7

Gaussian Mixture Models

Many natural statistics, such as the distribution of people's heights, can be modeled as a mixture of Gaussians. The components of the mixture represent the parts of the distribution coming from different subpopulations. But if we don't know about the subpopulations in advance, can we figure out what they are and learn their parameters? And can we then classify samples based on which subpopulation they are likely to have come from? In this chapter we will give the first algorithms for learning the parameters of a mixture of Gaussians at an inverse polynomial rate. The one-dimensional case was introduced by Karl Pearson, who was one of the founders of statistics. We will show the first provable guarantees for his method. Building on this, we will solve the high-dimensional learning problem too. Along the way, we will develop insights about systems of polynomial equations and how they can be used for parameter learning.

7.1 Introduction

Karl Pearson was one of the luminaries of statistics and helped to lay its foundation. He introduced revolutionary new ideas and methods, such as:

(a) p-values, which are now the de facto way to measure statistical significance
(b) The chi-squared test, which measures goodness of fit to a Gaussian distribution
(c) Pearson's correlation coefficient
(d) The method of moments for estimating the parameters of a distribution
(e) Mixture models for modeling the presence of subpopulations

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Believe it or not, the last two were introduced in the same influential study from 1894 that represented Pearson's first foray into biometrics [120]. Let's understand what led Pearson down this road. While on vacation, his colleague Walter Weldon and his wife had meticulously collected 1,000 Naples crabs and measured 23 different physical attributes of each of them. But there was a surprise lurking in the data. All but one of these statistics was approximately Gaussian. So why weren't they all Gaussian?

Everyone was quite puzzled, until Pearson offered an explanation: *Maybe the Naples crab is not one species, but rather two species*. Then it is natural to model the observed distribution as a mixture of two Gaussians, rather than just one. Let's be more formal. Recall that the density function of a one-dimensional Gaussian with mean μ and variance σ^2 is

$$\mathcal{N}(\mu, \sigma^2, x) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{\frac{-(x-\mu)^2}{2\sigma^2}\right\}.$$

And for a mixture of two Gaussians, it is

$$F(x) = w_1 \underbrace{\mathcal{N}(\mu_1, \sigma_1^2, x)}_{F_1(x)} + (1 - w_1) \underbrace{\mathcal{N}(\mu_2, \sigma_2^2, x)}_{F_2(x)}.$$

We will use F_1 and F_2 to denote the two Gaussians in the mixture. You can also think of it in terms of how you'd generate a sample from it: Take a biased coin that is heads with probability w_1 and tails with the remaining probability $1-w_1$. Then for each sample you flip the coin; i.e., decide which subpopulation your sample comes from. If it's heads, you output a sample from the first Gaussian, otherwise you output a sample from the second one.

This is already a powerful and flexible statistical model (see Figure 7.1). But Pearson didn't stop there. He wanted to find the parameters of a mixture of two Gaussians that best fit the observed data to test out his hypothesis. When it's just one Gaussian, it's easy, because you can set μ and σ^2 to be the empirical mean and empirical variance, respectively. But what should you do when there are five unknown parameters and for each sample there is a hidden variable representing which subpopulation it came from? Pearson used the method of moments, which we will explain in the next subsection. The parameters he found seemed to be a good fit, but there were still a lot of unanswered questions, such as: Does the method of moments always find a good solution if there is one?

Method of Moments

Here we will explain how Pearson used the method of moments to find the unknown parameters. The key observation is that the moments of a mixture

