for us to sample $O(2^{\sqrt{K}})$ such sets. This means that, in the recursive step (ii), the "error" in the selection is $\tilde{O}\left(\sqrt{K}\right)$ which can be absorbed by the gap if we select the parameters carefully.

While the above presents the overview for the "worst" instance, in general the instance can be a lot more complicated; e.g. for some $i \in [K]$, there might be more or fewer than 2^i candidates whose losses are within $\tilde{O}(i)$. For example, if there are many more candidates (say $2^{\omega(\sqrt{K})}$) whose loss are very close to the optimum, then sampling as above will not be sufficient. Thus, in the main algorithm, we also randomly sample the sampling probability (i.e. subset size) by taking them to be a random power of two. We can show that, at least one such power of two results in a large gap with probability at least $2^{-\Omega(\sqrt{K})}$ (Lemma 12). We also need to incorporate the minimum loss value in the subset into the loss function in the recursive step (ii) (see Algorithm 2 in Algorithm 2); this is to prevent us from selecting a subset with a large gap but with a large optimal loss value. Our entire algorithm is given in Algorithm 2.

While not the focus of our work, we remark that the running time of Algorithm 2 is dominated by the loop (Lines 7 to 10), which takes $O(|\mathcal{Y}| \cdot T) = |\mathcal{Y}|^{1+o(1)}$ time (i.e. nearly-linear time).

3.2.1 Proof of Theorem 9

We will now formalize the above intuition Theorem 9. It is simple to check that the created loss functions $\tilde{\ell}_t$ have sensitivity at most one. As such, our algorithm is valid in the model. We next proceed to argue its accuracy guarantee.

Recall the definition of ξ from Algorithm 2 of Algorithm 2. Throughout the remainder of the proof, we also let

$$\gamma(m, \rho, \beta) = 2 \cdot \lceil \log m \rceil \cdot \xi(\lceil \log m \rceil, \rho, \beta). \tag{1}$$

Algorithm 2 RecurGap$((\ell_y)_{y\in\mathcal{Y}}; \rho, \beta)$.

Input: Loss functions $(\ell_y)_{y\in\mathcal{Y}}$, **Parameter:** Privacy budget ρ, failure probability β
Output: Index $y_{\text{out}} \in \mathcal{Y}$

if $|\mathcal{Y}| \le 2^{1000}$ or $\beta \le 2^{-K}$ **then**
  return BinTree$((\ell_y)_{y\in\mathcal{Y}}; \rho)$
$K \leftarrow \lceil \log |\mathcal{Y}| \rceil$
$T \leftarrow \left\lceil 2^{3\sqrt{K}-1} \right\rceil$.
$\xi(K, \rho, \beta) \leftarrow \frac{1000}{\sqrt{\rho}}(1 + \log K)^{10} \log(1000(K+1)/\beta)$
$y^* \leftarrow$ minimum index of $\mathcal{Y}$ ▷ Tie broken arbitrarily
for $t \in [T]$ **do**
  Sample uniformly $k_t \sim [K]$.
  $\mathcal{S}_t \leftarrow$ a random subset of $\mathcal{Y}$ of size 2^{K-k_t}.
  $\tilde{\ell}_t \leftarrow \frac{1}{2} \cdot \max\left\{\min_{y\in\mathcal{S}_t} \ell_y - \ell_{y^*} - (K + \sqrt{K}) \cdot \xi(K, \rho, \beta), -\text{gap}\left((\ell_y)_{y\in\mathcal{S}_t}\right)\right\}$
$t_{\text{out}} \leftarrow$ RecurGap $\left((\tilde{\ell}_t)_{t\in[T]}; \frac{4\rho}{5}, \frac{4\beta}{5}\right)$
$y_{\text{out}} \leftarrow$ BinTree $\left((\ell_y)_{y\in\mathcal{S}_{t_{\text{out}}}}; \frac{\rho}{5}\right)$
return y_{out}

Our choice of γ above is based on the fact that we need the following two inequalities to hold. (Here K and T are as defined in Algorithm 2.)

