Outline The supplement is organized as follows. Section A provides additional technical details. All proofs are presented in Section B. We give experimental setups, details of datasets and ML APIs, and further empirical results in Section C.

A TECHNICAL DETAILS

Computation and space cost of MASA. One attractive property of MASA is its low computation and space cost. In fact, it can be easily verified from Algorithm 1 that, the computation cost is only linear in the number of samples N. The occupied space is only constant. Therefore, MASA can be easily applied for large number of samples.

Choice of parameter a. The parameter a is used to balance between exploiting and exploration in Algorithm 1. Throughout this paper, we set $a = 1$ as the default value. While theoretically a should depend on the partition size and sample number, in practice we found that $a = 1$ works well. An in-depth analysis for this remains an interesting open problem.

Stopping rule under loss requirements. For MASA, we establish the upper bound on the loss by (i) computing the upper bound on the estimated uncertainty score for each partition, and (ii) summing up all those upper bounds weighted by the partition size to form the upper bound on the loss. For Uniform sampling or stratified sampling, we directly use the upper bound on the Frobenius loss. Here, we adopt the standard upper bound for Bernoulli variables. That is to say, for any estimator using n samples, we use $\sqrt{\frac{c}{n}}$ as its upper bound, where c is a parameter to control the confidence. For both methods, we choose c to ensure a 1% error under 95% confidence level.

B PROOFS

We present all missing proofs here. For ease of expositions, let us first introduce a few notations. We let x_n denote the nth sample drawn in Algorithm 1, and use I_n to indicate from which partition the sample x_n is drawn. For example, $I_n = (i, k)$ indicates that x_n is drawn from the partition $D_{i,k}$.

Let $\boldsymbol{z}_{\ell,k,t} \in [L]$ denote the ML API's predicted label for the tth sample drawn from the partition $D_{\ell,k}$. Abusing the notation a little bit, let $\boldsymbol{N}_{\ell,k,n}$ denote the value of $\boldsymbol{N}_{\ell,k}$ after the $n-1$th iteration and before the nth iteration in Algorithm 1. Similarly, let $\hat{\boldsymbol{\sigma}}_{\ell,k,n}$ be the value of $\hat{\boldsymbol{\sigma}}_{\ell,k}$, $\hat{\boldsymbol{\mu}}_{\ell,k,j,n}$ be the value of $\hat{\boldsymbol{\mu}}_{\ell,k,j}$, and $\boldsymbol{H}_{\ell,k,,j,n}$ be the value of $\boldsymbol{H}_{\ell,k,j}$, all after the $n-1$th iteration and before the nth iteration in Algorithm 1. In addition, let $\Delta_{\ell,k} \triangleq \frac{\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k}}{\sum_{\ell',k'}\boldsymbol{p}_{\ell',k'}\boldsymbol{\sigma}_{\ell',k'}}$ and $\Delta_{\min} = \min \Delta_{\ell,k}$. Similarly, let us denote $\boldsymbol{\sigma}_{\min} \triangleq \min \boldsymbol{\sigma}_{\ell,k}$. By assumption that $\boldsymbol{p}_{\ell,k} > 0$ and $\boldsymbol{\sigma}_{\ell,k} > 0$, we must have $\Delta_{\min} > 0$ and $\boldsymbol{\sigma}_{\min} > 0$.

B.1 USEFUL LEMMAS

Let us first give a few useful lemmas. The first gives a high probability bound on our estimated uncertainty score.

Lemma 3. *Let the event A be*

$$A \triangleq \bigcap_{\substack{1\le k\le K, 1\le \ell\le L\\ 1\le t\le N}} \left\{ \left| \sqrt{1 - \frac{1}{t(t-1)} \sum_{i=1}^{t} \sum_{j=1, j\neq i}^{t} \mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}} - \boldsymbol{\sigma}_{\ell,k} \right| \le \sqrt[4]{\frac{\log 2/\delta}{2t}} \right\}$$

Then for any $\delta > 0$, we have $\Pr[A] \ge 1 - LKN\delta$.

Proof. For any fixed t, let us first denote

$$f(\boldsymbol{z}_{\ell,k,1}, \boldsymbol{z}_{\ell,k,2}, \cdots, \boldsymbol{z}_{\ell,k,t}) \triangleq 1 - \frac{1}{t(t-1)} \sum_{i=1}^{t} \sum_{j=1, j\neq i}^{t} \mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}$$

Its expectation is simply

$$\begin{aligned}\mathbb{E}[f(\boldsymbol{z}_{\ell,k,1},\boldsymbol{z}_{\ell,k,2},\cdots,\boldsymbol{z}_{\ell,k,t})] =&\mathbb{E}[1-\frac{1}{t(t-1)}\sum_{i=1}^{t}\sum_{j=1,j\neq i}^{t}\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}]\\ =&1-\frac{1}{t(t-1)}\mathbb{E}[\sum_{i=1}^{t}\sum_{j=1,j\neq i}^{t}\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}]\\ =&1-\mathbb{E}[\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}]\end{aligned}$$

where the second equation applies the linearity of expectation, and the third equation uses the fact that all $\boldsymbol{z}_{\ell,k,i}$ are identically independent. Note that

$$\begin{aligned}&\mathbb{E}[\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}]\\ =&\Pr[\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}]\\ =&\sum_{r=1}^{L}\Pr[\boldsymbol{z}_{\ell,k,i}=r]\Pr[\boldsymbol{z}_{\ell,k,j}=r]\\ =&\sum_{r=1}^{L}\Pr^2[\boldsymbol{z}_{\ell,k,i}=r]\end{aligned}$$

where the first equation uses the definition of indicator function, the second uses the fact that two sample are independent and there are only L many possible labels, and the last equation uses the fact that those samples' distribution is identical. Applying this in the above equation, we get

$$\begin{aligned}\mathbb{E}[f(\boldsymbol{z}_{\ell,k,1},\boldsymbol{z}_{\ell,k,2},\cdots,\boldsymbol{z}_{\ell,k,t})] =&1-\mathbb{E}[\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}]\\ =&1-\sum_{r=1}^{L}\Pr^2[\boldsymbol{z}_{\ell,k,i}=r]=\boldsymbol{\sigma}^2_{\ell,k}\end{aligned}$$

That is to say, its expectation is simply the uncertainty score $\boldsymbol{\sigma}^2_{\ell,k}$. On the other hand, we note that, for any i, we have

$$\begin{aligned}&f(\boldsymbol{z}_{\ell,k,1},\cdots,\boldsymbol{z}_{\ell,k,i-1},\boldsymbol{z}_{\ell,k,i},\boldsymbol{z}_{\ell,k,i+1},\cdots,\boldsymbol{z}_{\ell,k,t})-f(\boldsymbol{z}_{\ell,k,1},\cdots,\boldsymbol{z}_{\ell,k,i-1},\boldsymbol{z}'_{\ell,k,i},\boldsymbol{z}_{\ell,k,i+1},\cdots,\boldsymbol{z}_{\ell,k,t})\\ =&\frac{1}{t(t-1)}\sum_{j=1,j\neq i}^{t}\mathbb{1}_{\boldsymbol{z}'_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}-\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}\leq\frac{1}{t(t-1)}\cdot(t-1)=\frac{1}{t}\end{aligned}$$

where the inequality is due to the fact that the indicator function can only take values in $\{0,1\}$. Similarly, we have

$$\begin{aligned}&f(\boldsymbol{z}_{\ell,k,1},\cdots,\boldsymbol{z}_{\ell,k,i-1},\boldsymbol{z}_{\ell,k,i},\boldsymbol{z}_{\ell,k,i+1},\cdots,\boldsymbol{z}_{\ell,k,t})-f(\boldsymbol{z}_{\ell,k,1},\cdots,\boldsymbol{z}_{\ell,k,i-1},\boldsymbol{z}'_{\ell,k,i},\boldsymbol{z}_{\ell,k,i+1},\cdots,\boldsymbol{z}_{\ell,k,t})\\ =&\frac{1}{t(t-1)}\sum_{j=1,j\neq i}^{t}\mathbb{1}_{\boldsymbol{z}'_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}-\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}\geq\frac{1}{t(t-1)}\cdot-(t-1)=-\frac{1}{t}\end{aligned}$$

By Mcdiarmid inequality, we have

$$\Pr[|f(\boldsymbol{z}_{\ell,k,1},\boldsymbol{z}_{\ell,k,2},\cdots,\boldsymbol{z}_{\ell,k,t})-\mathbb{E}[f(\boldsymbol{z}_{\ell,k,1},\boldsymbol{z}_{\ell,k,2},\cdots,\boldsymbol{z}_{\ell,k,t})]|\geq\epsilon]\leq 2e^{-\frac{2\epsilon^2}{\sum_{i=1}^{t}t^{-2}}}=2e^{-2t\epsilon^2}$$

Set $\delta=2e^{-2t\epsilon^2}$. This simply becomes, with probability at most δ,

$$\begin{aligned}&|f(\boldsymbol{z}_{\ell,k,1},\boldsymbol{z}_{\ell,k,2},\cdots,\boldsymbol{z}_{\ell,k,t})-\boldsymbol{\sigma}^2_{\ell,k}|\\ =&|f(\boldsymbol{z}_{\ell,k,1},\boldsymbol{z}_{\ell,k,2},\cdots,\boldsymbol{z}_{\ell,k,t})-\mathbb{E}[f(\boldsymbol{z}_{\ell,k,1},\boldsymbol{z}_{\ell,k,2},\cdots,\boldsymbol{z}_{\ell,k,t})]|\geq\sqrt{\frac{\log 2/\delta}{2t}}\end{aligned}$$

Note that f is positive, we can take square root of both side, and obtain with probability at most δ,

$$|\sqrt{f(\boldsymbol{z}_{\ell,k,1},\boldsymbol{z}_{\ell,k,2},\cdots,\boldsymbol{z}_{\ell,k,t})}-\boldsymbol{\sigma}_{\ell,k}|\geq\sqrt[4]{\frac{\log 2/\delta}{2t}}$$

Or alternatively, with probability at least $1-\delta$,

$$|\sqrt{f(\boldsymbol{z}_{\ell,k,1}, \boldsymbol{z}_{\ell,k,2}, \cdots, \boldsymbol{z}_{\ell,k,t})} - \boldsymbol{\sigma}_{\ell,k}| \leq \sqrt[4]{\frac{\log 2/\delta}{2t}}$$

which holds for fixed t, ℓ, k. Taking union bound, we know that with probability $1 - KLN\delta$,

$$|\sqrt{f(\boldsymbol{z}_{\ell,k,1}, \boldsymbol{z}_{\ell,k,2}, \cdots, \boldsymbol{z}_{\ell,k,t})} - \boldsymbol{\sigma}_{\ell,k}| \leq \sqrt[4]{\frac{\log 2/\delta}{2t}}$$

which holds for all t, ℓ, k. Plugging in the form of f completes the proof. □

The next one is more technical: it gives a connection between stopping time and adaptive sampling. We omit the proof and refer the interested readers to (Athreya & Lahiri, 2006).