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

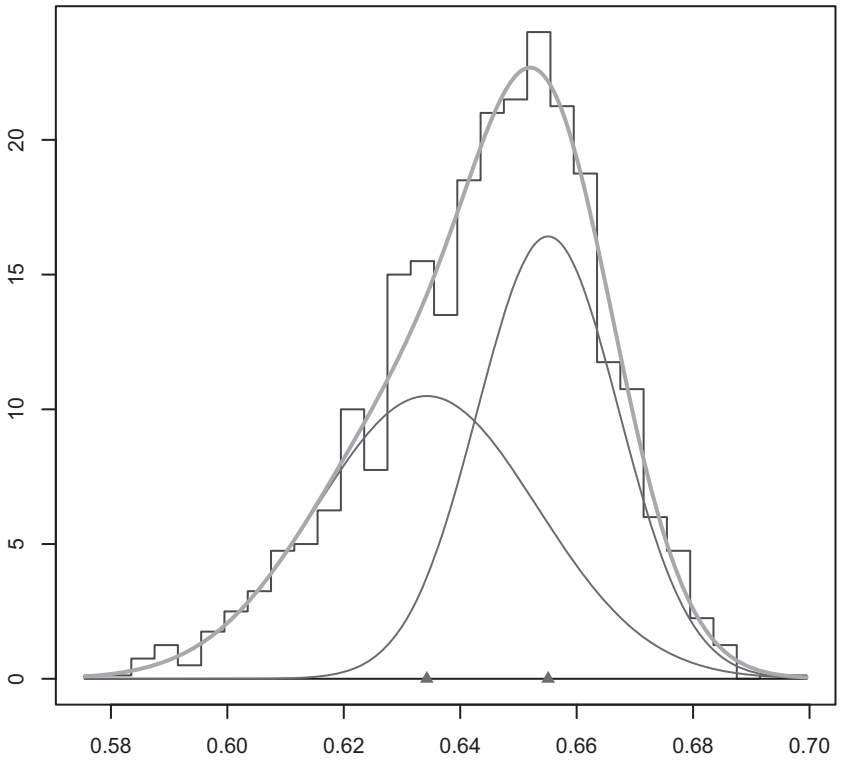

Figure 7.1: A fit of a mixture of two univariate Gaussians to Pearson's data on Naples crabs, created by Peter Macdonald using R.

of Gaussians are themselves polynomials in the unknown parameters. Let's denote the r^{th} raw moments of a Gaussian by M_r:

$$\mathop{\mathbb{E}}_{x \leftarrow F_1(x)} \left[x^r\right] = M_r(\mu, \sigma)$$

It is easy to compute $M_1(\mu, \sigma) = \mu$ and $M_2(\mu, \sigma) = \mu^2 + \sigma^2$, etc., and check that M_r is a degree r polynomial in μ and σ. Now we have

$$\mathop{\mathbb{E}}_{x \leftarrow F(x)} \left[x^r\right] = w_1 M_r(\mu_1, \sigma_1) + (1 - w_1) M_r(\mu_2, \sigma_2) = P_r(w_1, \mu_1, \sigma_1, \mu_2, \sigma_2).$$

And so the r^{th} raw moment of a mixture of two Gaussians is itself a degree $r + 1$ polynomial, which we denote by P_r, in the parameters we would like to learn.

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Pearson's Sixth Moment Test: We can estimate $\mathbb{E}_{x \leftarrow F}[x^r]$ from random samples. Let S be our set of samples. Then we can compute:

$$\widetilde{M}_r = \frac{1}{|S|} \sum_{x \in S} x^r$$

And given a polynomial number of samples (for any $r = O(1)$), $\widetilde{M}_r$ will be additively close to $\mathbb{E}_{x \leftarrow F(x)}[x^r]$. Pearson's approach was:

- Set up a system of polynomial equations

$$\left\{ P_r(w_1, \mu_1, \sigma_1, \mu_2, \sigma_2) = \widetilde{M}_r \right\}, \; r = 1, 2, \ldots, 5.$$

- Solve this system. Each solution is a setting of all five parameters that explains the first five empirical moments.

Pearson solved the above system of polynomial equations *by hand*, and he found a number of candidate solutions. Each solution corresponds to a way to set all five unknown parameters so that the moments of the mixture match the empirical moments. But how can we choose among these candidate solutions? Some of the solutions were clearly not right; some had negative values for the variance, or a value for the mixing weight that was not between zero and one. But even after eliminating these solutions, Pearson was still left with more than one candidate solution. His approach was to choose the candidate whose prediction was closest to the empirical sixth moment $\widetilde{M}_6$. This is called the *sixth moment test*.

Expectation Maximization

The workhorse in modern statistics is the *maximum likelihood estimator*, which sets the parameters so as to maximize the probability that the mixture would generate the observed samples. This estimator has lots of wonderful properties. Under certain technical conditions, it is *asymptotically efficient*, meaning that no other estimator can achieve asymptotically smaller variance as a function of the number of samples. Even the law of its distribution can be characterized, and is known to be normally distributed with a variance related to what's called the Fisher information. Unfortunately, for most of the problems we will be interested in, it is *NP*-hard to compute [19].

