ECE 6980
Algorithmic and Information-Theoretic Methods in Data Science

Instructor: Jayadev Acharya — Lecture #7
Scribe: Cody Freitag — 20th September, 2017

1 Introduction

In this lecture, we will show a general proper learning algorithm. We start by solving a simpler problem of approximating an unknown distribution by the closest distribution in a known set. Then we use that algorithm as a black box for general learning.

2 Choosing a Density Estimate

[1]For the standard distribution learning problem, you are given samples from some unknown distribution p, and you want to estimate any $\hat{p}$ that is "close" to p. In developing a general algorithm for this problem, we first focus our attention on a restricted version of this problem where we must choose $\hat{p}$ from a known set of distributions. We formalize this idea in Problem 1.

Problem 1 (Choosing a Density Estimate)**.**
Given:
- Distributions $p_1, \ldots, p_M$ (known)
- Samples $x_1, \ldots, x_n \sim p$ (unknown)

Goal:
- Output $\arg\min_i d_{\mathrm{TV}}(p, p_i)$

For general distributions, we can't solve this problem, so we instead try to find an approximate solution. The approximate solution should still be one of the original distributions and be within some multiplicative factor α and additive factor of β within the optimal solution. Also, we want to succeed with at least some fixed constant probability, say 0.95.

Approximate Goal:
- Output $p^* \in \{p_1, \ldots, p_M\}$ such that w.p. > 0.95, $d_{\mathrm{TV}}(p, p^*) \leq \alpha \cdot \min_i d_{\mathrm{TV}}(p, p_i) + \beta$.

We will show in this section that we can achieve this approximate goal for any possible inputs for $\alpha = 9$ and $\beta = O(\sqrt{\frac{\log M}{n}})$. We'll first solve a simpler case where $M = 2$ and extend this to the case of general M.

2.1 The Case of $M = 2$

Given two known possible distributions, p_1 and p_2, we first define $A = \{x : p_1(x) > p_2(x)\}$. We draw $x_1, \ldots, x_n \sim p$, and compute a test statistic of what fraction of the drawn sample are in A.

[1]Almost all of the material in this section comes from Chapter 6 of Devroye and Lugosi's book, Combinatorial Methods in Density Estimation [DL01].

Finally, we output the distribution whose probability mass on the elements in A is closest to our test statistic. Intuitively, if p is closer in total variation distance to p_1, it should have a greater mass on A than p_2. We explicitly state this idea, which we call Scheffé's Estimator, in Algorithm 1.

Algorithm 1 Scheffé's Estimator

Input: p_1, p_2 (known), p (unknown)

1: Let $A = \{x : p_1(x) > p_2(x)\}$.

2: Draw n samples $x_1, \ldots, x_n \sim p$.

3: Compute $\mu_n(A) = \frac{1}{n}|\{j : x_j \in A\}|$.

4: Output $p^* = \begin{cases} p_1 & \text{if } |p_1(A) - \mu_n(A)| < |p_2(A) - \mu_n(A)| \\ p_2 & \text{o.w.} \end{cases}$.

Theorem 1. *Let $\Delta = \min\{d_{\mathrm{TV}}(p, p_1), d_{\mathrm{TV}}(p, p_2)\}$. Given $x_1, \ldots, x_n \sim p$, Scheffé's Estimator of Algorithm 1 computes $p^* \in \{p_1, p_2\}$ such that w.p. > 0.95,*

$$d_{\mathrm{TV}}(p, p^*) \leq 3\Delta + O\left(\frac{1}{\sqrt{n}}\right).$$

Proof. We first show that $d_{\mathrm{TV}}(p, p^*) \leq 3\Delta + 2|p(A) - \mu_n(A)|$.

Let

$$\xi = \begin{cases} p_1 & \text{if } d_{\mathrm{TV}}(p, p_1) < d_{\mathrm{TV}}(p, p_2) \\ p_2 & \text{o.w.} \end{cases},$$

then $d_{\mathrm{TV}}(p, p^*) \leq d_{\mathrm{TV}}(p, \xi) + d_{\mathrm{TV}}(\xi, p^*)$ by the triangle inequality. Note that by definition $d_{\mathrm{TV}}(p, \xi) = \Delta$, so we need to show that $d_{\mathrm{TV}}(\xi, p^*) \leq 2\Delta + 2|p(A) - \mu_n(A)|$.

