A. Near-Matching Lower Bound: Theorem 3

Our function f witnessing the lower bound of Theorem 3 will be a variant of the well-known $\textsc{Tribes}_\ell : \{0,1\}^\ell \to \{0,1\}$ function from the analysis of boolean functions. Recall that $\textsc{Tribes}_\ell$ is a read-once DNF formula with $m \coloneqq \lfloor \frac{\ell}{w} \rfloor$ terms of width exactly w over disjoint variables (with some variables possibly left unused),

$$\textsc{Tribes}_\ell(x) \coloneqq T_1(x) \vee \cdots \vee T_m(x),$$

and $w \coloneqq \log \ell - \log \ln \ell + o_\ell(1)$ is chosen so that $\Pr[\textsc{Tribes}_\ell(\boldsymbol{x}) = 1]$ is as close to $\frac{1}{2}$ as possible.[4] (For more on the $\textsc{Tribes}$ function, see Chapter §4.2 of (O'Donnell, 2014).) Our construction will also involve the Majority function, $\textsc{Maj}_k(y) \coloneqq \mathbb{1}[\sum_{i=1}^k y_i \geq 0]$.

The function that witnesses the separation. Let $m' < m$ be chosen so that the function $\textsc{Tribes}'_\ell : \{0,1\}^\ell \to \{0,1\}$,

$$\textsc{Tribes}'_\ell(x) \coloneqq T_1(x) \vee \cdots \vee T_{m'}(x),$$

has acceptance probability $\Pr[\textsc{Tribes}'_\ell(\boldsymbol{x}) = 1]$ as close to 0.499 as possible.[5] We additionally define $\textsc{Rest}_\ell : \{0,1\}^\ell \to \{0,1\}$ to be:

$$\textsc{Rest}_\ell(x) \coloneqq \textsc{Tribes}_\ell(x) \wedge \neg\, \textsc{Tribes}'_\ell(x),$$

nothing that $\Pr[\textsc{Rest}_\ell(\boldsymbol{x}) = 1] = 0.001$.

Our function $f_{\ell,k} : \{0,1\}^\ell \times \{0,1\}^k \to \{0,1\}$ is defined as follows:

$$f_{\ell,k}(x,y) \coloneqq \begin{cases} 1 & \text{if } \textsc{Tribes}'_\ell(x) = 1 \\ \textsc{Maj}_k(y) & \text{if } \textsc{Rest}_\ell(x) = 1 \\ 0 & \text{otherwise,} \end{cases}$$

where k and ℓ are chosen so that Equation (2) in the statement of Lemma A.2 below is satisfied with equality.

Proposition A.1 (Upper bound on opt)**.** $\mathsf{opt}_{f_{\ell,k},2^\ell} \leq 0.01$.

Proof. We have that

$$\begin{aligned} \Pr[f_{\ell,k}(\boldsymbol{x},\boldsymbol{y}) \neq \textsc{Tribes}_\ell(\boldsymbol{x})] &= \Pr[\textsc{Rest}_\ell(\boldsymbol{x}) = 1 \text{ and } \textsc{Maj}_k(\boldsymbol{y}) \neq \textsc{Tribes}_\ell(\boldsymbol{x})] \\ &= \Pr[\textsc{Rest}_\ell(\boldsymbol{x}) = 1] \cdot \Pr[\textsc{Maj}_k(\boldsymbol{y}) = 0] \\ &= 0.001 \cdot \tfrac{1}{2} \;<\; 0.01. \end{aligned}$$

Since $\textsc{Tribes}_\ell$ is computed by a decision tree of size $\leq 2^\ell$, it follows that that $\mathsf{opt}_{f_{\ell,k},2^\ell} \leq 0.01$. □

Lemma A.2 (Lower bound on $\textsc{TopDownError}$)**.** *There are universal constants c_1 and c_2 such that the following holds. Suppose k and ℓ satisfy:*

$$\frac{c_1}{\sqrt{k}} \geq \frac{\log \ell}{\ell}. \tag{2}$$

Then $\textsc{TopDownError}(f_{\ell,k}, 2^{\,c_2 k/\log k}) \geq 0.49$.