Lemma 4 (Wald's second inequality)**.** *Let $\{\mathcal{F}_t\}_{t=1,\ldots,n}$ be a filtration and $\{X_t\}_{t=1,\ldots,n}$ be an $\mathcal{F}_t$ adapted sequence of i.i.d. random variables with finite expectation μ and variance Var. Assume that $\mathcal{F}_t$ and $\sigma(\{X_s : s \geq t+1\})$ are independent for any $t \leq n$, and let $T(\leq n)$ be a stopping time with respect to $\mathcal{F}_t$. Then*

$$\mathbb{E}\left[\Big(\sum_{i=1}^{T} X_i - T\ \mu\Big)^2\right] = \mathbb{E}[T]\ \ \mathrm{Var}\,.$$

B.2 Proof of Lemma 1

Proof. Recall that the loss, defined as the expected squared Frobenius norm error, is

$$\begin{aligned}\mathbb{E}\left[\|\Delta \boldsymbol{C} - \Delta\hat{\boldsymbol{C}}\|_F^2\right] &= \sum_{i,j} \mathbb{E}\left(\Delta \boldsymbol{C}_{i,j} - \Delta\hat{\boldsymbol{C}}_{i,j}\right)^2 = \sum_{i,j} \mathbb{E}\left(\sum_k \boldsymbol{p}_{i,k}[\boldsymbol{\mu}_{i,k,j} - \hat{\boldsymbol{\mu}}_{i,k,j}]\right)^2 \\ &= \sum_{i,j,k} \boldsymbol{p}_{i,k}^2 \mathbb{E}\left([\boldsymbol{\mu}_{i,k,j} - \hat{\boldsymbol{\mu}}_{i,k,j}]\right)^2\end{aligned}$$

Here we basically apply the definition of each entry. Suppose $\boldsymbol{N}_{i,k}$ samples are allocated to estimate $\boldsymbol{\mu}_{i,k,j}$. Then we have

$$\mathbb{E}\left([\boldsymbol{\mu}_{i,k,j} - \hat{\boldsymbol{\mu}}_{i,k,j}]\right)^2 = \frac{1}{\boldsymbol{N}_{i,k}} \Pr[\hat{y}(x) = j | x \in D_{i,k}](1 - \Pr[\hat{y}(x) = j | x \in D_{i,k}])$$

since $\boldsymbol{\mu}_{i,k,j}$ is effectively a Bernoulli variable. Then the loss becomes

$$\begin{aligned}\mathbb{E}\left[\|\Delta \boldsymbol{C} - \Delta\hat{\boldsymbol{C}}\|_F^2\right] &= \sum_{i,j,k} \boldsymbol{p}_{i,k}^2 \mathbb{E}\left([\boldsymbol{\mu}_{i,k,j} - \hat{\boldsymbol{\mu}}_{i,k,j}]\right)^2 \\ &= \sum_{i,j,k} \boldsymbol{p}_{i,k}^2 \frac{1}{\boldsymbol{N}_{i,k}} \Pr[\hat{y}(x) = j | x \in D_{i,k}](1 - \Pr[\hat{y}(x) = j | x \in D_{i,k}]) \\ &= \sum_{i,k} \boldsymbol{p}_{i,k}^2 \frac{1}{\boldsymbol{N}_{i,k}} \sum_j \Pr[\hat{y}(x) = j | x \in D_{i,k}](1 - \Pr[\hat{y}(x) = j | x \in D_{i,k}])\end{aligned}$$

where the last equation is simply by rearranging the summation. Note that

$$\sum_j \Pr[\hat{y}(x) = j | x \in D_{i,k}] = 1$$

The last summation is simply

$$\begin{aligned}\sum_j \Pr[\hat{y}(x) = j | x \in D_{i,k}](1 - \Pr[\hat{y}(x) = & j | x \in D_{i,k}]) = 1 - \sum_j \Pr[\hat{y}(x) = j | x \in D_{i,k}]^2) \\ &= \boldsymbol{\sigma}_{i,k}^2\end{aligned}$$

Thus, the loss becomes

$$\mathbb{E}\left[\|\Delta \boldsymbol{C} - \Delta \hat{\boldsymbol{C}}\|_F^2\right] = \sum_{i,k} \boldsymbol{p}_{i,k}^2 \boldsymbol{\sigma}_{i,k}^2 \frac{1}{\boldsymbol{N}_{i,k}}$$

By Cauchy Schwarz inequality, we have

$$\left(\sum_{i,k} \frac{\boldsymbol{p}_{i,k}^2 \boldsymbol{\sigma}_{i,k}^2}{\boldsymbol{N}_{i,k}}\right)\left(\sum_{i,k} \boldsymbol{N}_{i,k}\right) \geq \left(\sum_{i,k} \boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}\right)^2$$

where the equality holds if and only if

$$\frac{\boldsymbol{p}_{i,k}^2 \boldsymbol{\sigma}_{i,k}^2}{\boldsymbol{N}_{i,k}^2} = \frac{\boldsymbol{p}_{i',k'}^2 \boldsymbol{\sigma}_{i',k'}^2}{\boldsymbol{N}_{i',k'}^2}$$

for any i, i', k, k'. That is to say, there exists some constant c, such that

$$\frac{\boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}}{\boldsymbol{N}_{i,k}} = \frac{\boldsymbol{p}_{i',k'}\boldsymbol{\sigma}_{i',k'}}{\boldsymbol{N}_{i',k'}} = \frac{1}{c}$$

And thus, $\boldsymbol{N}_{i,k} = \boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}c$. Summing over i, k gives

$$N = \sum_{i,k} \boldsymbol{N}_{i,k} = \sum_{i,k} \boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}c$$

Thus,

$$c = \frac{N}{\sum_{i,k} \boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}}$$

and

$$\boldsymbol{N}_{i,k} = \boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}c = \boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k} \cdot \frac{1}{\sum_{i,k} \boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}} = \frac{\boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}}{\sum_{i,k} \boldsymbol{p}_{i,k}\boldsymbol{\sigma}_{i,k}}$$

which completes the proof. □

B.3 Proof of Theorem 2

Proof. To prove this theorem, we need a few more lemmas.

Lemma 5. *Algorithm 1's computational cost is $O(LKN)$ and space cost is $O(L^2K)$. Furthermore, for any $n > 2LK$, after the $n-1$th iteration and before the nth iteration, we have*

$$\boldsymbol{N}_{\ell,k} = \boldsymbol{N}_{\ell,k,n} = \sum_{i=1}^{n-1} \mathbb{1}_{I_i=(\ell,k)}$$

$$\hat{\boldsymbol{\mu}}_{\ell,k,j} = \hat{\boldsymbol{\mu}}_{\ell,k,j,n} = \frac{1}{\boldsymbol{N}_{\ell,k,n}} \sum_{i=1}^{n-1} \mathbb{1}_{I_i=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=j}$$

$$\hat{\boldsymbol{\sigma}}_{\ell,k} = \hat{\boldsymbol{\sigma}}_{\ell,k,n} = 1 - \frac{1}{\boldsymbol{N}_{\ell,k,n}(\boldsymbol{N}_{\ell,k,n}-1)} \sum_{i=1}^{n-1} \sum_{j=1,j\neq i}^{n-1} \mathbb{1}_{I_i=I_j=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_j)}$$

$$\boldsymbol{H}_{\ell,k,j} = \boldsymbol{H}_{\ell,k,j,n} = \sum_{i=1}^{n-1} \mathbb{1}_{I_i=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=j}$$

Proof. The computational and space cost can be easily verified: as shown in Algorithm 1, the variables $\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\mu}}, \boldsymbol{H}, \boldsymbol{N}$ take space $LK, L2K, L^2K, LK$. Therefore, the space is bounded by $O(L^2K)$. For the first $2LK$ iterations (line 3-8) in Algorithm 1, the computation cost is clearly $O(LK)$.

For the rest iterations (line 10- 16), the most expensive cost is computing I_n, which requires LK computations per iteration. Therefore, the total computational cost is $O(LKN)$.

Next we show that the above four equations hold for every $n > 2LK$. We prove this by induction.

1) $n = 2LK + 1$: One can easily verify this by plugging the initial values established in line 3-8 in Algorithm 1.

2) Suppose the four equations hold for the case when $n = m$. Now consider $n = m + 1$. Now let us consider two cases.

- Any ℓ, k such that $I_{m+1} \neq (\ell, k)$: There is nothing update,

$$\boldsymbol{N}_{\ell,k,m+1} = \boldsymbol{N}_{\ell,k,m} = \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell,k)} = \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell,k)} + 0 = \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell,k)} + \mathbb{1}_{I_m=(\ell,k)}$$
$$= \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell,k)}$$

  Similarly, one can show that

$$\hat{\boldsymbol{\mu}}_{\ell,k,j,m+1} = \hat{\boldsymbol{\mu}}_{\ell,k,j,m} = \frac{1}{\boldsymbol{N}_{\ell,k,m}} \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=j}$$

$$\hat{\boldsymbol{\sigma}}_{\ell,k,m+1} = \hat{\boldsymbol{\sigma}}_{\ell,k,m} = 1 - \frac{1}{\boldsymbol{N}_{\ell,k,m}(\boldsymbol{N}_{\ell,k,m} - 1)} \sum_{i=1}^{m} \sum_{j=1,j\neq i}^{m} \mathbb{1}_{I_i=I_j=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_j)}$$

$$\boldsymbol{H}_{\ell,k,j,m+1} = \boldsymbol{H}_{\ell,k,j,m} = \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=j}$$

- For some ℓ^*, k^* such that $I_{m+1} = (\ell^*, k^*)$.