Note that when $\xi = p^*$, $d_{\mathrm{TV}}(x_i, p^*) = 0$. So we need to bound the events when $\xi = p_1, p^* = p_2$ or $\xi = p_2, p^* = p_1$. Let E be the event that $\xi = p_1, p^* = p_2$.

$$\begin{aligned} d_{\mathrm{TV}}(\xi, p^*) \cdot \mathbb{I}_{\mathrm{E}} &= (p_1(A) - p_2(A)) \cdot \mathbb{I}_{\mathrm{E}} \\ &= (p_1(A) - \mu_n(A)) \cdot \mathbb{I}_{\mathrm{E}} + (\mu_n(A) - p_2(A)) \cdot \mathbb{I}_{\mathrm{E}} \\ &\leq 2(p_1(A) - \mu_n(A)) \cdot \mathbb{I}_{\mathrm{E}} \qquad (\diamond) \\ &= 2(p_1(A) - p(A)) \cdot \mathbb{I}_{\mathrm{E}} + 2(p(A) - \mu_n(A)) \cdot \mathbb{I}_{\mathrm{E}} \\ &\leq 2\Delta + 2|p(A) - \mu_n(A)| \end{aligned}$$

Line $(\diamond)$ follows because $p_1(A) \geq p_2(A)$ and $p^* = p_2$. This implies that $\mu_n(A) \leq \frac{1}{2} \cdot (p_1(A) + p_2(A))$. Thus, $\mu_n(A) - p_2(A) \leq p_1(A) - \mu_n(A)$.

We now make a similar argument for the case when $\xi = p_2$ and $p^* = p_1$. Let E' be the corresponding event.

$$\begin{aligned} d_{\mathrm{TV}}(\xi, p^*) \cdot \mathbb{I}_{\mathrm{E}'} &= (p_1(A) - p_2(A)) \cdot \mathbb{I}_{\mathrm{E}'} \\ &= (p_1(A) - \mu_n(A)) \cdot \mathbb{I}_{\mathrm{E}'} + (\mu_n(A) - p_2(A)) \cdot \mathbb{I}_{\mathrm{E}'} \\ &\leq 2(\mu_n(A) - p_2(A)) \cdot \mathbb{I}_{\mathrm{E}'} \qquad (\triangle) \\ &= 2(p(A) - p_2(A)) \cdot \mathbb{I}_{\mathrm{E}'} + 2(\mu_n(A) - p(A)) \cdot \mathbb{I}_{\mathrm{E}'} \\ &\leq 2\Delta + 2|p(A) - \mu_n(A)|. \end{aligned}$$

Line ($\triangle$) follows because $p_1(A) \geq p_2(A)$ and $p^* = p_1$, so $\mu_n(A) \geq \frac{1}{2} \cdot (p_1(A) + p_2(A))$. Thus, $\mu_n(A) - p_2(A) \geq p_1(A) - \mu_n(A)$.

Putting these pieces together, we get that

$$\begin{aligned} d_{\mathrm{TV}}(p, p^*) &\leq d_{\mathrm{TV}}(p, \xi) + d_{\mathrm{TV}}(\xi, p^*) \\ &\leq \Delta + d_{\mathrm{TV}}(\xi, p^*) \cdot (\mathbb{I}_{\mathrm{E}} + \mathbb{I}_{\mathrm{E}'}) \\ &\leq 3\Delta + 2|p(A) - \mu_n(A)| \end{aligned}$$