▶ **Lemma 10.** *If* $K \ge 1000$ *and* $\beta \in (0, 0.001)$, *then* $K \cdot \xi(K, \rho, \beta) + 2\gamma\left(T, \frac{4\rho}{5}, \frac{4\beta}{5}\right) \le \gamma(|\mathcal{Y}|, \rho, \beta)$

▶ **Lemma 11.** *If* $K \ge 1000$ *and* $\beta \in (0, 0.001)$, *then* $\sqrt{K} \cdot \xi(K, \rho, \beta) - 2\gamma\left(T, \frac{4\rho}{5}, \frac{4\beta}{5}\right) \ge \tau\left(|\mathcal{Y}|, \frac{\rho}{5}, \frac{\beta}{10}\right)$

The proof of these lemmas (which are mostly tedious calculations) are deferred to Appendix B. For the remainder of the proof, we write ξ as a shorthand for $\xi(K, \rho, \beta)$.

With these lemmas stated, we are now ready to prove Theorem 9. We prove by induction on $|\mathcal{Y}|$ that Algorithm 2 solves Selection with error at most $\gamma(|\mathcal{Y}|, \rho, \beta)$ with probability $1 - \beta$.

Base Case

For $|\mathcal{Y}| \le 2^{1000}$, we simply run BinTree. Similar to the proof of Lemma 8, with probability $1 - \beta$, the noises added in every step in Algorithm 1 is at most $\sqrt{\frac{K \log(K/\beta)}{\rho}}$. When this is the case, the error is at most $2K \cdot \sqrt{\frac{K \log(K/\beta)}{\rho}}$. For $K \le 1000$, it is simple to verify that this is at most $\gamma(|\mathcal{Y}|, \rho, \beta)$.

Inductive Step

Assume that the guarantee holds for all $|\mathcal{Y}| < m$ where $m \ge 2^{1000}$. We will show that this also holds when $|\mathcal{Y}| = m$. First, note that if $\beta \le 2^{-K}$, then we have that $\gamma(|\mathcal{Y}|, \rho, \beta) \ge 2K \cdot \sqrt{\frac{K \log(K/\beta)}{\rho}}$. Thus, as argued above, Algorithm 1 suffices to handle this case. Henceforth, we may assume that $\beta > 2^{-K}$.

We start with the following lemma, which shows that our random sampling procedure results in a "good" set with probability at least $2^{-\Omega(\sqrt{K})}$.

▶ **Lemma 12.** *For each $t \in [T]$,* $\Pr\left[2\tilde{\ell}_t \le -\sqrt{K}\cdot\xi\right] \ge \frac{1}{2^{2\sqrt{K}}}$.

Proof. Let $y^{(1)}, \dots, y^{(2^K)}$ be the elements of $\mathcal{Y}$ ordered in non-decreasing losses. (Note that $y^{(1)} = y^*$.) Let

$$i^* := \operatorname*{argmin}_{i\in\{0,\dots,K\}} \ell_{y^{(2^i)}} - i\cdot\xi.$$

Notice that, by our choice of i^*, we have

$$\ell_{y^{(2^{i^*})}} - \ell_{y^{(1)}} = i^*\cdot\xi + \left(\ell_{y^{(2^{i^*})}} - i^*\cdot\xi\right) - \left(\ell_{y^{(1)}} - 0\cdot\xi\right) \le i^*\cdot\xi \le K\cdot\xi.$$

Rearranging this yields

$$\ell_{y^{(2^{i^*})}} - \ell_{y^{(1)}} - (K+\sqrt{K})\cdot\xi \le -\sqrt{K}\cdot\xi \tag{2}$$

Next, let $k^* := i^* + \lceil\sqrt{K}\rceil$. Consider two cases:

- Case I: $k^* \ge K$. In this case, consider the event $k_t = K$ and the only element of $\mathcal{S}_t$ comes from $\{y^{(1)}, \dots, y^{(2^{i^*})}\}$. This event occurs with probability

  $$\frac{1}{K}\cdot\frac{2^{i^*}}{2^K} \ge \frac{1}{K}\cdot\frac{1}{2^{\sqrt{K}+1}} \ge \frac{1}{2^{2\sqrt{K}}},$$

  where the first inequality follows from $i^* + \lceil\sqrt{K}\rceil = k^* \ge K$ and the second inequality comes from the lower bound on K.
  When these event occurs, we have

  $$2\tilde{\ell}_t \le \ell_{y^{(2^{i^*})}} - \ell_{y^{(1)}} - (K+\sqrt{K})\cdot\xi \overset{(2)}{\le} -\sqrt{K}\cdot\xi,$$

  where, in the first inequality, we also use that fact that $\operatorname{gap}\left((\ell_y)_{y\in\mathcal{S}_t}\right) = \infty$ since $|\mathcal{S}_t| = 1$.
- Case II: $k^* \le K$. In this case, consider the following two events:
  - $\mathcal{E}$: $|\{y^{(1)}, \dots, y^{2^{(i^*)}}\} \cap \mathcal{S}_t| = 1$.
  - $\mathcal{E}'$: $\{y^{(2^{(i^*)}+1)}, \dots, y^{(2^{k^*})}\} \cap \mathcal{S}_t = \emptyset$.

