Learning-based Group Testing and Histogram Query

May 25, 2017

1 Introduction

In information retrieval process, we dedicate in reducing the communication cost between the interaction with the database. Most circumstances, we exploit the special structure of the problem, and reduce the complexity of the interaction (or *query*) between the user and the database. For example, in *sparse group testing* problem, the goal is to recover the binary categorical data, which may represent either positive or negative, through a series of queries. Typically the group testing problem is assumed to be sparse; that is, the number of defectives (negative items) are at most k, where k is much smaller then the number of total item $n.$ If we leverage the sparse structure, it is shown that instead of exhaustive search, which takes complexity $O\left(n\right)$, one can recover the entire dataset by (non-adaptive) random sampling with $O\left(k\log n\right)$ queries. The similar approach also works in *sparse histogram query problem*.

However, in real life, dataset or signal usually doesn't have natural sparse structure, and the approach based on sparsity thus not always hold. Therefore, in this research we develop a new strategy: first we *learn* the sparse sparsity through a small proportion of data, and then utilize the sparsity to further reduce the query complexity.

2 Main Results

In this section, we give some result about learning-based query, which can be roughly divided into adaptive and non-adaptive mechanism.

2.1 Learning-based Adaptive Query

First, we investigate the group testing problem. In the group testing problem, the dataset is denoted as $\mathcal{D} = (y_1, y_2, ..., y_n) \in \{0,1\}^n$. Besides, we are also given some side information $x_1, x_2, ..., x_n$. Under the assumption of realizable *PAC* model, there exists a *learnable* hypothesis class $\mathcal{H}$, such that

$$\exists h^* \in \mathcal{H},\ \text{s.t. } \forall i, y_i = h^*(x_i).$$

To reduce the query complexity, we proposed the following scheme: Initially, the entire dataset is divided into $\ell+1$ segments with size $n_1, ..., n_{\ell+1}$:

$$\mathcal{D} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup ... \cup \mathcal{S}_{\ell+1}, \forall i \neq j, \mathcal{S}_i \cap \mathcal{S}_j = \emptyset,$$

where the size of each segment is predetermined. In the beginning, we recover the first proportion of dataset, and use the recovered data to learn a hypothesis $h_1 \in \mathcal{H}$. Then, utilizing h_1, we can

recover the second proportion of dataset with sparsity ϵ_1 by decoding $\mathcal{S}_2 \cap h_1^{-1}(1)$ and $\mathcal{S}_2 \cap h_1^{-1}(0)$ respectively. Note that the sparsity level ϵ_1 depends on $|\mathcal{S}_1| = n_1$.
Therefore, one can decode the $i-$th proportion with ϵ_{i-1} sparsity of data by leveraging the previous samples $\bigcup_{j=1}^{i-1} \mathcal{S}_j$.

Algorithm 1 Adaptive Learning-based Group Testing under Realizable Assumption (AdpLGT)

```
Input: $\mathcal{D}, n_1, n_2, ..., n_{\ell+1}$
Output: The recovered data $(y_1, ..., y_n)$.
for $i = 1$ to $\ell + 1$ do
    $\mathcal{S}_i \leftarrow$ Random Select $n_i$ items from $\mathcal{D} - \bigcup_1^{i-1} \mathcal{S}_j$;
    $h_i \leftarrow$ ERM($\bigcup_1^{i-1} \mathcal{S}_j$);
    Run ordinary GT algorithm on $\mathcal{S}_i \cap h_i^{-1}(0)$ and $\mathcal{S}_i \cap h_i^{-1}(1)$;
    Determine $(y_{n_{i-1}+1}, y_{n_{i-1}+2}..., y_{n_i})$;
end for
```

Theorem 1 (Adaptive Learning-based Group Testing under Realizable Assumption) *The time complexity of AdpLGT is*

$$O(d \cdot \ell \cdot n^{\frac{1}{\ell+1}} \cdot (\log n)^2),$$

if we set $n_1, ..., n_{\ell+1}$ properly, where d is the VC-dimension *of $\mathcal{H}$.*