Proof of Theorem 3 assuming Lemma A.2. We choose k and ℓ such that Equation (2) is satisfied with equality, and define $s \coloneqq 2^\ell$. Theorem 3 follows as an immediate consequence of Proposition A.1 and Lemma A.2 since $2^{\Omega(k/\log k)} = 2^{\Omega(\ell^2/(\log \ell)^3)} = s^{\tilde{\Omega}(\log s)}$. □

[4] While this acceptance probability cannot be made *exactly* $\frac{1}{2}$ for all values of ℓ due to granularity issues, it will be the case that $\Pr[\textsc{Tribes}_\ell(\boldsymbol{x})] = \frac{1}{2} \pm O(\frac{\log \ell}{\ell})$. For clarity we will assume for the remainder of this proof the acceptance probability of $\textsc{Tribes}_\ell$ is exactly $\frac{1}{2}$, noting that the same calculations go through if one carries around the additive $o_\ell(1)$ factor.

[5] The same remark as in the previous footnote applies here.

A.1. Proof of Lemma A.2

Our proof of Lemma A.2 draws on many of the ideas in (Blanc et al., 2020)'s proof of their Theorem 6(b). Let T denote the tree constructed by $\textsc{BuildTopDownDT}(f_{\ell,k}, 2^{c_2 k/\log k})$, where c_2 is a universal constant that will be determined later.

Claim A.3. *Let $\overline{\pi}$ be a path in T that leads to a first query to an x-variable. Then $\mathrm{Inf}_{y_j}(\textsc{Maj}_k(y)_{\overline{\pi}}) \leq \frac{1}{100\sqrt{k}}$ for all $j \in [k]$.*

Proof. Suppose without loss of generality that $\overline{\pi}$ leads to a query to x_1. By the splitting criterion of $\textsc{BuildTopDownDT}$ and Fact 2.1, we have that

$$\mathrm{Inf}_{x_1}(f_{\overline{\pi}}) \geq \mathrm{Inf}_{y_j}(f_{\overline{\pi}}) \quad \text{for all } j \in [k].$$

Since

$$\mathrm{Inf}_{x_1}(f_{\overline{\pi}}) \leq \mathrm{Inf}_{x_1}(\textsc{Tribes}_\ell) + \mathrm{Inf}_{x_1}(\textsc{Tribes}'_\ell) \leq O\left(\frac{\log \ell}{\ell}\right)($$

and

$$\mathrm{Inf}_{y_j}(f_{\overline{\pi}}) = \Pr[\textsc{Rest}_\ell(\boldsymbol{x}) = 1] \cdot \mathrm{Inf}_{y_j}(\textsc{Maj}_k(y)_{\overline{\pi}}) = 0.001 \cdot \mathrm{Inf}_{y_j}(\textsc{Maj}_k(y)_{\overline{\pi}}),$$

it follows that $\mathrm{Inf}_{y_j}(\textsc{Maj}_k(y)_\pi) \leq O(\frac{\log \ell}{\ell})$. The claim follows by choosing c_1 to be a sufficiently small constant in Equation (2). □

Corollary A.4. *There is a universal constant c_3 such that the following holds. Let $(\boldsymbol{x}, \boldsymbol{y}) \sim \{0,1\}^k \times \{0,1\}^\ell$ be a uniform random input, and $\pi_{(\boldsymbol{x},\boldsymbol{y})}$ be the corresponding root-to-leaf path in T that $(\boldsymbol{x}, \boldsymbol{y})$ follows. The probability that $\pi_{(\boldsymbol{x},\boldsymbol{y})}$ queries an x-variable before at least $c_3 k/\log k$ many y-variables is at most 0.001.*

Proof. Call an input (x, y) *bad* if $\pi_{(x,y)}$ queries an x-variable before $c_3 k$ many y-variables. Let $\overline{\pi}$ denote the truncation of $\pi_{(x,y)}$ to its prefix before the first query to an x-variable. By Claim A.3, we have that $\mathrm{Inf}_{y_j}(\textsc{Maj}_k(y)_{\overline{\pi}}) \leq \frac{1}{100\sqrt{k}}$, and so the discrepancy between the number of 0's and 1's in $\overline{\pi}$ must be $\Omega(\sqrt{k})$. Therefore, we can bound

$$\begin{aligned}
\Pr[(\boldsymbol{x}, \boldsymbol{y}) \text{ is bad}] &\leq \sum_{t=1}^{c_3 k/\log k} \left(\Pr_{\boldsymbol{b} \sim \mathrm{Bin}(t, \frac{1}{2})} \left[\left| \boldsymbol{b} - \tfrac{t}{2} \right| \geq \Omega(\sqrt{k}) \right] \right) \left(\right. \\
&\leq \sum_{t=1}^{c_3 k/\log k} \left(e^{-\Theta(k/t)} \right) && \text{(Hoeffding's inequality)} \\
&\leq c_3 k \cdot e^{-\Theta(\log k/c_3)} \leq o_k(1).
\end{aligned}$$

where the final inequality holds by choosing c_3 to be a sufficiently small constant. □