  Notice that, when both of these events hold, we have

  $$\begin{aligned}\operatorname{gap}(\mathcal{S}_t) &\ge \ell_{y^{(2^{k^*}+1)}} - \ell_{y^{(2^{i^*})}} \\ &= (k^*\cdot\xi - i^*\cdot\xi) + \left(\ell_{y^{(2^{k^*})}} - k^*\cdot\xi\right) - \left(\ell_{y^{(2^{i^*})}} - i^*\cdot\xi\right) \\ &\ge (k^*\cdot\xi - i^*\cdot\xi) \\ &\ge \sqrt{K}\cdot\xi,\end{aligned}$$

  where the second inequality is due to our choice of i^* and the third is due to the choice of k^*.
  Recall also Equation (2). As a result, when these events occur, we have $2\tilde{\ell}_t \le -\sqrt{K}\cdot\xi$.
  Finally, notice that the probability that these events occur can be bounded as follows:

  $$\begin{aligned}\Pr[\mathcal{E}\wedge\mathcal{E}'] &\ge \Pr[k_t = k^*]\cdot\Pr[\mathcal{E}\wedge\mathcal{E}' \mid k_t = k^*] \\ &= \frac{1}{K}\cdot\frac{2^{i^*}\cdot\binom{2^K - 2^{k^*}}{2^{K-k^*}-1}}{\binom{2^K}{2^{K-k^*}}} \\ &= \frac{1}{K}\cdot\frac{2^{i^*}\cdot\binom{2^K - 2^{k^*}}{2^{K-k^*}-1}}{\frac{2^K}{2^{K-k^*}}\cdot\binom{2^K-1}{2^{K-k^*}-1}}\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{K} \cdot \frac{1}{2^{k^*-i^*}} \cdot \prod_{j=0}^{2^{K-k^*}-2} \frac{2^K - 2^{k^*} - j}{2^K - 1 - j} \\
&= \frac{1}{K} \cdot \frac{1}{2^{k^*-i^*}} \cdot \prod_{j=0}^{2^{K-k^*}-2} \left(1 - \frac{2^{k^*} - 1}{2^K - 1 - j}\right) \\
&\geq \frac{1}{K} \cdot \frac{1}{2^{\sqrt{K}+1}} \cdot \left(1 - \frac{2^{k^*} - 1}{2^K - 2^{K-k^*} + 1}\right)^{2^{K-k^*}} \\
&\geq \frac{1}{K} \cdot \frac{1}{2^{\sqrt{K}+1}} \cdot \left(1 - \frac{2^{k^*}}{2^K - 2^{K-k^*}}\right)^{2^{K-k^*}}.
\end{aligned}
$$

The last term on the RHS can be bounded using the Bernoulli's inequality as follows. (Recall that $k^* \leq K-1$ in this case.)

$$
\begin{aligned}
\left(1 - \frac{2^{k^*}}{2^K - 2^{K-k^*}}\right)^{2^{K-k^*}} &= \left(\left(1 - \frac{2^{k^*}}{2^K - 2^{K-k^*}}\right)^{2^{K-k^*-1}}\right)^2 \\
&\geq \left(1 - \frac{2^{K-1}}{2^K - 2^{K-k^*}}\right)^2 \\
&= \left(1 - \frac{1}{2 - 2^{1-k^*}}\right)^2 \\
&\geq \left(1 - \frac{1}{1.5}\right)^2 \geq 0.1,
\end{aligned}
$$

where we use the Bernoulli's inequality in the first inequality above, and the second inequality follows from $\sqrt{K} > 1$.
Combining the above two inequality, we get

$$\Pr[\mathcal{E} \wedge \mathcal{E}'] \geq \frac{1}{K} \cdot \frac{1}{2^{\sqrt{K}+1}} \cdot 0.1 \geq \frac{1}{2^{2\sqrt{K}}}.$$

Thus, in both cases, the desired inequality holds. ◀

With the above lemma ready, we are now ready to finish the accuracy proof.