  Let us first consider $\boldsymbol{N}_{\ell^*,k^*}$. We increment $\boldsymbol{N}_{\ell^*,k^*}$ by one, and thus

$$\boldsymbol{N}_{\ell^*,k^*,m+1} = \boldsymbol{N}_{\ell^*,k^*,m} + 1 = \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell^*,{}^*k)} + 1 = \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell^*,k^*)} + \mathbb{1}_{I_m=(\ell^*,k^*)}$$
$$= \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell^*,k^*)}$$

  Next we consider $\hat{\boldsymbol{\mu}}_{\ell^*,k^*,j}$. Using a similar argument as above, we have

$$\begin{aligned}
\hat{\boldsymbol{\mu}}_{\ell^*,k^*,j,m+1} =& \hat{\boldsymbol{\mu}}_{\ell^*,k^*,j,m} + \frac{\mathbb{1}_{\hat{y}(x_m)=j} - \hat{\boldsymbol{\mu}}_{\ell^*,k^*,j,m}}{\boldsymbol{N}_{\ell^*,k^*,m+1}}, \\
=& \frac{\boldsymbol{N}_{\ell^*,k^*,m+1} - 1}{\boldsymbol{N}_{\ell^*,k^*,m+1}} \hat{\boldsymbol{\mu}}_{\ell^*,k^*,j,m} + \frac{\mathbb{1}_{\hat{y}(x_m)=j}}{\boldsymbol{N}_{\ell^*,k^*,m+1}}, \\
=& \frac{\boldsymbol{N}_{\ell^*,k^*,m}}{\boldsymbol{N}_{\ell^*,k^*,m+1}} \hat{\boldsymbol{\mu}}_{\ell^*,k^*,j,m} + \frac{\mathbb{1}_{\hat{y}(x_m)=j}}{\boldsymbol{N}_{\ell^*,k^*,m+1}}, \\
=& \frac{\boldsymbol{N}_{\ell^*,k^*,m}}{\boldsymbol{N}_{\ell^*,k^*,m+1}} \frac{1}{\boldsymbol{N}_{\ell^*,k^*,m}} \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_i)=j} + \frac{\mathbb{1}_{\hat{y}(x_m)=j}}{\boldsymbol{N}_{\ell^*,k^*,m+1}} \\
=& \frac{1}{\boldsymbol{N}_{\ell^*,k^*,m+1}} \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_i)=j}
\end{aligned}$$

where the first equation is due to the update rule of $\hat{\boldsymbol{\mu}}_{\ell^*,k^*,j}$, the second equation is simply grouping by $\hat{\boldsymbol{\mu}}_{\ell^*,k^*,j,m}$, the third equation is due to the fact that $\boldsymbol{N}_{\ell^*,k^*,m+1} = \boldsymbol{N}_{\ell^*,k^*,m}+1$, the forth equation is due to the induction assumption, and the forth equation is simply algebraic rewriting.

Now let us consider $\hat{\boldsymbol{\sigma}}^2_{\ell^*,k^*,m+1}$. We can write

$$
\begin{aligned}
&\hat{\boldsymbol{\sigma}}^2_{\ell^*,k^*,m+1} \\
=&\hat{\boldsymbol{\sigma}}^2_{\ell^*,k^*,m} + \frac{2}{\boldsymbol{N}_{\ell^*,k^*,m+1}}(1 - \frac{\boldsymbol{H}_{\ell^*,k^*,\hat{y}(x_m),m}}{\boldsymbol{N}_{\ell^*,k^*,m+1}-1} - \hat{\boldsymbol{\sigma}}^2_{\ell^*,k^*,m}) \\
=&\frac{\boldsymbol{N}_{\ell^*,k^*,m+1}-2}{\boldsymbol{N}_{\ell^*,k^*,m+1}}\hat{\boldsymbol{\sigma}}^2_{\ell^*,k^*,m} + \frac{2}{\boldsymbol{N}_{\ell^*,k^*,m+1}}(1 - \frac{\boldsymbol{H}_{\ell^*,k^*,\hat{y}(x_m),m}}{\boldsymbol{N}_{\ell^*,k^*,m+1}-1}) \\
=&\frac{\boldsymbol{N}_{\ell^*,k^*,m+1}-2}{\boldsymbol{N}_{\ell^*,k^*,m+1}}(1 - \frac{1}{\boldsymbol{N}_{\ell^*,k^*,m}(\boldsymbol{N}_{\ell^*,k^*,m}-1)}\sum_{i=1}^{m-1}\sum_{j=1,j\neq i}^{m-1}\mathbb{1}_{I_i=I_j=(\ell^*,k^*)}\mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_j)}) \\
&+\frac{2}{\boldsymbol{N}_{\ell^*,k^*,m+1}}(1 - \frac{\boldsymbol{H}_{\ell^*,k^*,\hat{y}(x_m),m}}{\boldsymbol{N}_{\ell^*,k^*,m+1}-1}) \\
=&\frac{\boldsymbol{N}_{\ell^*,k^*,m+1}-2}{\boldsymbol{N}_{\ell^*,k^*,m+1}}(1 - \frac{1}{(\boldsymbol{N}_{\ell^*,k^*,m+1}-2)(\boldsymbol{N}_{\ell^*,k^*,m+1}-1)}\sum_{i=1}^{m-1}\sum_{j=1,j\neq i}^{m-1}\mathbb{1}_{I_i=I_j=(\ell^*,k^*)}\mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_j)}) \\
&+\frac{2}{\boldsymbol{N}_{\ell^*,k^*,m+1}}(1 - \frac{\boldsymbol{H}_{\ell^*,k^*,\hat{y}(x_m),m}}{\boldsymbol{N}_{\ell^*,k^*,m+1}-1}) \\
=&1 - \frac{1}{\boldsymbol{N}_{\ell^*,k^*,m+1}(\boldsymbol{N}_{\ell^*,k^*,m+1}-1)}\sum_{i=1}^{m-1}\sum_{j=1,j\neq i}^{m-1}\mathbb{1}_{I_i=I_j=(\ell^*,k^*)}\mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_j)} \\
&-\frac{2\boldsymbol{H}_{\ell^*,k^*,\hat{y}(x_m),m}}{\boldsymbol{N}_{\ell^*,k^*,m+1}(\boldsymbol{N}_{\ell^*,k^*,m+1}-1)}
\end{aligned}
$$

where the first equation is by the update rule in Algorithm 1, the second equation is simply rearranging the terms, the third equation uses the induction assumption, the forth one uses the update rule on $\boldsymbol{N}_{\ell^*,k^*}$ and thus $\boldsymbol{N}_{\ell^*,k^*,m} = \boldsymbol{N}_{\ell^*,k^*,m+1} - 1$, and the fifth equation is also rearranging the terms.

On the other hand, by induction assumption, we have

$$
\boldsymbol{H}_{\ell^*,k^*,\hat{y}(x_m),m} = \sum_{i=1}^{m-1}\mathbb{1}_{I_i=(\ell^*,k^*)}\mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_m)}
$$

And thus

$$
\begin{aligned}
&\sum_{i=1}^{m-1}\sum_{j=1,j\neq i}^{m-1}\mathbb{1}_{I_i=I_j=(\ell^*,k^*)}\mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_j)} + 2\boldsymbol{H}_{\ell^*,k^*,\hat{y}(x_m),m} \\
=&\sum_{i=1}^{m}\sum_{j=1,j\neq i}^{m}\mathbb{1}_{I_i=(\ell^*,k^*)}\mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_m)}
\end{aligned}
$$

Hence, the above equation becomes

$$
\hat{\boldsymbol{\sigma}}^2_{\ell^*,k^*,m+1} = 1 - \frac{1}{\boldsymbol{N}_{\ell^*,k^*,m+1}(\boldsymbol{N}_{\ell^*,k^*,m+1}-1)}\sum_{i=1}^{m}\sum_{j=1,j\neq i}^{m}\mathbb{1}_{I_i=I_j=(\ell^*,k^*)}\mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_j)}
$$

Finally, let us consider $\boldsymbol{H}_{\ell^*,k^*,j}$. If $j \neq \hat{y}(x_m)$, it is clear that

$$\begin{aligned}\boldsymbol{H}_{\ell^*,k^*,j,m+1} =& \boldsymbol{H}_{\ell^*,k^*,j,m} = \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_i)=j} + 0 \\ =& \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_i)=j} + \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_m)=j} \\ =& \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_i)=j}\end{aligned}$$

where the first is due to that there is no update for this j, the third equation is due to the fact that $\hat{y}(x_m) \neq j$, and all the other equations are algebraic rewriting.

If $j = \hat{y}(x_m)$, it is clear that

$$\begin{aligned}\boldsymbol{H}_{\ell^*,k^*,j,m+1} =& \boldsymbol{H}_{\ell^*,k^*,j,m} + 1 = \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_i)=j} + 1 \\ =& \sum_{i=1}^{m-1} \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_i)=j} + \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_m)=j} \\ =& \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell^*,k^*)} \mathbb{1}_{\hat{y}(x_i)=j}\end{aligned}$$

where the first is due to that there is no update for this j, the third equation is due to the fact that $\hat{y}(x_m) = j$, and all the other equations are algebraic rewriting.

That is to say, we have shown that,

$$\boldsymbol{N}_{\ell,k,m+1} = \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell,k)}$$

$$\hat{\boldsymbol{\mu}}_{\ell,k,j,m+1} = \frac{1}{\boldsymbol{N}_{\ell,k,m+1}} \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=j}$$

$$\hat{\boldsymbol{\sigma}}_{\ell,k,m+1} = 1 - \frac{1}{\boldsymbol{N}_{\ell,k,m+1}(\boldsymbol{N}_{\ell,k,m+1}-1)} \sum_{i=1}^{m} \sum_{j=1,j\neq i}^{m} \mathbb{1}_{I_i=I_j=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=\hat{y}(x_j)}$$

$$\boldsymbol{H}_{\ell,k,j,m+1} = \sum_{i=1}^{m} \mathbb{1}_{I_i=(\ell,k)} \mathbb{1}_{\hat{y}(x_i)=j}$$

always hold. By induction, we can say that for any $n > 2LK$, the original equations hold, which completes the proof. □

Lemma 6. *Suppose that the event A holds. Set $\delta = 2e^{-a}$. Then for each ℓ, k, we have*

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{1}{\boldsymbol{N}^*_{\ell,k}} \left[1 + 4LKN^{-1} + \frac{4}{\boldsymbol{\sigma}_{\min}} \sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}} N^{-\frac{1}{4}} \right]$$

for any $1 \leq \ell \leq L, 1 \leq k \leq K$.