The popular alternative is known as *expectation maximization* and was introduced in an influential paper by Dempster, Laird, and Rubin [61]. It is important to realize that this is just a heuristic for computing the maximum likelihood estimator and does not inherit any of its statistical guarantees.

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Expectation maximization is a general approach for dealing with latent variables where we alternate between estimating the latent variables given our current set of parameters, and updating our parameters. In the case of mixtures of two Gaussians, it repeats the following until convergence:

- For each $x \in S$, calculate the posterior probability:

$$\widehat{w}_1(x) = \frac{\widehat{w}_1 \widehat{F}_1(x)}{\widehat{w}_1 \widehat{F}_1(x) + (1 - \widehat{w}_1)\widehat{F}_2(x)}$$

- Update the mixing weights:

$$\widehat{w}_1 \leftarrow \frac{\sum_{x \in S} \widehat{w}_1(x)}{|S|}$$

- Reestimate the parameters:

$$\widehat{\mu}_i \leftarrow \frac{\sum_{x \in S} \widehat{w}_i(x) x}{\sum_{x \in S} \widehat{w}_i(x)}, \ \widehat{\Sigma}_i \leftarrow \frac{\sum_{x \in S} \widehat{w}_i(x)(x - \widehat{\mu}_i)(x - \widehat{\mu}_i)^T}{\sum_{x \in S} \widehat{w}_i(x)}$$

In practice, it seems to work well. But it can get stuck in local maxima of the likelihood function. Even worse, it can be quite sensitive to how it is initialized (see, e.g., [125]).

7.2 Clustering-Based Algorithms

Our basic goal will be to give algorithms that provably compute the true parameters of a mixture of Gaussians, given a polynomial number of random samples. This question was introduced in the seminal paper of Dasgupta [56], and the first generation of algorithms focused on the high-dimensional case where the components are far enough apart that they have essentially no *overlap*. The next generation of algorithms are based on algebraic insights and avoid clustering altogether.

The High-Dimensional Geometry of Gaussians

Before we proceed, we will discuss some of the counterintuitive properties of high-dimensional Gaussians. First, the density of a multidimensional Gaussian in $\mathbb{R}^n$ is given by

$$\mathcal{N}(\mu, \Sigma) = \frac{1}{(2\pi)^{n/2} det(\Sigma)^{1/2}} exp\left\{ \frac{-(x - \mu)^\top \Sigma^{-1}(x - \mu)}{2} \right\}.$$

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Here, Σ is the covariance matrix. If $\Sigma = \sigma^2 I_n$ and $\mu = \vec{0}$, then the distribution is just $\mathcal{N}(0, \sigma^2) \times \mathcal{N}(0, \sigma^2) \times \ldots \times \mathcal{N}(0, \sigma^2)$ and we call it a spherical Gaussian, because the density function is rotationally invariant.

Fact 7.2.1 *The maximum value of the density function is at $x = \mu$.*

Fact 7.2.2 *For a spherical Gaussian, almost all the weight of the density function has $\|x - \mu\|_2^2 = \sigma^2 n \pm \sigma^2 \sqrt{n \log n}$.*

At first, these facts might seem to be inconsistent. The first one tells us that the most probable value of a sample is at zero. The second one tells us that almost all of the samples are far from zero. It's easiest to think about what's happening in spherical coordinates. The maximum of the density function is when the radius $R = 0$. But the rate at which the surface area of the sphere increases is much faster than the rate that the density function decreases, until we reach a radius of $R = \sigma\sqrt{n}$. Really, we should think about a high-dimensional spherical Gaussian as being essentially a thin spherical shell.

The Cluster-Then-Learn Paradigm

Clustering-based algorithms are all based on the following strategy:

- Cluster all of the samples S into two sets S_1 and S_2 depending on whether they were generated by the first or second component.
- Output the empirical mean and covariance of each S_i along with the empirical mixing weight $\frac{|S_1|}{|S|}$.