Good Events

Consider the following "good" events:

- $\mathcal{E}_1$: $\min_{t\in[T]} 2\tilde{\ell}_t \leq -\sqrt{K} \cdot \xi$
- $\mathcal{E}_2$: $\tilde{\ell}_{t_{\text{out}}} \leq \gamma\left(T, \frac{4\rho}{5}, \frac{4\beta}{5}\right) + \min_{t\in[T]} \tilde{\ell}_t$
- $\mathcal{E}_3 : \ell_{y_{\text{out}}} = \min_{y\in\mathcal{S}_{t_{\text{out}}}} \ell_y$

From Good Events to Accuracy

Before we bound the probability that these event occurs, let us show argue that, when these good events occur, we get the desired accuracy. To do this, first note that $\mathcal{E}_1, \mathcal{E}_2$ together implies

$$
\begin{aligned}
&2\gamma\left(T, \frac{4\rho}{5}, \frac{4\beta}{5}\right) - \sqrt{K} \cdot \xi \\
&\geq 2\tilde{\ell}_{t_{\text{out}}} = \max\left\{\min_{y\in\mathcal{S}_{t_{\text{out}}}} \ell_y - \ell_{y^*} - (K + \sqrt{K}) \cdot \xi(K, \rho, \beta), -\text{gap}\left((\ell_y)_{y\in\mathcal{S}_{t_{\text{out}}}}\right)\right\}. \qquad (3)
\end{aligned}
$$

By focusing on just the first term in the max in (3), we get

$$\min_{y \in \mathcal{S}_{t_{\text{out}}}} \ell_y \leq \ell_{y^{(1)}} + K \cdot \xi + 2\gamma \left(T, \frac{4\rho}{5}, \frac{4\beta}{5} \right).$$

Then, from $\mathcal{E}_3$, we have

$$\ell_{y_{\text{out}}} = \min_{y \in \mathcal{S}_{t_{\text{out}}}} \ell_y \leq \ell_{y^{(1)}} + K \cdot \xi + 2\gamma \left(T, \frac{4\rho}{5}, \frac{4\beta}{5} \right) \overset{\text{(Lemma 10)}}{\leq} \ell_{y^{(1)}} + \gamma(|\mathcal{Y}|, \rho, \beta), \tag{4}$$

as desired.

Bounding the Probability of Good Events

We have

$$\Pr[\mathcal{E}_1 \wedge \mathcal{E}_2 \wedge \mathcal{E}_3] = \Pr[\mathcal{E}_3 \mid \mathcal{E}_1 \wedge \mathcal{E}_2] \cdot \Pr[\mathcal{E}_1 \wedge \mathcal{E}_2] \geq \Pr[\mathcal{E}_3 \mid \mathcal{E}_1 \wedge \mathcal{E}_2] - \Pr[\neg \mathcal{E}_1] - \Pr[\neg \mathcal{E}_2]. \tag{5}$$

We next bound each term separately.

Bounding $\Pr[\mathcal{E}_3 \mid \mathcal{E}_1 \wedge \mathcal{E}_2]$

Assume that both $\mathcal{E}_1, \mathcal{E}_2$ hold. By focusing on just the second term in the max in (3), we have

$$\text{gap}((\ell_y)_{y \in \mathcal{S}_{t_{\text{out}}}}) \geq \sqrt{K} \cdot \xi - 2\gamma \left(T, \frac{4\rho}{5}, \frac{4\beta}{5} \right) \overset{\text{(Lemma 11)}}{\geq} \tau \left(|\mathcal{Y}|, \frac{\rho}{5}, \frac{\beta}{10} \right).$$

Thus, by Lemma 8, $\Pr[\mathcal{E}_3 \mid \mathcal{E}_1 \wedge \mathcal{E}_2] \geq 1 - \beta/10$.

Bounding $\Pr[\neg \mathcal{E}_1]$

From Lemma 12, we have

$$\Pr[\neg \mathcal{E}_1] \leq \left(1 - \frac{1}{2^{2\sqrt{K}}} \right)^T \leq e^{-\frac{T}{2^{2\sqrt{K}}}} \leq e^{-2^{\sqrt{K}-1}} \leq \beta/10,$$

where the penultimate inequality is from our choice of T and the last inequality is from our assumptions $K \geq 1000$, which implies $\frac{2^{-K}}{10} \geq e^{-2^{\sqrt{K}-1}}$, and $\beta \geq 2^{-K}$.