We now briefly argue why $2|p(A) - \mu_n(A)| \leq O(\frac{1}{\sqrt{n}})$. Note $n \cdot \mu_n(A)$ is the sum of n i.i.d. Bernoulli random variables with mean $p(A)$. Using Chebyshev's inequality, we get

$$\begin{aligned} \Pr(|n \cdot \mu_n(A) - n \cdot p(A)| > c \cdot \sqrt{n}) &\leq \frac{n \cdot p(A) \cdot (1 - p(A))}{c^2 \cdot n} \\ &\leq \frac{1}{4c^2} \end{aligned}$$

where the second inequality follows because $p(A) \cdot (1 - p(A)) \leq \frac{1}{4}$ for any value of $p(A) \in [0, 1]$. Normalizing by a factor of n implies that $|p(A) - \mu_n(A)| \leq O(\frac{1}{\sqrt{n}})$ with probability at least $1 - \delta$ for any constant $\delta > 0$. In particular, with probability at least 0.95, we have that

$$\begin{aligned} d_{\mathrm{TV}}(p, p^*) &\leq 3\Delta + 2|p(A) - \mu_n(A)| \\ &\leq 3\Delta + O\left(\frac{1}{\sqrt{n}}\right). \end{aligned}$$

□

2.2 The Case of General M

We will use Scheffe's Estimator for $M = 2$ as a black box to construct an estimator for general M, which we will call the General Scheffé's Estimator.

The high level idea is that we run a "match" using Scheffé's Estimator between every pair of M known probability distributions. We say that a distribution p_i "wins" the match against p_j if Scheffé's Estimator outputs p_i. The algorithm then outputs the distribution with the most wins. Intuitively, any distribution close to p should win against many other distributions, so this should find a good approximation. We explicitly state the General Scheffé's Estimator in Algorithm 2.

Algorithm 2 General Scheffé's Estimator

Input: $p_1, \ldots, p_M$ (known), p (unknown)

1: For each $i \neq j$, run Scheffé's Estimator on p_i, p_j for p.
2: Let W_i be the number of times p_i is output by Scheffé's estimator.
3: Output p_i such that $i = \arg\max_i W_i$.

Theorem 2. *Let $\Delta = \min_i d_{\mathrm{TV}}(p, p_i)$. Given $x_1, \ldots, x_n \sim p$, General Scheffé's Estimator of Algorithm 2 computes $p^* \in \{p_1, \ldots, p_M\}$ such that w.p. > 0.95,*

$$d_{\mathrm{TV}}(p, p^*) \leq 9\Delta + O\left(\sqrt{\frac{\log M}{n}}\right).$$

Proof. For the match between p_i and p_j, let $\Delta_{i,j} = \min\{d_{\mathrm{TV}}(p, p_i), d_{\mathrm{TV}}(p, p_j)\}$ and $A_{i,j} = \{x : p_i(x) > p_j(x)\}$. In the proof of Theorem 1, we showed that for a single "match" between p_i, p_j, the winner p^* must satisfy

$$d_{\mathrm{TV}}(p, p^*) \leq 3\Delta_{i,j} + 2|p(A_{i,j}) - \mu_n(A_{i,j})|.$$

However, note the match between p_i and p_j depends on the random variable $|p(A) - \mu_n(A)|$. Thus, we need a bound that holds for all $\binom{M}{2}$ matches rather than just a single match. This means the general bound we get for any match between p_i, p_j is

$$d_{\mathrm{TV}}(p, p^*) \leq 3\min\{d_{\mathrm{TV}}(p, p_i), d_{\mathrm{TV}}(p, p_j)\} + 2\max_{A_{i,j}}\{|p(A_{i,j}) - \mu_n(A_{i,j})|\}.$$

Each random variable $\mu_n(A_{i,j})$ is a binomial distribution with mean $p(A_{i,j})$, so it is subgaussian. Subtracting $p(A)$ and taking the absolute value to get $|p(A) - \mu_n(A)|$ doesn't affect the tail behavior, so $|p(A_{i,j} - \mu_n(A_{i,j})|$ is also subgaussian. In particular, this implies that the expectation of the maximum of M^2 such random variables is at most $\sqrt{\log m}$ times the expected value of a single random vairable, which we showed was $\frac{1}{\sqrt{n}}$. This implies there is some constant c such that for any constant $\delta > 0$,

$$d_{\mathrm{TV}}(p, p^*) \leq 3\min\{d_{\mathrm{TV}}(p, p_i), d_{\mathrm{TV}}(p, p_j)\} + c \cdot \sqrt{\frac{\log m}{n}}$$

with probability at least $1 - \delta$ for all matches run in the General Scheffe's Estimator.