We are now ready to prove Lemma A.2. Let $\xi : \{0,1\}^k \times \{0,1\}^\ell \to \{0,1\}$ be the indicator

$$\xi(x, y) := \mathbb{1}\left[\left(\pi_{(x,y)} \text{ queries an } x\text{-variable before at least } \frac{c_3 k}{\log k} \text{ many } y\text{-variables} \right] \right) \left(\right.$$

By Corollary A.4 we have that $\Pr[\xi(\boldsymbol{x}, \boldsymbol{y}) = 1] \leq 0.001$. If (x, y) is such that $\Pr[\xi(x, y) = 0]$, then either

1. $|\pi_{(x,y)}| \geq \dfrac{c_3 k}{\log k}$, or
2. $|\pi_{(x,y)}| < \dfrac{c_3 k}{\log k}$ and $\pi_{(x,y)}$ does not query any x-variables.

Since the fraction of inputs that follow any specific path of length $\geq c_3 k/\log k$ is at most $2^{-c_3 k/\log k}$, and the size of T is $2^{c_2 k/\log k}$ by assumption, choosing $c_2 = \frac{1}{2} c_3$ ensures that the fraction of inputs (x, y) such that $\pi_{(x,y)}$ falls into the first case above (i.e. $|\pi_{(x,y)}| \geq c_3 k/\log k$) is at most 0.001.

Therefore, at least a 0.998 fraction of inputs (x, y) are such that $\pi_{(x,y)}$ falls into the second case above. For such (x, y)'s, we have that

$$\Pr[f_{\pi_{(x,y)}} = 1] \geq \Pr[\textsc{Tribes}'_\ell = 1] = 0.499$$
$$\Pr[f_{\pi_{(x,y)}} = 0] \geq \Pr[\textsc{Tribes}_\ell = 0] = 0.5,$$

and so $\Pr[T_{\pi_{(x,y)}} \neq f_{\pi_{(x,y)}}] \geq 0.49$. We conclude that $\Pr[T(\boldsymbol{x}, \boldsymbol{y}) \neq f(\boldsymbol{x}, \boldsymbol{y})] \geq 0.998 \times 0.499 > 0.49$, which completes the proof of Lemma A.2.

B. Proof of Theorem 1

We first prove Theorem 5.

Proof of Theorem 5. First, without loss of generality, we can assume that $\mathcal{D}$ is the uniform distribution on $[0, 1]^n$. Otherwise, we can transform each variable by its CDF. This also means that we will set $\theta^\star = \frac{1}{2}$.

We will use Theorem 4 as a black box in our proof of Theorem 5. In order to do this, we will relate T, a decision tree with real-valued inputs, to a specially constructed decision tree with boolean-valued inputs. Each input of T will be encoded into a boolean vector of width $w = O(\log(s/\varepsilon))$ using the encoder $E : [0, 1] \to \{\pm 1\}^w$ defined as follows.

$$E(x) = \textsc{Binary}(\lfloor x \cdot 2^w \rfloor),$$

where BINARY is the function that encodes an integer in binary form. We want to ensure that if two inputs are encoded as the same Boolean vector evaluate to the same value. This requires rounding the thresholds in T.

Lemma B.1. *Let $T : [0, 1]^n \to \{0, 1\}$ be a balanced size-s tree and $\varepsilon > 0$. Then for $w = O(\log(s/\varepsilon))$ let T_{round} be the tree formed by rounding all thresholds of T to the nearest 2^{-w}. Then $\mathrm{dist}(T, T_{round}) \leq \varepsilon/2$.*

Proof of Lemma B.1. Since T and T_{round} have the same label at every leaf, if $T(x) \neq T_{\text{round}}(x)$, it means that x reaches different leaves in T and T_{round}. Fix some leaf ℓ of T. Then, ℓ is reached by some subcube of inputs of the form.