Proof. To show this, let us first establish the following useful lemma.

Lemma 7. *Suppose that the event A holds. If Algorithm 1 draws at least one sample from D_{ℓ_0,k_0} after the first $2LK$ iterations, we must have, for every ℓ, k,*

$$\boldsymbol{N}_{\ell,k} \geq (\boldsymbol{N}_{\ell_0,k_0} - 1) \boldsymbol{\sigma}_{\ell,k} \frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{p}_{\ell_0,k_0}} \left(\boldsymbol{\sigma}_{\ell_0,k_0} + 2\sqrt[4]{\frac{\log 2/\delta}{2(\boldsymbol{N}_{\ell_0,k_0} - \text{`1})}} \right)^{-1}$$

Proof. Since the event A holds, we have

$$\left\{\left|\sqrt{1-\frac{1}{t(t-1)}\sum_{i=1}^{t}\sum_{j=1,j\neq i}^{t}\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}}-\boldsymbol{\sigma}_{\ell,k}\right|\leq\sqrt[4]{\frac{\log 2/\delta}{2t}}\right\}$$

for every ℓ, k, t. Since this holds for every fixed t, it should also holds for any random variable t. Specifically, we must have

$$\left|\sqrt{1-\frac{1}{\boldsymbol{N}_{\ell,k,n}(\boldsymbol{N}_{\ell,k,n}-1)}\sum_{i=1}^{\boldsymbol{N}_{\ell,k,n}}\sum_{j=1,j\neq i}^{\boldsymbol{N}_{\ell,k,n}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}}-\boldsymbol{\sigma}_{\ell,k}\right|\leq\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}$$

Note that, by definition,

$$\hat{\boldsymbol{\sigma}}_{\ell,k,n}=\sqrt{1-\frac{1}{\boldsymbol{N}_{\ell,k,n}(\boldsymbol{N}_{\ell,k,n}-1)}\sum_{i=1}^{\boldsymbol{N}_{\ell,k,n}}\sum_{j=1,j\neq i}^{\boldsymbol{N}_{\ell,k,n}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,i}=\boldsymbol{z}_{\ell,k,j}}}$$

We can then rewrite the above inequality as

$$|\hat{\boldsymbol{\sigma}}_{\ell,k,n}-\boldsymbol{\sigma}_{\ell,k}|\leq\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}$$

That is to say,

$$\boldsymbol{\sigma}_{\ell,k}-\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}\leq\hat{\boldsymbol{\sigma}}_{\ell,k,n}\leq\boldsymbol{\sigma}_{\ell,k}+\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}$$

Adding $\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}$ to both sides, this becomes

$$\boldsymbol{\sigma}_{\ell,k}\leq\hat{\boldsymbol{\sigma}}_{\ell,k,n}+\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}\leq\boldsymbol{\sigma}_{\ell,k}+2\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}$$

Multiplying both sides by $\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n}}$, we have

$$\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n}}\boldsymbol{\sigma}_{\ell,k}\leq\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n}}\left(\hat{\boldsymbol{\sigma}}_{\ell,k,n}+\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}\right)\leq\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n}}\left(\boldsymbol{\sigma}_{\ell,k}+2\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n}}}\right)\tag{B.1}$$

which holds for any ℓ, k, n. Note that $N > 2LK$, there must exist some ℓ_0, k_0, such that Algorithm 1 draws a sample from the data partition D_{ℓ_0,k_0} after the first $2LK$ iterations. Suppose the last time a sample is drawn from D_{ℓ_0,k_0} is $n_0 > 2LK$. That is to say, $\boldsymbol{N}_{\ell_0,k_0,n_0} = \boldsymbol{N}_{\ell_0,k_0,n} - 1, \forall n = n_0 + 1, \cdots, N$. Since Algorithm 1 chooses ℓ_0, k_0 at iteration n_0, by line 11 in Algorithm 1, we have

$$\ell_0, k_0 = \arg\max \frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n_0}}(\hat{\boldsymbol{\sigma}}_{\ell,k,n_0}+\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n_0}}})$$

By definition of $\arg\max$, we have

$$\frac{\boldsymbol{p}_{\ell_0,k_0}}{\boldsymbol{N}_{\ell_0,k_0,n_0}}(\hat{\boldsymbol{\sigma}}_{\ell_0,k_0,n_0}+\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell_0,k_0,n_0}}})\geq\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n_0}}(\hat{\boldsymbol{\sigma}}_{\ell,k,n_0}+\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n_0}}})$$

Setting $n = n_0$ in the first half of inequality B.1, we have

$$\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n_0}}\left(\hat{\boldsymbol{\sigma}}_{\ell,k,n_0}+\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell,k,n_0}}}\right)\geq\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n_0}}\boldsymbol{\sigma}_{\ell,k,n_0}$$

Combining the above two inequalities gives

$$\frac{\boldsymbol{p}_{\ell_0,k_0}}{\boldsymbol{N}_{\ell_0,k_0,n_0}}(\hat{\boldsymbol{\sigma}}_{\ell_0,k_0,n_0}+\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell_0,k_0,n_0}}})\geq\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k,n_0}}\boldsymbol{\sigma}_{\ell,k}$$

Noting that by definition, $\boldsymbol{N}_{\ell,k,n_0} \leq \boldsymbol{N}_{\ell,k,N} = \boldsymbol{N}_{\ell,k}$, we can lower bound $1/\boldsymbol{N}_{\ell,k,n_0}$ by $1/\boldsymbol{N}_{\ell,k}$, and the above inequality becomes

$$\frac{\boldsymbol{p}_{\ell_0,k_0}}{\boldsymbol{N}_{\ell_0,k_0,n_0}}(\hat{\boldsymbol{\sigma}}_{\ell_0,k_0,n_0} + \sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell_0,k_0,n_0}}}) \geq \frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k}}\boldsymbol{\sigma}_{\ell,k}$$

Now setting $n = n_0, \ell = \ell_0, k = k_0$ in the second half of inequality B.1, we have

$$\frac{\boldsymbol{p}_{\ell_0,k_0}}{\boldsymbol{N}_{\ell_0,k_0,n}}\left(\hat{\boldsymbol{\sigma}}_{\ell_0,k_0,n} + \sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell_0,k_0,n_0}}}\right) \leq \frac{\boldsymbol{p}_{\ell_0,k_0}}{\boldsymbol{N}_{\ell_0,k_0,n_0}}\left(\boldsymbol{\sigma}_{\ell_0,k_0} + 2\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell_0,k_0,n_0}}}\right)$$

Combining the above two inequalities, we have

$$\frac{\boldsymbol{p}_{\ell_0,k_0}}{\boldsymbol{N}_{\ell_0,k_0,n_0}}(\boldsymbol{\sigma}_{\ell_0,k_0} + 2\sqrt[4]{\frac{\log 2/\delta}{2\boldsymbol{N}_{\ell_0,k_0,n_0}}}) \geq \frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k}}\boldsymbol{\sigma}_{\ell,k}$$

Observe that n_0 is the last time a sample is drawn from partition D_{ℓ_0,k_0}, we have $\boldsymbol{N}_{\ell_0,k_0,n_0} = \boldsymbol{N}_{\ell_0,k_0,n} - 1, \forall n = n_0+1, \cdots, N$. Specifically, $\boldsymbol{N}_{\ell_0,k_0,n_0} = \boldsymbol{N}_{\ell_0,k_0,N} - 1 = \boldsymbol{N}_{\ell_0,k_0} - 1$. Replacing $\boldsymbol{N}_{\ell_0,k_0,n_0}$ by $\boldsymbol{N}_{\ell_0,k_0} - 1$ in the above inequality, we get

$$\frac{\boldsymbol{p}_{\ell_0,k_0}}{\boldsymbol{N}_{\ell_0,k_0} - 1}(\boldsymbol{\sigma}_{\ell_0,k_0} + 2\sqrt[4]{\frac{\log 2/\delta}{2(\boldsymbol{N}_{\ell_0,k_0} - \text{`1})}}) \geq \frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{N}_{\ell,k}}\boldsymbol{\sigma}_{\ell,k}$$

which holds for every ℓ, k. Rearranging the terms completes the proof. □

Now we are ready to prove the bound on $\boldsymbol{N}_{\ell,k} - \boldsymbol{N}^*_{\ell,k}$.

Let us first consider the lower bound. By definition, we have

$$\sum_{\ell=1}^{L}\sum_{k=1}^{K}\boldsymbol{N}_{\ell,k} = N$$

Subtracting 2 from each element, we have

$$\sum_{\ell=1}^{L}\sum_{k=1}^{K}(\boldsymbol{N}_{\ell,k} - 2) = N - 2LK = \frac{N-2LK}{N}N$$

Note that by definition, $N = \sum_{\ell=1}^{L}\sum_{k=1}^{K}\boldsymbol{N}^*_{\ell,k}$. We can now replace the second N in the above equality, and obtain

$$\sum_{\ell=1}^{L}\sum_{k=1}^{K}(\boldsymbol{N}_{\ell,k} - 2) = \frac{N-2LK}{N}N = \frac{N-2LK}{N}\sum_{\ell=1}^{L}\sum_{k=1}^{K}\boldsymbol{N}^*_{\ell,k} = \sum_{\ell=1}^{L}\sum_{k=1}^{K}\frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N}$$

Now let us consider two cases.