Let $\Delta_{\min} = \min_{i,j} \Delta_{i,j}$. We define the following four groups that the distributions $p_1, \ldots, p_M$ can fall into.

- $G_1 = \{p_i : d_{\mathrm{TV}}(p, p_i) = \Delta\}$
- $G_2 = \{p_i : d_{\mathrm{TV}}(p, p_i) \in (\Delta, 3\Delta + c \cdot \sqrt{\frac{\log m}{n}}]\}$
- $G_3 = \{p_i : d_{\mathrm{TV}}(p, p_i) \in (3\Delta + c \cdot \sqrt{\frac{\log m}{n}}, 9\Delta + 4c \cdot \sqrt{\frac{\log m}{n}}]\}$
- $G_4 = \{p_i : d_{\mathrm{TV}}(p, p_i) \in (9\Delta + 4c \cdot \sqrt{\frac{\log m}{n}}, \infty)\}$

Note that any distribution p_i in G_1 must win against any p_j in G_3 or G_4. Furthermore, there must be at least one distribution in G_1 by definition of Δ. This means that some distribution p_i wins at least $|G_3| + |G_4|$ matches. Now consider any distribution p_ℓ in G_4. p_ℓ must lose against any distribution in G_1 or G_2. In particular, any p_ℓ in G_4 can win at most $|G_3| + |G_4| - 1$ matches. This means that any distribution that wins the most matches must be in G_1, G_2, or G_3, so

$$d_{\mathrm{TV}}(p, p^*) \leq 9\Delta + O\left(\sqrt{\frac{\log m}{n}}\right).$$

□

Note this theorem implies that for our algorithm to output $p^* \in \{p_1, \ldots, p_M\}$ with $d_{\mathrm{TV}}(p, p^*) \leq 9\Delta + O(\epsilon)$, we requires that $n = \Omega(\frac{\log M}{\epsilon^2})$.

3 A General Learning Algorithm

Using the General Scheffé's Estimator for choosing a density estimate given in the previous section, we construct a general learning algorithm. The main idea is to first sample a set of points from

p that allows you to construct a lot of candidate distributions. Then you can use the result of Theorem 2 to pick one that is closest to p. As long as some candidate distribution is close to p, this will yield a good distribution.

We'll see how this method is applied to learning mixtures of k 1-dimensional Gaussians. We start with the easier case of $k = 1$ and then show how this can be generalized to learn a mixture of k Gaussians. We'll finish with a discussion of how this can be applied to arbitrary distributions.

3.1 Learning a Single Gaussian

Suppose we want to learn a single Gaussian $p = \mathcal{N}(\mu, \sigma^2)$. If we can estimate the mean μ by $\hat{\mu}$ and variance σ^2 by $\hat{\sigma}^2$, then that gives an estimate $\hat{p} = \mathcal{N}(\hat{\mu}, \hat{\sigma}^2)$ close to p. Our algorithm will sample n_1 points from p. We will consider every sampled point as possible values for μ and use the squared distance between every pair of points as possible values for σ^2. We consider all distributions with those mean and variance values, and apply the General Scheffé's Estimator on those distributions to find one close to p. We explicitly state this idea in Algorithm 3.

Algorithm 3 Learning Algorithm for a Single Gaussian

Input: $p = \mathcal{N}(\mu, \sigma^2)$ (known Gaussian with unknown parameters)

1: Draw n_1 samples $x_1, \ldots, x_{n_1} \sim p$.
2: Let $\bar{\mu} = \{x_1, \ldots, x_{n_1}\}$ and $\bar{\sigma}^2 = \{(x_i - x_j)^2 : i \neq j \in [n_1]\}$.
3: Let $P = \{\mathcal{N}(\mu', \sigma'^2) : \mu' \in \bar{\mu}, \sigma'^2 \in \bar{\sigma}\}$.
4: Let $p^* \in P$ be the output of General Scheffé's Estimator on P for p.
5: Output $\hat{p} = p^*$.