$$a_i \leq x_i < b_i \quad \text{for } i = 1 \in [n]$$

The corresponding leaf, ℓ_{round}, of T_{round} is reached by all inputs satisfying the following, where $\mathsf{round}(\cdot)$ rounds an input to the nearest multiple of 2^{-w}.

$$\mathsf{round}(a_i) \leq x_i < \mathsf{round}(b_i) \quad \text{for } i = 1 \in [n]$$

We will upper bound the probability that a randomly chosen $x \in [0, 1]^n$ reaches ℓ in T but does not reach ℓ_{round} in T_{round}. For that to occur, there must be one $i \in [n]$ such that $x_i \in [a_i, \mathsf{round}(a_i)]$ or $x_i \in [\mathsf{round}(b_i), b_i]$. Since $|\mathsf{round}(z) - z| \leq 2^{-w}$, for any fixed i, x_i falls in one of those ranges with probability at most $2 \cdot 2^{-w}$.

Furthermore, since T is balanced, ℓ has depth at most $O(\log(s))$. This means that for all but $O(\log(s))$ choices for i, $a_i = 0$ and $b_i = 1$, in which case there is no chance that $x_i \in [a_i, \mathsf{round}(a_i)]$ or $x_i \in [\mathsf{round}(b_i), b_i]$. By union bounding the up to $O(\log(s))$ input coordinates that matter for ℓ, we have that the probability that x reaches ℓ in T but does not reach ℓ_{round} in T_{round} is at most $O(\log(s) \cdot 2^{-w})$.

By union bound over the s leaves, the probability x reaches a different leaf in T as in T_{round} is at most $O(s \log s \cdot 2^{-w})$. Setting this equal to $\varepsilon/2$ and solving for w yields the desired result. □

We are now ready to complete the proof of Theorem 5. Note that

$$\begin{aligned}
\mathrm{bias}(f) - \mathrm{dist}(f, T_{\text{round}}) &\geq \mathrm{bias}(f) - (\mathrm{dist}(f, T) + \mathrm{dist}(T, T_{\text{round}})) \\
&\geq \mathrm{bias}(f) - \mathrm{dist}(f, T) - \frac{\varepsilon}{2} \\
&\geq \frac{\varepsilon}{2}.
\end{aligned}$$

We will create a tree with Boolean inputs, $S : \{\pm 1\}^{wn} \to \{0,1\}$, satisfying the following relation

$$S(E(x_1), ..., E(x_n)) = T_{\text{round}}(x) \quad \text{for all } x \in [0,1]^n.$$

The decision tree computing S is the tree computing T where each threshold is replaced with the tree of size at most w specifying that threshold. Replacing each node of T with a tree of size w creates a tree that is a factor of w^d larger than T, where d is the depth of T. Using $w = O(\log(s/\varepsilon))$, we then have that S has size

$$O(\log(s/\varepsilon))^d = 2^{O(\log(s)\log\log(s/\varepsilon))}.$$

We are now ready to apply Theorem 4. We have determined that $\text{bias}(f) - \text{dist}(f, T_{\text{round}}) \geq \varepsilon/2$, and that T_{round} is encodable as a decision tree of size $2^{\log(s)\log(\log(s/\varepsilon))}$ with Boolean inputs. By Theorem 4, there is a bit of the encoding with influence, with respect to f, at least $\Omega(\varepsilon/\log(s)\log\log(s/\varepsilon))$.

All that remains is to show that the most influential bit of the encoding, is a single threshold of the form $\mathbb{1}[x_i \geq \frac{1}{2}]$. All bits of the encoding for a variable, x_i, restrict x_i to exactly half of the space in $[0,1]$. Since f is monotone, the most influential such restriction is $\mathbb{1}[x_i \geq \frac{1}{2}]$. Hence, the split $\mathbb{1}[x_i \geq \frac{1}{2}]$ has influence at least $\Omega\left(\varepsilon/\log(s)\log\log(s/\varepsilon)\right)$. Since f is monotone, this is also the correlation of $\mathbb{1}[x_i \geq \frac{1}{2}]$ with f, proving Theorem 5. □

The remainder of the proof of Theorem 1 is the same as the proof of Theorem 2. Let T° be the size-$(j+1)$ partial tree built BUILDTOPDOWNDTafter j iterations. As long as $\text{dist}(f, T^\circ_f) > \mathsf{balanced\_opt}_s + \varepsilon$, we know there is a split that results in purity gain of at least[6]

$$\Omega\left(\frac{\kappa \cdot \varepsilon^2}{j \cdot (\log(s)\log\log(s/\varepsilon))^2}\right)\Bigg($$