Bounding $\Pr[\neg \mathcal{E}_2]$

It follows immediately from the inductive hypothesis that $\Pr[\neg \mathcal{E}_2] \leq 4\beta/5$.

Putting Things Together

Putting the three bounds together with Equation (5), we have that $\Pr[\mathcal{E}_1 \wedge \mathcal{E}_2 \wedge \mathcal{E}_3] \geq 1 - \beta$. As argued earlier, when $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$ all occur, Equation (4) holds. This concludes our proof of the inductive step.

Algorithm 3 $\textsc{Combined}((\ell_y)_{y\in\mathcal{Y}};\rho)$.

Input: Loss functions $(\ell_y)_{y\in\mathcal{Y}}$, **Parameter:** Privacy budget ρ
Output: Index $y_{\text{out}} \in \mathcal{Y}$

$K \leftarrow \lceil \log|\mathcal{Y}| \rceil$
$\beta \leftarrow \frac{1}{K}$
$y^1_{\text{out}} \leftarrow \textsc{RecurGap}\left((\ell_y)_{y\in\mathcal{Y}}; \frac{\rho}{3}, \beta\right)$
$y^2_{\text{out}} \leftarrow \textsc{BinTree}\left((\ell_y)_{y\in\mathcal{Y}}; \frac{\rho}{3}\right)$
Query $q := \frac{1}{2}\left(\ell_{y^1_{\text{out}}}(X) - \ell_{y^2_{\text{out}}}(X)\right)$ with budget $\frac{\rho}{3}$; let the answer be $\tilde{q}$
if $\tilde{q} > 0$ **then**
    $y_{\text{out}} \leftarrow y^2_{\text{out}}$
else
    $y_{\text{out}} \leftarrow y^1_{\text{out}}$
return y_{out}

3.3 From High-Probability to Expected Error: Proof of Theorem 5

Finally, we can turn the high-probability guarantee from Theorem 9 to an expected error guarantee. The idea is simple: We run our algorithm and the binary tree algorithm. We then use a query to compare the two outputs and select the best one. This is described more formally in Algorithm 3 and analyzed below.

Proof of Theorem 5. Let $Z = q - \tilde{q}$ denote the noise added to the query q. Recall that $Z \sim \mathcal{N}\left(0, \frac{3}{2\rho}\right)$. Note that $\ell_{y_{\text{out}}} - \min\left\{\ell_{y^1_{\text{out}}}, \ell_{y^2_{\text{out}}}\right\} \le 2|Z|$. As a result, we have

$$\mathbb{E}[\ell_{y_{\text{out}}}] \le \mathbb{E}\left[\min\left\{\ell_{y^1_{\text{out}}}, \ell_{y^2_{\text{out}}}\right\}\right] + \mathbb{E}[2|Z|] \le \mathbb{E}\left[\min\left\{\ell_{y^1_{\text{out}}}, \ell_{y^2_{\text{out}}}\right\}\right] + O\left(\frac{1}{\sqrt{\rho}}\right).$$

Next, let $\gamma = O\left(\frac{1}{\sqrt{\rho}} \cdot \log|\mathcal{Y}| \cdot (\log\log|\mathcal{Y}|)^{10} \cdot \log\left(\frac{\log|\mathcal{Y}|}{\beta}\right)\right)$ denote the error guarantee from Theorem 9. From this, consider y' defined as follows:

$$y' = \begin{cases} y^1_{\text{out}} & \text{if } \ell_{y^1_{\text{out}}} \le \min_{y\in\mathcal{Y}} \ell_y + \gamma, \\ y^2_{\text{out}} & \text{otherwise.} \end{cases}$$