We first start off by showing that if n_1 is large enough, then some choice of $\mathcal{N}(\mu', \sigma'^2)$ for $\mu' \in \bar{\mu}$ and $\sigma'^2 \in \bar{\sigma}^2$ is close to $\mathcal{N}(\mu, \sigma^2)$.

Lemma 1. *There exists a constant c such that if $n_1 \geq \frac{c}{\epsilon^2}$, then with probability > 0.95 some $\mu' \in \bar{\mu}$ and $\sigma'^2 \in \bar{\sigma}^2$ from Algorithm 3 satisfy*

$$d_{\mathrm{TV}}(\mathcal{N}(\mu', \sigma'^2), \mathcal{N}(\mu, \sigma^2)) \leq \epsilon.$$

Proof. With $n_1 = O(\frac{1}{\epsilon^2})$ samples, you can ensure with probability > 0.95 that there are some $i \neq j \in [n_1]$ such that $x_i \in [\mu - \frac{\sigma\epsilon^2}{4}, \mu + \frac{\sigma\epsilon^2}{4}]$ and $x_j \in [\mu + \sigma - \frac{\sigma\epsilon^2}{4}, \mu + \sigma + \frac{\sigma\epsilon^2}{4}]$. Let $\mu' = x_i$ and $\sigma' = x_j - x_i$.

You can show that the KL divergence between two gaussians is

$$KL(\mathcal{N}(\mu_1, \sigma_1^2), \mathcal{N}(\mu_2, \sigma_2^2)) = \log \frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_2 - \mu_1)^2}{2\sigma_2^2} - \frac{1}{2}$$

We use the fact that $|\mu' - \mu| \leq \frac{\sigma\epsilon^2}{4}$, $\sigma' < (1 + \frac{\epsilon^2}{2})\sigma$, and $\epsilon < 1$. Then plugging in $\mu_1 = \mu'$, $\sigma_1 = \sigma'$, $\mu_2 = \mu$, and $\sigma_2 = \sigma$ to the above equation, we get

$$\begin{aligned} KL(\mathcal{N}(\mu', \sigma'^2), \mathcal{N}(\mu, \sigma^2)) &\leq \frac{\epsilon^4}{16} + \frac{\epsilon^2}{2} + \log(1 + \frac{\epsilon^2}{2}) \\ &\leq 2\epsilon^2 \end{aligned}$$

Applying Pinsker's inequality, this implies that

$$\begin{aligned} d_{\mathrm{TV}}(\mathcal{N}(\mu', \sigma'^2), \mathcal{N}(\mu, \sigma^2)) &\leq \sqrt{\frac{1}{2} KL(\mathcal{N}(\mu', \sigma'^2), \mathcal{N}(\mu, \sigma^2))} \\ &\leq \sqrt{\frac{1}{2}(2\epsilon^2)} \\ &\leq \epsilon \end{aligned}$$

□

Theorem 3. *There exists some constant c such that with $n \geq \frac{c \log \frac{1}{\epsilon}}{\epsilon^2}$ samples, Algorithm 3 computes $\hat{p}$ such that $d_{\mathrm{TV}}(p, \hat{p}) \leq \epsilon$ with probability at least 0.9.*

Proof. Lemma 1 implies that there is some constant c' such that after $n_1 \geq \frac{c'}{\epsilon'^2}$ samples, some p_i in P has $d_{\mathrm{TV}}(p, p_i) \leq \epsilon'$ with probability at least 0.95.

Let M be the number of distributions in P. Then $M = n_1^3 \geq \frac{c'^3}{\epsilon'^6}$. Then with probability at least 0.9, running the General Scheffé's Estimator for M distributions returns a distribution $p^* \in P$ such that $d_{\mathrm{TV}}(p, p^*) \leq 9\epsilon' + O(\sqrt{\frac{\log M}{n}})$. Note that there exists some constant c'' such that if $n > \frac{c'' \log M}{\epsilon^2}$, then $O(\sqrt{\frac{\log M}{n}})$ is bounded by $\epsilon/2$. Also, we can set $\epsilon' = \epsilon/18$. This implies that $d_{\mathrm{TV}}(p, p^*) \leq \epsilon$.