Proof. This is equivalent to the following optimization problem:

$$\min_{k,\epsilon} \; k_1 + k_2\epsilon_1 \log n + k_3\epsilon_2 \log n + ... + k_{\ell+1}\epsilon_\ell \log n,$$

$$\begin{aligned}
\text{subject to } \; & k_1 +, ..., k_{\ell+1} = n, \\
& k_1 \geq C \frac{d \log\left(\frac{1}{\epsilon_1}\right) + \log\left(\frac{\ell}{\delta}\right)}{\epsilon_1}, \\
& k_1 + k_2 \geq C \frac{d \log\left(\frac{1}{\epsilon_2}\right) + \log\left(\frac{\ell}{\delta}\right)}{\epsilon_2}, \\
& k_1 + k_2 + k_3 \geq C \frac{d \log\left(\frac{1}{\epsilon_3}\right) + \log\left(\frac{\ell}{\delta}\right)}{\epsilon_3}, \\
& \vdots \\
& k_1 + ... + k_\ell \geq C \frac{d \log\left(\frac{1}{\epsilon_\ell}\right) + \log\left(\frac{\ell}{\delta}\right)}{\epsilon_\ell}.
\end{aligned}$$

Assume that δ is a constant, then the following is a feasible solution:

$$\begin{aligned}
&\epsilon_1 = n^{\frac{1}{\ell+1}}, \epsilon_2 = n^{\frac{2}{\ell+1}}, ..., \epsilon_\ell = n^{\frac{\ell}{\ell+1}},\\
&k_1 = C \cdot n^{\frac{1}{\ell+1}} \left(\frac{d}{\ell+1} \log n + C_2 \log \ell \right),\\
&k_2 = C \cdot n^{\frac{2}{\ell+1}} \left(\frac{2d}{\ell+1} \log n + C_2 \log \ell \right),\\
&k_3 = C \cdot n^{\frac{3}{\ell+1}} \left(\frac{3d}{\ell+1} \log n + C_2 \log \ell \right),\\
&\vdots\\
&k_\ell = C \cdot n^{\frac{\ell}{\ell+1}} \left(\frac{\ell d}{\ell+1} \log n + C_2 \log \ell \right).
\end{aligned}$$

By choosing that $n_i = k_i$ in the AdpLGT, we see that the query complexity turns out to be

$$O(d \cdot \ell \cdot n^{\frac{1}{\ell}} \cdot (\log n)^2).$$

■

Corollary 2 *The result of the optimization problem given in Theorem 1 is asymptotically tight up to* $\log n$ *factor.*

Corollary 3 *The same asymptotic bound holds for the adaptive histogram query problem under realizable assumption.*

2.2 Agnostic Learning-based Adaptive Query

So far, we assume that the side information $\{x_i\}$ is related through a hypothesis class $\mathcal{H}$ to the target dataset $\{y_i\}$. However, in the real world, this assumption usually too strong to hold. Therefore in this section, we investigate the similar adaptive querying approach under the assumption of *agnostic* PAC learnable.

We say a concept is agnostic PAC learnable, if there exists a algorithm $\mathcal{A}_{\epsilon,\delta}$, such that with probability at least $1 - \delta$,

$$\text{err}\left(\text{output}\left(\mathcal{A}_{\epsilon,\delta}\right)\right) - \min_{h \in \mathcal{H}} \text{err}(h) \leq \epsilon.$$

According to literatures from learning theory, a hypothesis class is agnostic PAC learnable if and only if it has finite VC dimension, and the sample complexity is given by

$$m_{\mathcal{H}}(\epsilon, \delta) = C \cdot \frac{d + \log\left(\frac{1}{\delta}\right)}{\epsilon^2},$$

for some universal constant C.