Clearly, $\ell_{y'} \ge \min\left\{\ell_{y^1_{\text{out}}}, \ell_{y^2_{\text{out}}}\right\}$. Also recall that $K = \lceil \log|\mathcal{Y}| \rceil$. Thus, we have

$$\begin{aligned}
&\mathbb{E}\left[\min\left\{\ell_{y^1_{\text{out}}}, \ell_{y^2_{\text{out}}}\right\}\right] \\
&\le \mathbb{E}\left[\ell_{y'}\right] \\
&= \mathbb{E}\left[\ell_{y^1_{\text{out}}} \;\middle|\; \ell_{y^1_{\text{out}}} \le \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \cdot \Pr\left[\ell_{y^1_{\text{out}}} \le \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \\
&\quad + \mathbb{E}\left[\ell_{y^2_{\text{out}}} \;\middle|\; \ell_{y^1_{\text{out}}} > \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \cdot \Pr\left[\ell_{y^1_{\text{out}}} > \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \\
&= \mathbb{E}\left[\ell_{y^1_{\text{out}}} \;\middle|\; \ell_{y^1_{\text{out}}} \le \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \cdot \Pr\left[\ell_{y^1_{\text{out}}} \le \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \\
&\quad + \mathbb{E}\left[\ell_{y^2_{\text{out}}}\right] \cdot \Pr\left[\ell_{y^1_{\text{out}}} > \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \qquad (6) \\
&\le \left(\min_{y\in\mathcal{Y}} \ell_y + \gamma\right) \cdot \Pr\left[\ell_{y^1_{\text{out}}} \le \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \\
&\quad + \left(\min_{y\in\mathcal{Y}} \ell_y + O\left(\frac{K^{3/2}}{\sqrt{\rho}}\right)\right) \cdot \Pr\left[\ell_{y^1_{\text{out}}} > \min_{y\in\mathcal{Y}} \ell_y + \gamma\right] \qquad (7)
\end{aligned}$$

$$\leq \min_{y\in\mathcal{Y}} \ell_y + \gamma + O\left(\frac{K^{3/2}}{\sqrt{\rho}}\right) \cdot \Pr\left[\ell_{y^1_{\text{out}}} > \min_{y\in\mathcal{Y}} \ell_y + \gamma\right]$$

$$\leq \min_{y\in\mathcal{Y}} \ell_y + \gamma + O\left(\frac{K^{3/2}}{\sqrt{\rho}}\right) \cdot \beta \tag{8}$$

$$\leq \min_{y\in\mathcal{Y}} \ell_y + O\left(\frac{1}{\sqrt{\rho}} \cdot K(\log K)^{11}\right), \tag{9}$$

where (6) is due to the fact that y^2_{out} is independent of the event $\ell_{y^1_{\text{out}}} > \min_{y\in\mathcal{Y}} \ell_y + \gamma$, (7) follows from Lemma 4, (8) follows from the guarantee of RecurGap (Theorem 9), and (9) is due to our setting $\beta = \frac{1}{K}$ in Algorithm (3).

Thus, we can conclude that $\mathbb{E}[\ell_{y_{\text{out}}}] \leq \min_{y\in\mathcal{Y}} \ell_y + O\left(\frac{1}{\sqrt{\rho}} \cdot K(\log K)^{11}\right)$ as desired. ◀

References

1. Raef Bassily, Adam D. Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *FOCS*, pages 464–473, 2014. doi:10.1109/FOCS.2014.56.
2. Amos Beimel, Kobbi Nissim, and Uri Stemmer. Private learning and sanitization: Pure vs. approximate differential privacy. *Theory Comput.*, 12(1):1–61, 2016. doi:10.4086/TOC.2016.V012A001.
3. Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *TCC*, pages 635–658, 2016. doi:10.1007/978-3-662-53641-4_24.
4. Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, pages 486–503, 2006. doi:10.1007/11761679_29.
5. Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284, 2006. doi:10.1007/11681878_14.
6. Cynthia Dwork and Guy N. Rothblum. Concentrated differential privacy. *CoRR*, abs/1603.01887, 2016. arXiv:1603.01887.
7. Anupam Gupta, Katrina Ligett, Frank McSherry, Aaron Roth, and Kunal Talwar. Differentially private combinatorial optimization. In *SODA*, pages 1106–1125, 2010. doi:10.1137/1.9781611973075.90.
8. Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *FOCS*, pages 94–103, 2007. doi:10.1109/FOCS.2007.41.
9. Thomas Steinke. Open problem: Selection via low-sensitivity queries. DifferentialPrivacy.org, May 2025. URL: https://differentialprivacy.org/open-problem-selection/.
10. Justin Whitehouse, Aaditya Ramdas, Ryan Rogers, and Steven Wu. Fully-adaptive composition in differential privacy. In *ICML*, pages 36990–37007, 2023. URL: https://proceedings.mlr.press/v202/whitehouse23a.html.