The total number of samples used n must be at least

$$\begin{aligned} n &\geq n_1 + \frac{c'' \log M}{\epsilon^2} \\ &\geq \frac{c'}{\epsilon'^2} + \frac{c'' \log \frac{c'^3}{\epsilon'^6}}{\epsilon^2} \\ &= \Omega\left(\frac{\log \frac{1}{\epsilon}}{\epsilon^2}\right) \end{aligned}$$

The probability of success is at least 0.9 by a union bound since there will be a good distribution $p_i \in P$ with probability at least 0.95 and the General Scheffé's Estimator will find a good approximation with at least probability 0.95. □

3.2 Learning a Mixture of k Gaussians

We now briefly show how this general learning algorithm can be applied to estimating a mixture of k Gaussians. There are two main differences between this algorithm and the case where $k = 1$. First, our proposed distributions P must now be a sum of k Gaussians with unknown parameters and possible weights. This means we need to estimate $3k$ parameters instead of just 2. The second difference is that to account for this, we need to use more initial samples n_1 to construct the set P. We highlight these differences in Algorithm 4.

We state the following Lemma without proof for sake of brevity and because the idea is similar to the proof of Lemma 1.

Lemma 2. *If $n_1 = \Omega(\frac{k}{\epsilon^2})$, then with probability > 0.95 some $w_{x_1}, \ldots, w_{x_k} \in \bar{w}$, $\mu_{i_1}, \ldots, \mu_{i_k} \in \bar{\mu}$ and $\sigma^2_{j_1}, \ldots, \sigma^2_{j_k} \in \bar{\sigma}^2$ from Algorithm 4 satisfy*

$$d_{\mathrm{TV}}\left(\sum_{\ell=1}^{k} w_{x_\ell} \cdot \mathcal{N}(\mu_{i_\ell}, \sigma^2_{j_\ell}), p\right) \leq \epsilon.$$

Algorithm 4 Learning Algorithm for a Single Gaussian

Input: p (known to be a sum of k Gaussians with unknown parameters)

1: Draw n_1 samples $x_1, \ldots, x_{n_1} \sim p$.
2: Let $\bar{\mu} = \{x_1, \ldots, x_{n_1}\}$, $\bar{\sigma}^2 = \{(x_i - x_j)^2 : i \neq j \in [n_1]\}$, and $\bar{w} = \{\frac{\epsilon}{k}, \frac{2\epsilon}{k}, \ldots, 1\}$.
3: Let $P = \{\sum_{\ell=1}^{k} w_{x_\ell} \cdot \mathcal{N}(\mu_{i_\ell}, \sigma^2_{j_\ell}) : w_{x_\ell} \in \bar{w}, \mu_{i_\ell} \in \bar{\mu}, \sigma^2_{j_\ell} \in \bar{\sigma}^2\}$.
4: Let $p^* \in P$ be the output of General Scheffé's Estimator on P for p.
5: Output $\hat{p} = p^*$.

Theorem 4. *If $n = \Omega(\frac{k \log \frac{k}{\epsilon}}{\epsilon^2})$ samples, Algorithm 4 computes $\hat{p}$ such that $d_{\mathrm{TV}}(p, \hat{p}) \leq \epsilon$ with probability at least 0.9.*

Proof. The proof is very similar to Theorem 3 except for the number of distributions input to the Genreal Scheffé's Estimator, M, so we focus only on that.

From Lemma 2, $n_1 = \Omega(\frac{k}{\epsilon^2})$ samples suffice to find a good distribution with probability at least 0.9. Given this, there are n_1^k choices for μ, n_1^{2k} choices for σ^2, and $(\frac{k}{\epsilon})^k$ choices for w. This means in total there are $M = (n_1^3 \frac{k}{\epsilon})^k = \Omega((\frac{k^4}{\epsilon^7})^k)$ total distributions input to the General Scheffé's Estimator. Thus, the total number of samples used n must be at least

$$\begin{aligned} n &\geq n_1 + \Omega\left(\frac{\log M}{\epsilon^2}\right) \\ &= \Omega\left(\frac{k \log \frac{k}{\epsilon}}{\epsilon^2}\right). \end{aligned}$$

□

3.3 Learning Arbitrary Distributions

We have seen two examples of how to use a solution to Problem 1 to construct a general learning algorithm. What about for arbitrary distributions?