For the agnostic scenario, we follow the same approach :
Initially, the entire dataset is divided into $\ell + 1$ segments:

$$\mathcal{D} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup ... \cup \mathcal{S}_{\ell+1}, \forall i \neq j, \mathcal{S}_i \cap \mathcal{S}_j = \emptyset.$$

In the beginning, we recover the first proportion, and use the recovered data to learn a hypothesis $h_1 \in \mathcal{H}$. However, in agnostic case, utilizing h_1 we can only recover the second proportion of dataset with $(\epsilon_0 + \epsilon_1)-$sparse, where ϵ_0 denotes $\min_{h \in \mathcal{H}} \text{err}(h)$. Therefore, one can decode the $i-$th proportion with $(\epsilon_0 + \epsilon_{i-1})$ sparsity of data by leveraging the previous samples $\bigcup_{j=1}^{i-1} \mathcal{S}_j$. Intuitively, one can not obtain a better (smaller) sparsity then ϵ_0, and hence the ultimate query complexity should be at least $\Omega\left(\epsilon_0 \cdot n \log n\right)$. Thus we see the following result:

Theorem 4 (Agnostic Learning-based Group Testing) .
The time complexity of AdpLGT under agnostic assumption *is*

- *If* $n \ll \frac{1}{\epsilon_0^3}$*, then the query complexity is*

$$O(d \cdot \ell \cdot n^{\frac{2}{3}} \cdot (\log n)^2), \textit{ for } \ell = 1$$
$$O(d \cdot \ell \cdot n^{\frac{4}{3\ell+4}} \cdot (\log n)^2), \textit{ for } \ell > 1$$

- *If* $n \gg \frac{1}{\epsilon_0^3}$*, then the query complexity is*

$$O(\epsilon_0 \cdot n \cdot (\log n)^2),$$

if we set $n_1, ..., n_{\ell+1}$ *properly.*

Proof. The proof is similar to Theorem 1, and is ommited here. ■

Corollary 5 *This result hold for the histogram query problem.*

2.3 Non-adaptive Learning-based Query

In this section, we consider the converse bound of non-adaptive learning based query, under the realizable assumption. First, we give a converse bound on group testing problem and histogram query problem.

Theorem 6 (Converse Bound on Non-adaptive Group Testing) .
For non-adaptive group testing problem, the query complexity is at least $\Omega\left(d \log n\right)$*, where* d *is the VC dimension of* $\mathcal{H}$*.*

Proof. First note that the recovery will fail if and only if there exist one more $\tilde{\mathbf{y}} = (\tilde{y_1}, \tilde{y_2}, ..., \tilde{y_n}) \neq \mathbf{y}$, such that $\tilde{\mathbf{y}}$ consistent with $\mathbf{x} = (x_1, x_2, ..., x_n)$.
Formally, denote the query matrix as $\mathbf{Q}_m \in \{0,1\}^{n \times m}$, then the event of failure is:

$$\exists \tilde{\mathbf{y}} \neq \mathbf{y}, \text{such that } \mathbf{Q}_m \mathbf{y} = \mathbf{Q}_m \tilde{\mathbf{y}}, \text{and } \exists h_1, h_2 \in \mathcal{H}, \tilde{\mathbf{y}} = h_1(\mathbf{x}), \mathbf{y} = h_2(\mathbf{x}).$$

Note that the cardinality of query output is $|\{\mathbf{Q}_m \mathbf{y} \mid \mathbf{y} \in \{0,1\}^n\}| = 2^m$, and the possible input is at most

$$\left|\{h(\mathbf{x}) \mid h \in \mathcal{H}, \mathbf{x} \in \mathcal{X}^n\}\right|.$$

Since the VC dimension of $\mathcal{H}$ is d, the growth of possible $\mathbf{y}$ is upper bounded by $\left(\frac{en}{d}\right)^d$. Therefore, setting

$$2^m \geq \left(\frac{en}{d}\right)^d,$$

we obtain

$$m \geq d\left(\log e + \log n - \log d\right) \to d \log n,$$

and thus $m = \Omega(d \log n)$. ■

Corollary 7 *The query complexity is at least* $\Omega\left(d\right)$ *for the histogram query problem.*

Proof. Similar to the proof of thm 6, the event of failure is:

$$\exists \tilde{\mathbf{y}} \neq \mathbf{y}, \text{such that } \mathbf{Q}_m\mathbf{y} = \mathbf{Q}_m\tilde{\mathbf{y}}, \text{and } \exists h_1, h_2 \in \mathcal{H}, \tilde{\mathbf{y}} = h_1(\mathbf{x}), \mathbf{y} = h_2(\mathbf{x}).$$