Therefore, after we have run for

$$\begin{aligned} t &= 2^{O(\log(s)\log\log(s/\varepsilon))^2/\kappa\varepsilon^2} \\ &= s^{\tilde{O}((\log s)/\kappa\varepsilon^2)} \end{aligned}$$

we must have that $\text{dist}(f, T^\circ_f) < \mathsf{balanced\_opt}_s + \varepsilon$, proving Theorem 1.

C. The Realizable Case

(Blanc et al., 2020)'s work on the realizable setting analyzed the performance of a *variant* of the top-down heuristics; their variant does not correspond to any impurity function $\mathscr{G}$. Consider $\text{BUILDTOPDOWNDT}_{\text{Inf}}$, defined in Figure 2.

[6] From Theorem 5, we actually know there is a split with this much purity that is the median of one input coordinate, although BUILDTOPDOWNDTmay choose one that is not the median if it results in more purity gain.

BUILDTOPDOWNDT$_{\mathrm{Inf}}(f,t)$:

Initialize T° to be the empty tree.

while (size$(T^\circ) < t$ {

1. (Score) For every leaf ℓ in T°, let $x_{i(\ell)}$ denote the most influential variable of the subfunction f_ℓ:

$$\mathrm{Inf}_{i(\ell)}(f_\ell) \geq \mathrm{Inf}_j(f_\ell) \quad \text{for all } j \in [n].$$

   Assign ℓ the score:

$$\mathrm{score}_{\mathrm{Inf}}(\ell) := 2^{-|\ell|} \cdot \mathrm{Inf}_{i(\ell)}(f_\ell),$$

   where $|\ell|$ denotes the depth of ℓ in T°.

2. (Split) Let $\ell^\star$ be the leaf with the highest score. Grow T° by replacing $\ell^\star$ with a query to $x_{i(\ell^\star)}$.

}

Output f-completion of T°.

Figure 2. Top-down heuristic for building a decision tree approximation for f from (Blanc et al., 2020)

Lemma C.1 ((Theorem 5 of (Blanc et al., 2020))**.** *For every $\varepsilon \in (0, \frac{1}{2})$ and monotone function $f : \{\pm 1\}^n \to \{0,1\}$ exactly computable by a size s decision tree,*

$$\textsc{TopDownError}_{\mathrm{Inf}}(f, s^{O(\sqrt{\log s}/\varepsilon)}) \leq \varepsilon.$$

We will show that as a consequence of Lemma C.1, a similar upper bound can be shown for BUILDTOPDOWNDT$_{\mathscr{G}}$ for any strongly-concave $\mathscr{G}$. We first briefly summarize the proof of Lemma C.1 given in (Blanc et al., 2020). They define the following potential function:

$$u_f(T^\circ) := \sum_{\text{leaves } \ell \in T^\circ} 2^{-|\ell|} \cdot \mathrm{Inf}(f_\ell).$$

For monotone functions $f : \{\pm 1\}^n \to \{0,1\}$ that are computable by size-s decision trees, a result of (O'Donnell & Servedio, 2007) gives the following bound:

$$u_f(\text{empty tree}) = \mathrm{Inf}(f) \leq \sqrt{\log s}.$$

Furthermore, they show that if T° is the size-$(j+1)$ tree built after j iterations of BUILDTOPDOWNDT$_{\mathrm{Inf}}(f)$, and $\mathrm{dist}(f, T^\circ_f) \geq \varepsilon$, then the leaf, $\ell^\star$, selected in the next iteration satisfies.

$$\mathrm{score}(\ell^\star) \geq \frac{\varepsilon}{(j+1)\log s}.$$

Since $u_f(T^\circ)$ decreases by the score of the leaf selected, and $u_f(T^\circ)$ upper bounds $\mathrm{dist}(f, T^\circ_f)$, (Blanc et al., 2020) are able to conclude that a growing a decision tree of size

$$2^{O(\sqrt{\log s} \cdot \log(s)/\varepsilon)} = s^{O(\sqrt{\log s}/\varepsilon)}$$

suffices to ensure an error of $\leq \varepsilon$. Using this same proof outline, we will establish the same guarantee for BUILDTOPDOWNDT$_{\mathscr{G}}$ for any strongly concave impurity function $\mathscr{G}$:

Theorem 6 (Extending Theorem 5 of (Blanc et al., 2020) to actual top-down heuristics)**.** *Let $\mathscr{G}$ be any κ-strongly concave impurity function, $\varepsilon \in (0, \frac{1}{2})$, and $f : \{\pm 1\}^n \to \{0,1\}$ be a monotone function computable by a size-s decision tree. Then,*

$$\textsc{TopDownError}_{\mathscr{G}}(f, s^{O(\sqrt{\log s}/\varepsilon)}/\kappa) \leq \varepsilon.$$

Proof. Recalling Proposition 2.2, for any leaf $\ell \in T^\circ$, the variable with maximum influence in f_ℓ will also be the variable that results in the maximum purity gain when split. Therefore, BUILDTOPDOWNDT$_{\mathscr{G}}$ will always choose the same variable at any leaf to split as BUILDTOPDOWNDT$_{\mathrm{Inf}}$, but may just choose a different order of leaves to split. Splitting extra leaves can only decrease the error, so once BUILDTOPDOWNDT$_{\mathscr{G}}$ has split every leaf that BUILDTOPDOWNDT$_{\mathrm{Inf}}$ would in $s^{O(\sqrt{\log s}/\varepsilon)}$ iterations, the resulting tree must have error less than ε.

We know that running BUILDTOPDOWNDT$_{\mathrm{Inf}}$ for $s^{O(\sqrt{\log s}/\varepsilon)}$ iterations is sufficient to ensure an error of at most ε. Let T° be the size-$(j+1)$ tree built after j iterations of BUILDTOPDOWNDT$_{\mathrm{Inf}}(f)$. (Blanc et al., 2020) proved that if $\mathrm{dist}(f, T^\circ_f) \geq \varepsilon$, then the leaf, $\ell^\star$, selected in the next iteration satisfies:

$$\mathrm{score}(\ell^\star) \geq \frac{\varepsilon}{(j+1)\log s}.$$

Therefore, we can substitute in $j = s^{O(\sqrt{\log s}/\varepsilon)}$ to find that if BUILDTOPDOWNDT$_{\mathrm{Inf}}$ has split all leaves with score at least

$$\frac{\varepsilon}{s^{O(\sqrt{\log s}/\varepsilon)} \cdot \log s} = \frac{1}{s^{O(\sqrt{\log s}/\varepsilon)}},$$

the tree it has built must have error ε. We will show that BUILDTOPDOWNDT$_{\mathscr{G}}$ will not take too long to split any leaf with score at least $1/s^{O(\sqrt{\log s}/\varepsilon)}$, therefore proving an upper bound on the number of iterations it needs to reach error ε. Let ℓ be any leaf with score at least $1/s^{O(\sqrt{\log s}/\varepsilon)}$ and i be the index of its most influential variable. Then,

$$\begin{aligned}
\mathscr{G}\text{-purity-gain}_f(T^\circ, \ell, x_i) &\geq 2^{|-\ell|} \cdot \frac{\kappa}{32} \cdot \mathrm{Inf}_i(f)^2 && \text{(Lemma 2.4)} \\
&\geq \frac{\kappa}{32} \cdot (2^{|-\ell|} \cdot \mathrm{Inf}_i(f))^2 \\
&= \frac{\kappa}{32} \cdot \mathrm{score}(\ell)^2 \\
&\geq \frac{\kappa}{s^{O(\sqrt{\log s}/\varepsilon)}}.
\end{aligned}$$

Let $\ell^\star$ be some leaf with larger purity gain that ℓ, and $i^\star$ be the associated variable that is split. Then, since $\mathscr{G}\text{-purity-gain}_f(T^\circ, \ell^*, x_{i^\star}) \leq 2^{-|\ell^\star|}$, we must have that $|\ell^\star| < \log(s^{O(\sqrt{\log s}/\varepsilon)}/\kappa)$. There are at most $s^{O(\sqrt{\log s}/\varepsilon)}/\kappa$ possible such nodes, so after constructing a tree of size $s^{O(\sqrt{\log s}/\varepsilon)}/\kappa$, BUILDTOPDOWNDT$_{\mathscr{G}}$ must split ℓ, and we conclude that it must have achieved error at most ε. □