The general learning algorithm depends on the "complexity" of the distribution to be learned. Intuitively, a mixture of $k > 1$ Gaussians is more complex than a single Gaussian, so learning $k > 1$ Gaussians requires more samples. This notion of "complexity" is related to the metric entropy of a set of distributions.

3.3.1 Metric Entropy

We define the metric entropy by first introducing the concepts of an ϵ-cover and the covering number of a space.

Definition 1 (ϵ-Cover)**.** Let $(\mathcal{X}, d)$ be a metric space. A set $\mathcal{C} = \{x_1, \ldots, x_m\}$ is an ϵ-cover if for every $x \in \mathcal{X}$ there is some $x_i \in \mathcal{C}$ such that $d(x, x_i) < \epsilon$.

Definition 2 (Covering Number)**.** The covering number for a metric space $(\mathcal{X}, d)$, N_ϵ, is the smallest M such that there exists an ϵ-cover of size M.

In our application, we consider the metric space of probability distributions from some class, say a mixture of gaussians, under total variation distance. We then define the metric entropy to be the logarithm of the covering number.

Definition 3 (Metric Entropy)**.** Let $(\mathcal{X}, d)$ be a metric space with covering number N_ϵ. The metric entropy is defined to be $\log N_\epsilon$.

3.3.2 The Metric Entropy Learning Algorithm

We now give the high level of the general learning algorithm. Suppose we know that we are trying to learn a distribution p from a space of distributions $\mathbb{P}$. Then we can use an ϵ-cover of N_ϵ different distributions covering $\mathbb{P}$. We plug all of these distributions into the General Scheffé's Estimator for p and get back a close distribution. This gives the following theorem.

Theorem 5. *Let p be an unknown distribution in a space $\mathbb{P}$. Let $\log N_\epsilon$ be the metric entropy of $\mathbb{P}$ under d_{TV}. There exists an algorithm using $n = \Omega(\frac{\log N_\epsilon}{\epsilon^2})$ samples that outputs $\hat{p}$ such that $d_{\mathrm{TV}}(\hat{p}, p) \leq \epsilon$.*

Note that the above theorem only gives good guarantees on space complexity. However, the algorithm as given requires N_ϵ^2 calls to Scheffé's Estimator given in Algorithm 1. This means that when N_ϵ is large, the algorithm isn't very practical. Unfortunately, this is the best known learning algorithm in many cases.

In general, there exists an ϵ-cover on distributions over k elements of size $O(\frac{1}{\epsilon^k})$, but this may be hard to find. By quantizing the probabilities on each element up to ϵ/k, we can give a cover of size $O((\frac{k}{\epsilon})^k)$. This implies an algorithm with space complexity $O(\frac{k \log(k/\epsilon)}{\epsilon^2})$ and time complexity $O((\frac{k}{\epsilon})^{2k})$. For space complexity, this is tight up to log factors. For time complexity, it is open to even prove $\mathrm{poly}(k, \frac{1}{\epsilon})$ lower bounds.

3.3.3 Application to Bernoulli Distributions

We'll conclude with a corollary of Theorem 5 applied to learning Bernoulli distributions.

Corollary 1. *There exists an algorithm using $O(\frac{1}{\epsilon^2} \log \frac{k}{\epsilon})$ samples for learning a Bernoulli distribution.*

Proof. Let $\mathbb{P}$ be the space of Bernoulli distributions. $\mathcal{C} = \{\mathrm{Bern}(\frac{\epsilon i}{k}) : i \in [0..k]\}$ is an ϵ-cover of $\mathbb{P}$ of size $\frac{k}{\epsilon} + 1$. Then the corollary follows immediately from Theorem 5. □

In essence, we've reduced the problem of proper learning to the problem of finding an ϵ-cover for the space our distribution can come from.

4 References

[DL01] L. Devroye and G. Lugosi, Choosing a Density Estimate, *Combinatorial Methods in Density Estimation* Springer New York, 2001. 47–57.