Note that the cardinality of query output is $|\{\mathbf{Q}_m\mathbf{y} \mid \mathbf{y} \in \{0,1,...,n\}^n\}| = (n+1)^m$, and the possible input is at most

$$|\{h(\mathbf{x}) \mid h \in \mathcal{H}, \mathbf{x} \in \mathcal{X}^n\}|.$$

Since the VC dimension of $\mathcal{H}$ is d, the growth of possible $\mathbf{y}$ is upper bounded by $\left(\frac{en}{d}\right)^d$. Therefore, setting

$$(n+1)^m \geq \left(\frac{en}{d}\right)^d,$$

we obtain

$$m \geq d\left(\frac{\log e + \log n - \log d}{\log(n+1)}\right) \to d,$$

and thus $m = \Omega(d)$. ■

Inspired by the converse proof, we give the following algorithm for (randomized) non-adaptive histogram query problem:

Algorithm 2 Non-adaptive Learning-based Histogram Query (NAdpLHQ)

```
Input: $\mathcal{D}, \mathbf{x}, \mathcal{H}$
Output: The recovered data $\mathbf{y} = (y_1, ..., y_n)$.
Randomly generate query matrix $\mathbf{Q}_d \in \{0,1\}^{n\times d}$;
Compute $\mathbf{z} \leftarrow \mathbf{Q}_d \cdot \mathbf{y}$        ▷ $\mathbf{z} \in \{0,1\}^d$.
$\tilde{\mathbf{z}} \leftarrow \mathbf{0}$;
while $\tilde{\mathbf{z}} \neq \mathbf{z}$ do        ▷ exhaustively search for $\mathcal{H}$ until find a consistent one.
    $h \leftarrow h_i \in \mathcal{H}$;
    $\tilde{\mathbf{y}} \leftarrow h(\mathbf{x})$;
    $\tilde{\mathbf{z}} \leftarrow \mathbf{Q}_d \cdot \mathbf{y}$;
end while
output $h(\mathbf{x})$;
```

A modified algorithm for group testing:

Algorithm 3 Non-adaptive Learning-based Group Testing (NAdpLGT)

```
Input: $\mathcal{D}, \mathbf{x}, \mathcal{H}$
Output: The recovered data $\mathbf{y} = (y_1, ..., y_n)$.
Calculate $m = d \cdot \log n$
Randomly generate query matrix $\mathbf{Q}_m \in \{0,1\}^{n\times m}$;
Compute $\mathbf{z} \leftarrow \mathbf{Q}_m \cdot \mathbf{y}$        ▷ $\mathbf{z} \in \{0,1\}^m$.
$\tilde{\mathbf{z}} \leftarrow \mathbf{0}$;
while $\tilde{\mathbf{z}} \neq \mathbf{z}$ do        ▷ exhaustively search for $\mathcal{H}$ until find a consistent one.
    $h \leftarrow h_i \in \mathcal{H}$;
    $\tilde{\mathbf{y}} \leftarrow h(\mathbf{x})$;
    $\tilde{\mathbf{z}} \leftarrow \mathbf{Q}_m \cdot \mathbf{y}$;
end while
output $h(\mathbf{x})$;
```

Similar to the idea of random querying in histogram query's literatures, these algorithm is proved to succeed with high probability, as n tends to infinity. We state as the following theorem:

Theorem 8 (Achievability of Learning-base Non-adaptive Histogram Query) .

$$\mathbb{P}\left\{\textit{Algorithm}2 \textit{ successfully recover the data } \right\} \to 1, \textit{ as } n \to \infty.$$

In other words, the query complexity of histogram query is $O\left(d\right)$.

Proof. The proof is very similar to the proof of ordinary histogram query problem, and is ommited here. ■

Remark 9 *Theorem 8 guarantees that as* n *large enough, there will exists one (and* only *one) hypothesis* $h^* \in \mathcal{H}$ *consistent with all side information* $\mathbf{x}$ *and all the queried output* $\mathbf{z} = \mathbf{Q}_d \cdot \mathbf{y}$*, and therefore* d *randomly generated queries are enough to decode the entire dataset.*
However, though we proved that d*'s queries are sufficient to uniquely determine the hypothesis* h^**, we did not give a computational efficient algorithm to find the* exact h^**. Therefore the VC dimension* d *is the information-theoretic bound but not the computational bound.*