(i) Assume $\boldsymbol{N}_{\ell,k} - 2 \geq \frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N}$. That is to say, $\boldsymbol{N}_{\ell,k} \geq \frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N} + 2$. Then we have

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{1}{\frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N} + 2}$$

subtracting $\frac{1}{\boldsymbol{N}^*_{\ell,k}}$ from both sides, we get

$$\begin{aligned}
\frac{1}{\boldsymbol{N}_{\ell,k}} - \frac{1}{\boldsymbol{N}^*_{\ell,k}} &\leq \frac{1}{\frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N} + 2} - \frac{1}{\boldsymbol{N}^*_{\ell,k}} \\
&= \frac{\boldsymbol{N}^*_{\ell,k} - \frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N} - 2}{\boldsymbol{N}^*_{\ell,k} \cdot (\frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N} + 2)} \\
&\leq \frac{\boldsymbol{N}^*_{\ell,k} - \frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N}}{\boldsymbol{N}^*_{\ell,k} \cdot (\frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N})} \\
&= \frac{\frac{2LK\boldsymbol{N}^*_{\ell,k}}{N}}{\boldsymbol{N}^*_{\ell,k} \cdot (\frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N})} \\
&= \frac{2LK}{\boldsymbol{N}^*_{\ell,k}(N-2LK)}
\end{aligned}$$

where the last inequality is simply by removing the constant 2. Now by assumption, $N > 4LK$, we have $N - 2LK < \frac{1}{2}N$. The above inequality can be further simplified as

$$\frac{1}{\boldsymbol{N}_{\ell,k}} - \frac{1}{\boldsymbol{N}^*_{\ell,k}} \leq \frac{2LK}{\boldsymbol{N}^*_{\ell,k}(N-2LK)} \leq \frac{4LK}{\boldsymbol{N}^*_{\ell,k}N}$$

By definition, we have $\boldsymbol{N}^*_{\ell,k} = N\Delta_{\ell,k} \leq N\Delta_{\min}$. Therefore, we have

$$\frac{1}{\boldsymbol{N}_{\ell,k}} - \frac{1}{\boldsymbol{N}^*_{\ell,k}} \leq \frac{2LK}{\boldsymbol{N}^*_{\ell,k}(N-2LK)} \leq \frac{4LK}{\boldsymbol{N}^*_{\ell,k}N}$$

That is to say,

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{1}{\boldsymbol{N}^*_{\ell,k}} \left[1 + \frac{4LK}{N}\right]$$

And thus, apparently,

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{1}{\boldsymbol{N}^*_{\ell,k}} \left[1 + 4LKN^{-1} + \frac{4}{\boldsymbol{\sigma}_{\min}} \sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}} N^{-\frac{1}{4}}\right]$$

(ii) Assume $\boldsymbol{N}_{\ell,k} - 2 < \frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N}$. Then there must exists some ℓ_0, k_0 such that $\boldsymbol{N}_{\ell_0,k_0} - 2 > \frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N} > 0$. That is to say, Algorithm 1 draws at least one sample from D_{ℓ_0,k_0} after the first $2LK$ iterations. By Lemma 7, we must have

$$\boldsymbol{N}_{\ell,k} \geq (\boldsymbol{N}_{\ell_0,k_0} - 1)\boldsymbol{\sigma}_{\ell,k} \frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{p}_{\ell_0,k_0}} \left(\boldsymbol{\sigma}_{\ell_0,k_0} + 2\sqrt[4]{\frac{\log 2/\delta}{2(\boldsymbol{N}_{\ell_0,k_0} - \text{`1})}}\right)^{-1}$$

$\boldsymbol{N}_{\ell_0,k_0} - 2 > \frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}$ implies

$$\boldsymbol{N}_{\ell_0,k_0} - 1 > \boldsymbol{N}_{\ell_0,k_0-2} > \frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}$$

Therefore, we can use this lower bound on $\boldsymbol{N}_{\ell_0,k_0} - 1$ in the above inequality and obtain

$$\begin{aligned}\boldsymbol{N}_{\ell,k} \geq & \frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}\boldsymbol{\sigma}_{\ell,k}\frac{\boldsymbol{p}_{\ell,k}}{\boldsymbol{p}_{\ell_0,k_0}}\left(\boldsymbol{\sigma}_{\ell_0,k_0}+2\sqrt[4]{\frac{\log 2/\delta}{2\frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}}}\right)^{-1}\\ =&\frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}\frac{\boldsymbol{\sigma}_{\ell,k}\boldsymbol{p}_{\ell,k}}{\boldsymbol{\sigma}_{\ell_0,k_0}\boldsymbol{p}_{\ell_0,k_0}}\left(1+\frac{2}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{2\frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}}}\right)^{-1}\\ =&\frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N}\left(1+\frac{2}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{2\frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}}}\right)^{-1}\end{aligned}$$

where the first equality is by dividing $\boldsymbol{\sigma}_{\ell_0,k_0}$ at both denominator and numerator, and the second equality uses the fact that $\boldsymbol{N}^*_{\ell,k}$ is proportional to $p_{\ell,k}\boldsymbol{\sigma}_{\ell,k}$. Taking inverse of the above inequality gives

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{N}{\boldsymbol{N}^*_{\ell,k}(N-2LK)}\left(1+\frac{2}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{2\frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}}}\right)$$

Now let us simplify this inequality. Let us first expand all terms and obtain

$$\begin{aligned}\frac{1}{\boldsymbol{N}_{\ell,k}} &\leq \frac{N}{\boldsymbol{N}^*_{\ell,k}(N-2LK)}\left(1+\frac{2}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{2\frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}}}\right)\\ &= \frac{1}{\boldsymbol{N}^*_{\ell,k}}+\frac{2LK}{\boldsymbol{N}^*_{\ell,k}(N-2LK)}+\frac{N}{\boldsymbol{N}^*_{\ell,k}(N-2LK)}\cdot\frac{2}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{2\frac{\boldsymbol{N}^*_{\ell_0,k_0}(N-2LK)}{N}}}\\ &= \frac{1}{\boldsymbol{N}^*_{\ell,k}}+\frac{2LK}{\boldsymbol{N}^*_{\ell,k}(N-2LK)}+\frac{2}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\left(\frac{N}{N-2LK}\right)^5\frac{\log 2/\delta}{2\boldsymbol{N}^{*4}_{\ell,k}\boldsymbol{N}^*_{\ell_0,k_0}}}\end{aligned}$$

For the second term, by assumption, $N > 4LK$ and thus $N - 2LK > 1/2N$, we have

$$\frac{2LK}{N-2LK} \leq \frac{4LK}{N}$$

Thus the above equation becomes

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{1}{\boldsymbol{N}^*_{\ell,k}}+\frac{4LK}{\boldsymbol{N}^*_{\ell,k}N}+\frac{2}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\left(\frac{N}{N-2LK}\right)^5\frac{\log 2/\delta}{2\boldsymbol{N}^{*4}_{\ell,k}\boldsymbol{N}^*_{\ell_0,k_0}}}$$

For the third term, $N > 4LK$ also implies

$$\frac{N}{N-2LK} = 1+\frac{2LK}{N-2LK} < 1+\frac{2LK}{4LK-2LK} = 2$$

Thus the above inequality can be further simplified as

$$\begin{aligned}\frac{1}{\boldsymbol{N}_{\ell,k}} &\leq \frac{1}{\boldsymbol{N}^*_{\ell,k}}+\frac{4LK}{\boldsymbol{N}^*_{\ell,k}N}+\frac{4}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{\boldsymbol{N}^{*4}_{\ell,k}\boldsymbol{N}^*_{\ell_0,k_0}}}\\ &\leq \frac{1}{\boldsymbol{N}^*_{\ell,k}}\left[1+\frac{4LK}{N}+\frac{4}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{\boldsymbol{N}^*_{\ell_0,k_0}}}\right]\end{aligned}$$

Now by definition, $\boldsymbol{\sigma}_{\ell_0,k_0} \geq \boldsymbol{\sigma}_{\min}$, and $\boldsymbol{N}^*_{\ell_0,k_0} = N\Delta_{\ell_0,k_0} \geq N\Delta_{\min}$, we can further simplify the above inequality

$$\begin{aligned}\frac{1}{\boldsymbol{N}_{\ell,k}} &\leq \frac{1}{\boldsymbol{N}^*_{\ell,k}}\left[1+\frac{4LK}{N}+\frac{4}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{\boldsymbol{N}^*_{\ell_0,k_0}}}\right] \\ &\leq \frac{1}{\boldsymbol{N}^*_{\ell,k}}\left[1+\frac{4LK}{N}+\frac{4}{\boldsymbol{\sigma}_{\ell_0,k_0}}\sqrt[4]{\frac{\log 2/\delta}{N\Delta_{\min}}}\right] \\ &\leq \frac{1}{\boldsymbol{N}^*_{\ell,k}}\left[1+\frac{4LK}{N}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{N\Delta_{\min}}}\right]\end{aligned}$$

That is to say,

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{1}{\boldsymbol{N}^*_{\ell,k}}\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]$$

That is to say, no matter $\boldsymbol{N}_{\ell,k} - 2 < \frac{\boldsymbol{N}^*_{\ell,k}(N-2LK)}{N}$ or not, this inequality always holds, which completes the proof. □

Now we are ready to prove Theorem 2. Let us first note that the loss can be written as

$$\begin{aligned}\mathcal{L}_N &= \sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}^2_{\ell,k}\mathbb{E}[\boldsymbol{\mu}_{\ell,k,j}-\hat{\boldsymbol{\mu}}_{\ell,k,j}]^2 \\ &= \sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}^2_{\ell,k}\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\mathbb{1}_A] \\ &+ \sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}^2_{\ell,k}\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\mathbb{1}_{A^C}]\end{aligned} \tag{B.2}$$

Let us first consider the first term.

$$\begin{aligned}&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}^2_{\ell,k}\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\hat{\boldsymbol{\mu}}_{\ell,k,j})^2\mathbb{1}_A] \\ =&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}^2_{\ell,k}\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\mathbb{1}_A] \\ =&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}^2_{\ell,k}\mathbb{E}\left[\frac{1}{\boldsymbol{N}^2_{\ell,k}}\left(\boldsymbol{N}_{\ell,k}\boldsymbol{\mu}_{\ell,k,j}-\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j}\right)^2\mathbb{1}_A\right]\end{aligned} \tag{B.3}$$

where we plug in the definition of $\hat{\boldsymbol{\mu}}$. By Lemma 6, we have the upper bound on $1/\boldsymbol{N}_{\ell,k}$

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{1}{\boldsymbol{N}^*_{\ell,k}}\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]$$