A On Equal vs Unequal Budgets

We use a model (Definition 2) in which we are allowed to associate different budgets to the different queries. Strictly speaking, this is not the same as the model proposed in [9], which uses the same budget for all queries. This is stated more precisely below.

▶ **Definition 13** (Gaussian Mechanism with Equal Budget [9]). *In the* Gaussian mechanism with equal budget *model, the algorithm is given a finite budget $\rho > 0$ and can select the number of rounds M. At the i-th iteration, the algorithm can issue a sensitivity-1 query $q_i : \mathcal{X} \to \mathbb{R}$, and it will receive back $q_i(X) + Z_i$ where $Z_i \sim \mathcal{N}\left(0, \frac{M}{2\rho}\right)$.*

Nevertheless, using the fact that the sum of Gaussian noises remain Gaussian, it is simple to show that the two models are equivalent, as formalized below. Note that the condition (that an upper bound on the number of rounds M is known) is quite mild, and it can be easily seen that all algorithms discussed in this paper (e.g. Algorithm 2) satisfy this condition.

▶ **Lemma 14.** *Let $\mathcal{A}$ be any algorithm in the Gaussian mechanism with budget model that works in at most M rounds and uses total budget ρ. Then, $\mathcal{A}$ can be implemented in the Gaussian mechanism with equal budget model with total budget 2ρ.*

Proof. We simulate $\mathcal{A}$ using $M' = 2M$ rounds in the Gaussian mechanism with equal budget model with total budget $\rho' = 2\rho$ as follows:

- At round i of $\mathcal{A}$, suppose that $\mathcal{A}$ issues a query q_i with budget ρ_i.
- We simulate it in the equal budget model by issuing q_i a total of $m_i = \left\lceil \frac{M'\rho_i}{\rho'} \right\rceil$ times to get back answers $\tilde{q}_i^1, \ldots, \tilde{q}_{m_i}^i$.
- We then send back the answer $\tilde{q}_i = \frac{1}{m_i}\left(\tilde{q}_i^1 + \cdots + \tilde{q}_{m_i}^i\right) + \mathcal{N}\left(0, \frac{1}{2\rho_i} - \frac{M'}{2\rho' m_i}\right)$.

It is simple to verify that the distribution of the answer $\tilde{q}_i$ is the same as that in the Gaussian mechanism with budget model, and that the total budget used is no more than 2ρ as desired. ◀

We note that our reduction does *not* work for Laplace mechanism because the sum of Laplace noises does not follow the Laplace distribution.

B Proof of Lemmas 10 and 11

To prove these lemmas, we will use the following proposition. Note that here and throughout, we allows the first arguments of ξ, γ to be any positive real numbers.

▶ **Proposition 15.** *For all $K \in \mathbb{R}$ such that $K \geq 1000$ and for all $\beta \in \mathbb{R}$ such that $0 < \beta \leq 0.001$, the following inequality holds:*

$$\xi(K, \rho, \beta) \geq 36 \cdot \xi\left(3\sqrt{K}, \frac{4\rho}{5}, \frac{4\beta}{5}\right)$$

Proof. We begin by substituting the definition of ξ into the inequality. The inequality becomes:

$$\frac{1000}{\sqrt{\rho}}(1+\log_2 K)^{10}\log_2\left(\frac{1000(K+1)}{\beta}\right) \geq 18\sqrt{5}\cdot\frac{1000}{\sqrt{\rho}}\left(1+\log_2(3\sqrt{K})\right)^{10}\log_2\left(\frac{1250(3\sqrt{K}+1)}{\beta}\right)$$

Rearranging yields the equivalent inequality:

$$\left(\frac{1+\log_2 K}{1+\log_2(3\sqrt{K})}\right)^{10} \cdot \frac{\log_2\left(\frac{250}{\beta}\cdot 4(K+1)\right)}{\log_2\left(\frac{250}{\beta}\cdot 5(3\sqrt{K}+1)\right)} \geq 18\sqrt{5}$$

The proof proceeds by establishing lower bounds for the two multiplicative terms on the left.

Lower Bound for the First Term. Let us analyze the first term, which we denote $R_K(K)$:

$$R_K(K) = \left(\frac{1+\log_2 K}{1+\log_2(3\sqrt{K})}\right)^{10} = \left(\frac{1+\log_2 K}{1+\log_2 3+\frac{1}{2}\log_2 K}\right)^{10}$$

Consider the base of this term as a function of $x = \log_2 K$. Let $g(x) = \frac{1+x}{1+\log_2 3+0.5x}$. The derivative with respect to x is:

$$g'(x) = \frac{(1)(1+\log_2 3+0.5x) - (1+x)(0.5)}{(1+\log_2 3+0.5x)^2} = \frac{0.5+\log_2 3}{(1+\log_2 3+0.5x)^2} > 0,$$

which implies that $g(x)$ is a strictly increasing function of x. For the domain $K \geq 1000$, the minimum value of $g(x)$ occurs at the minimum value of x, which is $x_{\min} = \log_2 1000$. We have $g(x) \geq g(x_{\min}) > 1.448$. Thus, $R_K(K) > (1.448)^{10} > 40.5$.

Lower Bound for the Second Term. We claim that the second term is at least one. This is because $4(K+1) - 5(3\sqrt{K}+1) = 3\sqrt{K}(\sqrt{K}-5) + (K-1) > 0$ under our assumption $K \geq 1000$.

Conclusion. By combining the lower bounds established in the preceding paragraphs, we have:

$$\left(\frac{1+\log_2 K}{1+\log_2(3\sqrt{K})}\right)^{10} \cdot \frac{\log_2\left(\frac{250}{\beta}\cdot 4(K+1)\right)}{\log_2\left(\frac{250}{\beta}\cdot 5(3\sqrt{K}+1)\right)} > 40.5 > 18\sqrt{5}. \quad \blacktriangleleft$$

Lemmas 10 and 11 now follow rather simply from the above proposition.

Proof of Lemma 10. Notice that ξ is non-decreasing in its first argument and that $\lceil \log T \rceil \leq 3\sqrt{K}$. Thus, we have

$$\begin{aligned}
K \cdot \xi\left(K,\rho,\beta\right) + 2\gamma\left(T, \frac{4\rho}{5}, \frac{4\beta}{5}\right) &= K \cdot \xi(K,\rho,\beta) + 2 \cdot 2\lceil \log T \rceil \cdot \xi\left(\lceil \log T \rceil, \frac{4\rho}{5}, \frac{4\beta}{5}\right) \\
&\leq K \cdot \xi(K,\rho,\beta) + 12\sqrt{K} \cdot \xi\left(3\sqrt{K}, \frac{4\rho}{5}, \frac{4\beta}{5}\right) \\
\text{(Proposition 15)} &\leq \left(K + \frac{\sqrt{K}}{3}\right) \cdot \xi(K,\rho,\beta) \\
&\leq 2K \cdot \xi(K,\rho,\beta) \\
&= \gamma(|\mathcal{Y}|,\rho,\beta). \quad \blacktriangleleft
\end{aligned}$$

Proof of Lemma 11. Similar to the proof of Lemma 10, we have

$$\begin{aligned}
\sqrt{K} \cdot \xi\left(K,\rho,\beta\right) - 2\gamma\left(T, \frac{4\rho}{5}, \frac{4\beta}{5}\right) &= \sqrt{K} \cdot \xi\left(K,\rho,\beta\right) - 2 \cdot 2\lceil \log T \rceil \cdot \xi\left(\lceil \log T \rceil, \frac{4\rho}{5}, \frac{4\beta}{5}\right) \\
&\geq \sqrt{K} \cdot \xi\left(K,\rho,\beta\right) - 12\sqrt{K} \cdot \xi\left(3\sqrt{K}, \frac{4\rho}{5}, \frac{4\beta}{5}\right) \\
\text{(Proposition 15)} &\geq \frac{\sqrt{K}}{2} \cdot \xi\left(K,\rho,\beta\right) \\
&= \frac{\sqrt{K}}{2} \cdot \frac{1000}{\sqrt{\rho}} (1+\log K)^{10} \log(1000(K+1)/\beta) \\
&\geq \frac{\sqrt{K}}{2} \cdot \frac{4\sqrt{5}}{\sqrt{\rho}} \cdot 1 \cdot \sqrt{\log(10K/\beta)} \\
&= \tau\left(|\mathcal{Y}|, \frac{\rho}{5}, \frac{\beta}{10}\right) \quad \blacktriangleleft
\end{aligned}$$