Therefore, we can use this inequality to obtain

$$\begin{aligned}&\mathbb{E}\left[\frac{1}{\boldsymbol{N}^2_{\ell,k}}\left(\boldsymbol{N}_{\ell,k}\boldsymbol{\mu}_{\ell,k,j}-\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j}\right)^2\mathbb{1}_A\right] \\ \leq&[\frac{1}{\boldsymbol{N}^*_{\ell,k}}+\frac{4LK}{\Delta_{\min}}N^{-2}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta^5_{\min}}}N^{-\frac{5}{4}}]^2\mathbb{E}\left[\left(\boldsymbol{N}_{\ell,k}\boldsymbol{\mu}_{\ell,k,j}-\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j}\right)^2\mathbb{1}_A\right]\end{aligned} \tag{B.4}$$

It is not hard to see that $\boldsymbol{N}_{\ell,k}$ is a stopping time. In fact, for any ℓ, k, and any time n, a new sample is drawn purely based on estimated uncertainty score $\hat{\boldsymbol{\sigma}}$ and observed sample number $\boldsymbol{N}_{\ell,k,n-1}$ up to the current iteration, which is part of the history. As $\boldsymbol{N}_{\ell,k} < N$ is bounded, $\boldsymbol{N}_{\ell,k}$ is a stopping time. Hence, we can apply Lemma 4, and obtain

$$\begin{aligned}
&\mathbb{E}\left[\left(\boldsymbol{N}_{\ell,k}\boldsymbol{\mu}_{\ell,k,j} - \sum_{t=1}^{\boldsymbol{N}_{\ell,k}} \mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j}\right)^2 \mathbb{1}_A\right] \leq \mathbb{E}\left[\left(\boldsymbol{N}_{\ell,k}\boldsymbol{\mu}_{\ell,k,j} - \sum_{t=1}^{\boldsymbol{N}_{\ell,k}} \mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j}\right)^2\right] \\
\leq &\mathbb{E}[\boldsymbol{N}_{\ell,k}] \Pr[\boldsymbol{z}_{\ell,k,1} = j](1 - \Pr[\boldsymbol{z}_{\ell,k,1} = j])
\end{aligned}$$

where the first inequality uses the fact that square term must be non-negative, and the second inequality uses the fact that, for Bernoulli distribution with mean a, its variance is $a(1-a)$. Applying this in inequality B.4, we have

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{\boldsymbol{N}_{\ell,k}^2}\left(\boldsymbol{N}_{\ell,k}\boldsymbol{\mu}_{\ell,k,j} - \sum_{t=1}^{\boldsymbol{N}_{\ell,k}} \mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j}\right)^2 \mathbb{1}_A\right] \\
\leq &[\frac{1}{\boldsymbol{N}_{\ell,k}^*} + \frac{4LK}{\Delta_{\min}}N^{-2} + \frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}^5}}N^{-\frac{5}{4}}]^2\mathbb{E}\left[\left(\boldsymbol{N}_{\ell,k}\boldsymbol{\mu}_{\ell,k,j} - \sum_{t=1}^{\boldsymbol{N}_{\ell,k}} \mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j}\right)^2 \mathbb{1}_A\right] \\
\leq &[\frac{1}{\boldsymbol{N}_{\ell,k}^*} + \frac{4LK}{\Delta_{\min}}N^{-2} + \frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}^5}}N^{-\frac{5}{4}}]^2\mathbb{E}[\boldsymbol{N}_{\ell,k}] \Pr[\boldsymbol{z}_{\ell,k,1} = j](1 - \Pr[\boldsymbol{z}_{\ell,k,1} = j])
\end{aligned}$$

Now applying this in equality B.3, we get

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L} \boldsymbol{p}_{\ell,k}^2 \mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j} - \hat{\boldsymbol{\mu}}_{\ell,k,j})^2 \mathbb{1}_A] \\
= &\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L} \boldsymbol{p}_{\ell,k}^2 \mathbb{E}\left[\frac{1}{\boldsymbol{N}_{\ell,k}^2}\left(\boldsymbol{N}_{\ell,k}\boldsymbol{\mu}_{\ell,k,j} - \sum_{t=1}^{\boldsymbol{N}_{\ell,k}} \mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j}\right)^2 \mathbb{1}_A\right] \\
\leq &\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L} \boldsymbol{p}_{\ell,k}^2 [\frac{1}{\boldsymbol{N}_{\ell,k}^*} + \frac{4LK}{\Delta_{\min}}N^{-2} + \frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}^5}}N^{-\frac{5}{4}}]^2\mathbb{E}[\boldsymbol{N}_{\ell,k}] \Pr[\boldsymbol{z}_{\ell,k,1} = j](1 - \Pr[\boldsymbol{z}_{\ell,k,1} = j]) \\
= &\sum_{\ell=1}^{L}\sum_{k=1}^{K} \boldsymbol{p}_{\ell,k}^2 \boldsymbol{\sigma}_{\ell,k}^2 [\frac{1}{\boldsymbol{N}_{\ell,k}^*} + \frac{4LK}{\Delta_{\min}}N^{-2} + \frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}^5}}N^{-\frac{5}{4}}]^2\mathbb{E}[\boldsymbol{N}_{\ell,k}]
\end{aligned} \tag{B.5}$$

where the last equation uses the fact that $\boldsymbol{\sigma}_{\ell,k} = 1 - \sum_{j=1}^{L} \Pr^2[\boldsymbol{z}_{\ell,k,1} = j] = \sum_{j=1}^{L} \Pr[\boldsymbol{z}_{\ell,k,1} = j](1 - \Pr[\boldsymbol{z}_{\ell,k,1} = j])$. Applying the inequality $1/(1+x) \leq 1 - x$

$$\frac{1}{\boldsymbol{N}_{\ell,k}} \leq \frac{1}{\boldsymbol{N}_{\ell,k}^*}\left[1 + 4LKN^{-1} + \frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{\frac{1}{4}}\right]$$

Note that

$$\begin{aligned}
&\boldsymbol{p}_{\ell,k}^2\boldsymbol{\sigma}_{\ell,k}^2[\frac{1}{\boldsymbol{N}_{\ell,k}^*}\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]]^2\mathbb{E}[\boldsymbol{N}_{\ell,k}]\\
=&(\frac{\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k}}{\boldsymbol{N}_{\ell,k}^*})^2\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]^2\mathbb{E}[\boldsymbol{N}_{\ell,k}]\\
=&N^{-2}(\sum_{\ell',k'}\boldsymbol{p}_{\ell',k'}\boldsymbol{\sigma}_{\ell',k'})^2\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]^2\mathbb{E}[\boldsymbol{N}_{\ell,k}]
\end{aligned}$$

where the last equation is by definition of $\boldsymbol{N}_{\ell,k}$. Now applying this in inequality B.5, we have

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\hat{\boldsymbol{\mu}}_{\ell,k,j})^2\mathbb{1}_A]\\
\leq&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\boldsymbol{p}_{\ell,k}^2\boldsymbol{\sigma}_{\ell,k}^2[\frac{1}{\boldsymbol{N}_{\ell,k}^*}+\frac{4LK}{\Delta_{\min}}N^{-2}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}^5}}N^{-\frac{5}{4}}]^2\mathbb{E}[\boldsymbol{N}_{\ell,k}]\\
=&\sum_{\ell=1}^{L}\sum_{k=1}^{K}N^{-2}(\sum_{\ell',k'}\boldsymbol{p}_{\ell',k'}\boldsymbol{\sigma}_{\ell',k'})^2\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]^2\mathbb{E}[\boldsymbol{N}_{\ell,k}]\\
=&N^{-2}(\sum_{\ell',k'}\boldsymbol{p}_{\ell',k'}\boldsymbol{\sigma}_{\ell',k'})^2\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]^2\sum_{\ell=1}^{L}\sum_{k=1}^{K}\mathbb{E}[\boldsymbol{N}_{\ell,k}]\\
=&N^{-2}(\sum_{\ell',k'}\boldsymbol{p}_{\ell',k'}\boldsymbol{\sigma}_{\ell',k'})^2\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]^2N\\
=&N^{-1}(\sum_{\ell,k}\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k})^2\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]^2
\end{aligned}\tag{B.6}$$

where the second equation uses the fact that only $\mathbb{E}[\boldsymbol{N}_{\ell,k}]$ depends on ℓ,k, the third equation uses the fact that $\sum_{\ell=1}^{L}\sum_{k=1}^{K}\boldsymbol{N}_{\ell,k}=N$ and thus $\sum_{\ell=1}^{L}\sum_{k=1}^{K}\mathbb{E}[\boldsymbol{N}_{\ell,k}]=N$. Note that $\delta=L^{-1}K^{-1}N^{-\frac{5}{4}}$, we have

$$\begin{aligned}
&N^{-1}(\sum_{\ell,k}\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k})^2\left[1+4LKN^{-1}+\frac{4}{\boldsymbol{\sigma}_{\min}}\sqrt[4]{\frac{\log 2/\delta}{\Delta_{\min}}}N^{-\frac{1}{4}}\right]^2\\
=&N^{-1}(\sum_{\ell,k}\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k})^2\left[1+O(N^{-\frac{1}{4}}\log^{\frac{1}{4}}N)\right]\\
=&N^{-1}(\sum_{\ell,k}\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k})^2+O(N^{-\frac{5}{4}}\log^{\frac{1}{4}}N)
\end{aligned}$$

Applying this back to inequality B.6, we have

$$\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\hat{\boldsymbol{\mu}}_{\ell,k,j})^2\mathbb{1}_A]\leq N^{-1}(\sum_{\ell,k}\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k})^2+O(N^{-\frac{5}{4}}\log^{\frac{1}{4}}N)\tag{B.7}$$

Now consider the second term in equation B.2. As $\boldsymbol{\mu}$ and $\hat{\boldsymbol{\mu}}$ are within $\{0,1\}$, we have

$$(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\in[0,1]$$

Therefore,

$$\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\mathbb{1}_{A^C}]\leq\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\Pr[A^C]$$

By Lemma 3, the probability of A is at least $1-KLN\delta$. Hence, the probability of A^C is at most $KLN\delta$. Hence,

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\mathbb{1}_{A^C}]\\
\leq&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\Pr[A^C]\\
\leq&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2LKN\delta\\
\leq&\sum_{j=1}^{L}\boldsymbol{L}KN\delta=L^2KN\delta
\end{aligned}$$

where the last inequality uses the fact that $\sum_{\ell=1}^{L}\sum_{k=1}^{K}\boldsymbol{p}_{\ell,k}^2\leq 1$ since $\sum_{\ell=1}^{L}\sum_{k=1}^{K}\boldsymbol{p}_{\ell,k}=1$ and $\boldsymbol{p}_{\ell,k}\geq 0$. Since $\delta=L^{-2}K^{-1}N^{-\frac{9}{4}}$, we have

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\mathbb{1}_{A^C}]\\
\leq&L^2KN\delta\leq N^{-\frac{5}{4}}
\end{aligned}$$

Applying this as well as inequality B.7 to the equation B.2, we have

$$\begin{aligned}
\mathcal{L}_N=&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\mathbb{E}[\boldsymbol{\mu}_{\ell,k,j}-\hat{\boldsymbol{\mu}}_{\ell,k,j}]^2\\
=&\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\mathbb{1}_{A}]\\
&+\sum_{\ell=1}^{L}\sum_{k=1}^{K}\sum_{j=1}^{L}\boldsymbol{p}_{\ell,k}^2\mathbb{E}[(\boldsymbol{\mu}_{\ell,k,j}-\frac{1}{\boldsymbol{N}_{\ell,k}}\sum_{t=1}^{\boldsymbol{N}_{\ell,k}}\mathbb{1}_{\boldsymbol{z}_{\ell,k,t}=j})^2\mathbb{1}_{A^C}]\\
\leq&N^{-1}(\sum_{\ell,k}\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k})^2+O(N^{-\frac{5}{4}}\log^{\frac{1}{4}}N)+N^{-\frac{5}{4}}
\end{aligned}$$

Note that the loss of the optimal allocation is simply $\mathcal{L}_N^*=N^{-1}(\sum_{\ell,k}\boldsymbol{p}_{\ell,k}\boldsymbol{\sigma}_{\ell,k})^2$. The above inequality is simply

$$\mathcal{L}_N-\mathcal{L}_N^*\leq O(N^{-\frac{5}{4}}\log^{\frac{1}{4}}N)$$

which completes the proof. □

C EXPERIMENTAL DETAILS

Experimental Setups. All experiments were run on a machine with 2 E5-2690 v4 CPUs, 160 GB RAM and 500 GB disk with Ubuntu 18.04 LTS as the OS. Our code is implemented and tested in python 3.7. All experimental results were averaged over 1500 runs, except the case study. Overall the experiments took about two month, including debugging and evaluation on all datasets. Running MASA once to draw a few thousand samples typically only takes a few seconds. Our implementation is purely in Python for demonstration purposes, and more code optimization (e.g., using cython or multi-thread) can generate a much faster implementation.

Table 2: Dataset statistics.

Dataset	Size	# Classes	Dataset	Size	# Classes	Tasks
FER+	6358	7	RAFDB (Li et al.)	15339	7	*FER*
EXPW	31510	7	AFFECTNET	87401	7	
YELP	20000	2	SHOP	62774	2	*SA*
IMDB	25000	2	WAIMAI	11987	2	
DIGIT	2000	10	AUDIOMNIST	30000	10	*STT*
FLUENT	30043	31	COMMAND	64727	31	

Table 3: ML services used for each task. Price unit: USD/10,000 queries. We consider three tasks, sentiment analysis (SA), facial emotion recognition (FER), and spoken command recognition (SCR)

.

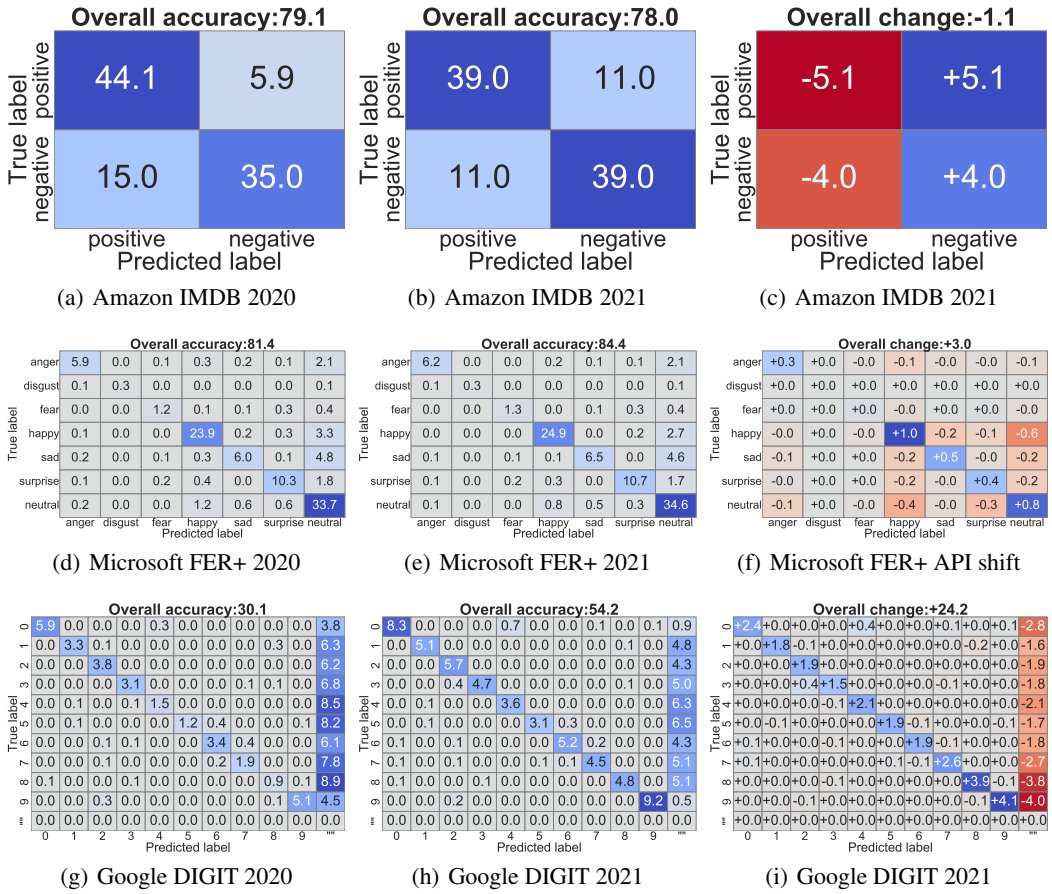
Tasks	ML service	Price	ML service	Price	ML service	Price
SA	Google NLP (GoN)	2.5	AMZN Comp (Ama)	0.75	Baidu NLP (Bai)	3.5
FER	Google Vision (Goo, a)	15	MS Face (Mic, a)	10	Face++ (Fac)	5
SCR	Google Speech (Goo, b)	60	MS Speech (Mic, b)	41	IBM Speech (IBM)	25

ML APIs and Dataset Statistics. We focus on three common classification tasks, namely, sentiment analysis, facial emotion recognition, and spoken command recognition. For each of the tasks, we evaluated three APIs' performance in spring 2020 and spring 2021, respectively, for four datasets. The details of datasets and ML APIs are summarized in Table 2 and Table 3 respectively. Now we give more context of the datasets.

For sentiment analysis, we use four datasets, YELP, IMDB, SHOP, and WAIMAI. YELP and IMDB are both English text datasets. YELP (Dat, c) is generated by drawn twenty thousand samples from the large YELP review challenge dataset. Each original review is labeled by rating in {1,2,3,4,5}. We generate the binary label by transforming rating 1 and 2 into negative, and rating 4 and 5 into positive. Ten thousand positive reviews and ten thousand negative reviews are then randomly drawn, respectively. IMDB (Maas et al.) is a polarized sentiment analysis dataset with provided training and testing partitions. We use its testing partition which has twenty-five thousand text paragraphs. SHOP (Dat, a) and WAIMAI (Dat, b) are two Chinese text datasets. SHOP contains polarized labels for reviews for various purchases including fruits, hotels, computers. WAIMAI is a dataset for polarized delivery reviews. Both SHOP and WAIMAI are publicly available without licence requirements. There is a dataset user agreement for YELP dataset, which disallows commercial usage of the datasets but encourages academic study. Same thing applies to the IMDB dataset.

For facial emotion recognition, we use four datasets: FER+, RAFDB, EXPW, and AFNET. All the datasets are annotated by the standard seven basic emotions, i.e., {anger, disgust, fear, happy, sad, surprise, neutral}. The images in FER+ (Goodfellow et al., 2015) are from the ICML 2013 Workshop on Challenges in Representation. We use the provided testing portion in FER+. RAFDB (Li et al.) and AFFECTNET (Mollahosseini et al., 2019) were annotated with both basic emotions and fine-grained labels. In this paper, we only use basic emotions since commercial APIs cannot work for compound emotions. EXPW (Zhang et al.) contains raw images and bound boxes pointing out the face locations. Here we use the true bounding box associated with the dataset to create aligned faces first, and only pick the images that are faces with confidence larger than 0.6. We cotnacted the creators of RAFDB and AFNET to obtain the data access for academic purposes. FER+ and EXPW are both publicly available online without consent or licence requirements.

For spoken command recognition, we use DIGIT, AMNIST, CMD, and FLUENT. DIGIT (Dat, d) and AMNIST (Becker et al., 2018) are spoken digit datasets, where the label is is a spoken digit (i.e., 0-9). The sampling rate is 8 kHz for DIGIT and 48 kHz for AMNIST. Each sample in CMD (Warden, 2018) is a spoken command such as "go", "left", "right", "up", and "down", with a sampling rate of 16 kHz. In total, there are 30 commands and a few white noise utterances. FLUENT (Lugosch et al.) is another recently developed dataset for speech command. The commands in FLUENT are typically



(a) Amazon IMDB 2020 (b) Amazon IMDB 2021 (c) Amazon IMDB 2021

(d) Microsoft FER+ 2020 (e) Microsoft FER+ 2021 (f) Microsoft FER+ API shift

(g) Google DIGIT 2020 (h) Google DIGIT 2021 (i) Google DIGIT 2021

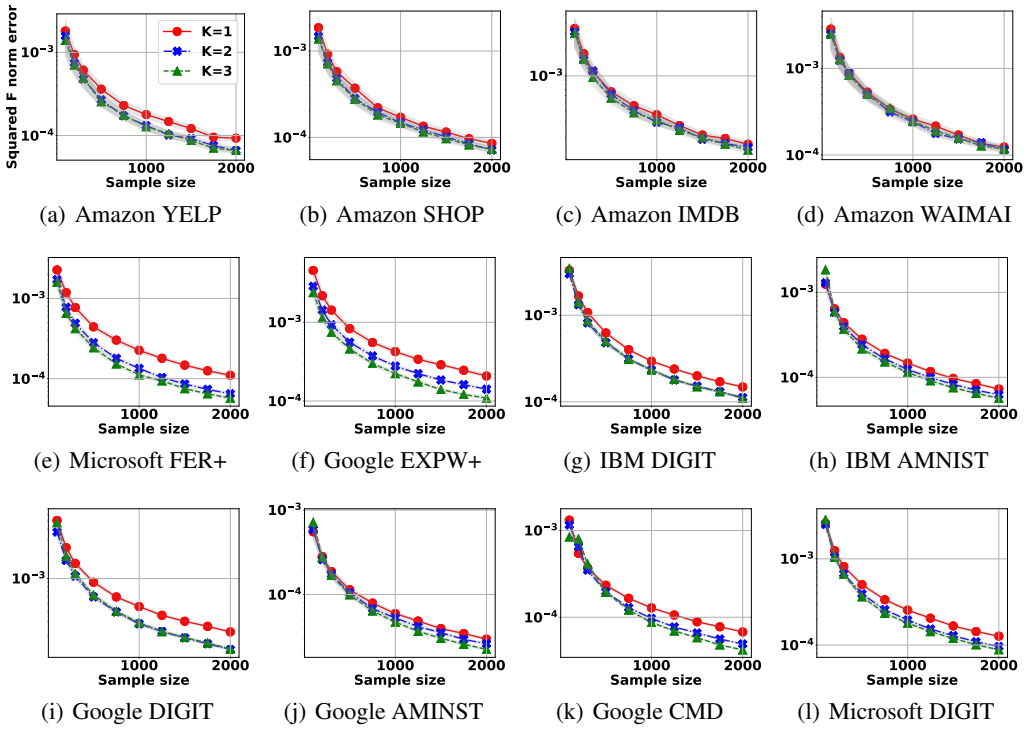
Figure 6: Confusion matrices of a few APIs in spring 2020/2021, along with their API shifts.

a phrase (e.g., "turn on the light" or "turn down the music"). There are in total 248 possible phrases, which are mapped to 31 unique labels. The sampling rate is also 16 kHz. All those datasets are freely available online for academic purposes.

Some of the datasets may contain personal information. For example, the human faces contained in the facial emotion recognition dataset may be deemed as personal information. On the other hand, our study focuses on whether there is a performance change on the dataset, and does not use or disclose any personal information.

For sentiment analysis, we use the Google NLP API (GoN), Amazon Comprehend API (Ama), and the Baidu NLP API (Bai). For facial emotion recognition, we use Google Vision API (Goo, a), Microsoft Face API (Mic, a), and the Face++ API (Fac). For spoken command recognition, we adopt Google speech API (Goo, b), Microsoft Speech API (Mic, b), and IBM speech API (IBM).

Details of observed ML API Shifts. Now we present a few more observed ML API shifts, as shown in Figure 6. One observation is that individual entry's change in the API shift can be larger than the overall accuracy's. For example, as shown in Figure 6 (c), the overall accuracy change is about -1.1% for Amazon on IDMB, but the performance drop for positive texts is as large as 5%. This indicates the importance of using fine-grained confusion matrix difference to measure API shifts. In addition, when the overall accuracy increases, it is possible that the accuracy for each label has been improved. This can be easily verified by Figure 6 (d-f). On the other hand, as shown in Figure 6 (g-i), Google API's large accuracy improvement (24%) is mostly because it is able to correctly predict many samples that were previously deemed as empty. One possible explanation is that Google API internally uses a higher threshold to generate a recognition. When the number of label increases, it



(a) Amazon YELP (b) Amazon SHOP (c) Amazon IMDB (d) Amazon WAIMAI

(e) Microsoft FER+ (f) Google EXPW+ (g) IBM DIGIT (h) IBM AMNIST

(i) Google DIGIT (j) Google AMINST (k) Google CMD (l) Microsoft DIGIT

Figure 7: Effects of partition parameter K. The total number of partitions is LK, and thus Larger K implies more partitions. Generally, across 12 cases where API shifts are identified, larger number of partitions usually leads to smaller estimation error for large samples. In practice, we observe that $K = 3$ is enough to reach good error rate.

might become hard to manually check the API shifts. For those cases, an anomaly detector can be applied to quickly identify the most surprising components in the API shifts.

Partition size's effects on MASA. Now we study how the partition number affects the performance of MASA, as shown in Figure 7. Across all API shifts we estimated, we note that larger number of partitions leads to a smaller overall Frobenious norm in general. This is expected, as larger K effectively introduces more parameters to estimate and thus is more powerful. The trade-off is that the computational cost increases, and more samples are needed for initial estimation. Interestingly, as K becomes large, the relative error reduction improvement becomes small. This is probably because there is no strong uncertainty difference within small partitions. In practice, we found that $K = 3$ already gives a small enough error reduction.

Comparison with baselines for case study on YELP. To further understand MASA's performance, We compared the performance of MASA with two baselines: random sampling and standard stratified sampling (proportionate allocation). We drawed 2000 samples for all methods, and repeated the experiments 1000 times to obtain an average of the Frobenius norm error. MASA outperforms both baselines significantly: the observed error is 0.015 for random sampling, 0.009 for stratified sampling, and 0.006 for MASA.

Understanding uncertainty score. MASA is developed based on the notion of uncertainty scores, and thus it is worthy understanding how uncertainty scores of different partitions for an ML API are computed. Here, we provide an illustrative example, as shown in Figure 8. The dataset contains three partitions and each partition includes six data points. We use a small ball to represent each data point, its interior color to denote its true label, and its edge color to denote the predicted label of an evaluated ML API. For example, on partition 1 and partition 3, all edge colors match interior colors,

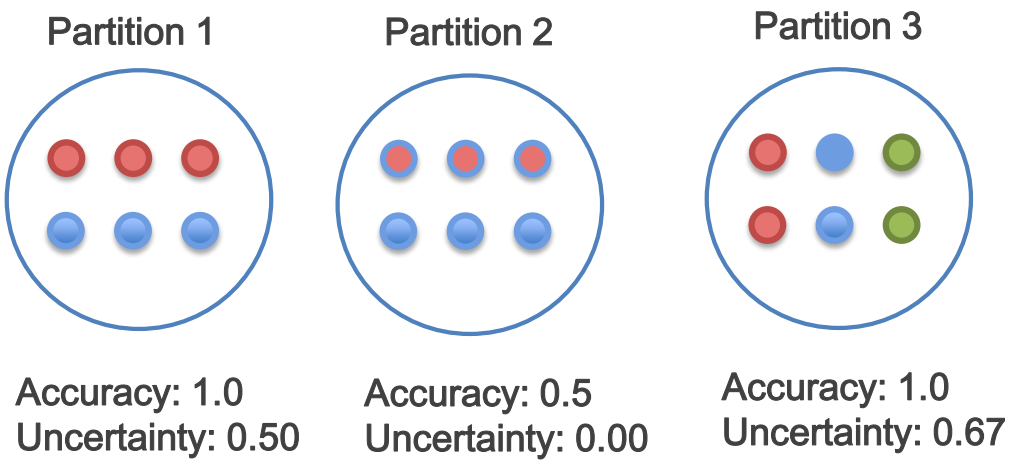


Figure 8: Illustrative examples of uncertainty scores. The dataset contains three partitions, each of which includes six data points. Here we use a ball to represent a data point, its interior color to denote its true label, and its edge color to indicate an ML API's predicted label. For example, as shown in the left panel, three points are labeled as red and the other three are labeled as blue. All points are predicted correctly, and thus the accuracy is 1.0. As the ML API predicts half of the points as red and half as blue, the uncertainty score is $1 - 0.5 \times 0.5 - 0.5 \times 0.5 = 0.5$. Note that high accuracy does not necessarily imply low uncertainty. For example, accuracy on partition 1 (1.0) is higher than that on partition 2(0.5), but its uncertainty score is actually larger than the latter. Yet, high diversity in the predicted labels does imply higher uncertainty. For example, while accuracy on partition 1 and partition 3 are both perfect, partition 3 incurs a higher uncertainty. This is because while only two unique predicted labels exist in partition 1, three occur in partition 3.

and thus the accuracy is 1.0. On partition 2, interior and edge colors match only on half of the points, and thus the accuracy is only 0.5.

To understand the calculation of the uncertainty score, let us take partition 1 as an example. The ML API predicts the label red for half of the partition and blue for the other half. Thus, the uncertainty score is 1 subtracting the sum of the square of likelihood of each predicted label, i.e., $1 - 0.5 \times 0.5 - 0.5 \times 0.5 = 0.5$. Similarly, on partition 2, the ML API always predicts the label blue, and thus the uncertainty is simply $1 - 1 = 0$. On partition 3, the ML API evenly predicts three unique labels, and thus the uncertainty score becomes $1 - \frac{1}{3} \times \frac{1}{3} - \frac{1}{3} \times \frac{1}{3} - \frac{1}{3} \times \frac{1}{3} = \frac{1}{3} \approx 0.67$.

Two observations are worthy mentioning about uncertainty scores, in addition to their non-negativity and upper bound of 1. First, higher accuracy on a partition does not imply lower uncertainty. To see this, note that the accuracy on partition 1 (1.0) is higher than that on partition 2 (0.5), but its uncertainty score is actually larger than that of partition 2. In fact, an API's accuracy on a partition is orthogonal to its uncertainty, as uncertainty score only depends on the predicted labels and is independent of the true labels. Second, diversity of the predicted labels is correlated to the uncertainty score. For example, the accuracy is same on partition 1 and 3, but the uncertainty score is higher on partition 3, mainly because there are three unique labels (red, blue, and green) in partition 3. This is expected, as uncertainty scores are designed to capture how diverse an ML API's prediction can be.