The details of how we will implement the first step and what types of conditions we need to impose will vary from algorithm to algorithm. But first let's see that if we could design a clustering algorithm that succeeds with high probability, the parameters we find would be provably good estimates for the true ones. This is captured by the following lemmas. Let $|S| = m$ be the number of samples.

Lemma 7.2.3 *If $m \geq C\frac{\log 1/\delta}{\epsilon^2}$ and clustering succeeds, then*

$$|\widehat{w}_1 - w_1| \leq \epsilon$$

with probability at least $1 - \delta$.

Now let $w_{min} = \min(w_1, 1 - w_1)$. Then

Lemma 7.2.4 *If $m \geq C\frac{n \log 1/\delta}{w_{min}\epsilon^2}$ and clustering succeeds, then*

$$\|\widehat{\mu}_i - \mu_i\|_2 \leq \epsilon$$

for each i, with probability at least $1 - \delta$.

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Finally, let's show that the empirical covariance is close too:

Lemma 7.2.5 *If* $m \geq C\frac{n \log 1/\delta}{w_{min}\epsilon^2}$ *and clustering succeeds, then*

$$\|\widehat{\Sigma}_i - \Sigma_i\| \leq \epsilon$$

for each i*, with probability at least* $1 - \delta$.

All of these lemmas can be proven via standard concentration bounds. The first two follow from concentration bounds for scalar random variables, and the third requires more high-powered matrix concentration bounds. However, it is easy to prove a version of this that has a worse but still polynomial dependence on n by proving that each entry of $\widehat{\Sigma}_i$ and Σ_i are close and using the union bound. What these lemmas together tell us is that if we really could solve clustering, then we would indeed be able to provably estimate the unknown parameters.

Dasgupta [56]: $\widetilde{\Omega}(\sqrt{n})$ Separation

Dasgupta gave the first provable algorithms for learning mixtures of Gaussians, and required that $\|\mu_i - \mu_j\|_2 \geq \widetilde{\Omega}(\sqrt{n}\sigma_{max})$ where σ_{max} is the maximum variance of any Gaussian in any direction (e.g., if the components are not spherical). Note that the constant in the separation depends on w_{min}, and we assume we know this parameter (or a lower bound on it).

The basic idea behind the algorithm is to project the mixture onto $\log k$ dimensions uniformly at random. This projection will preserve distances between each pair of centers μ_i and μ_j with high probability, but will contract distances between samples from the same component and make each component closer to spherical, thus making it easier to cluster. Informally, we can think of this separation condition as: if we think of each Gaussian as a spherical ball, then if the components are far enough apart, these balls will be *disjoint*.

Arora and Kannan [19] and Dasgupta and Schulman [64]: $\widetilde{\Omega}(n^{1/4})$ Separation

We will describe the approach in [19] in detail. The basic question is, if $\sqrt{n}$ separation is the threshold where we can think of the components as disjoint, how can we learn when the components are much closer? In fact, even if the components are only $\widetilde{\Omega}(n^{1/4})$ separated, it is still true that *every* pair of samples from the same component is closer than *every* pair of samples from different components. How can this be? The explanation is that even though the balls representing each component are no longer disjoint, we are still very unlikely to sample from their overlap region.

Consider $x, x' \leftarrow F_1$, and $y \leftarrow F_2$.

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Claim 7.2.6 *All of the vectors $x - \mu_1$, $x' - \mu_1$, $\mu_1 - \mu_2$, $y - \mu_2$ are nearly orthogonal (whp).*

This claim is immediate, since the vectors $x - \mu_1, x' - \mu_1, y - \mu_2$ are uniform from a sphere, and $\mu_1 - \mu_2$ is the only fixed vector. In fact, any set of vectors in which all but one are uniformly random from a sphere are nearly orthogonal.

Now we can compute:

$$\begin{aligned}\|x - x'\|^2 &\approx \|x - \mu_1\|^2 + \|\mu_1 - x'\|^2 \\ &\approx 2n\sigma^2 \pm 2\sigma^2\sqrt{n \log n}\end{aligned}$$

And similarly:

$$\begin{aligned}\|x - y\|^2 &\approx \|x - \mu_1\|^2 + \|\mu_1 - \mu_2\|^2 + \|\mu_2 - y\|^2 \\ &\approx 2n\sigma^2 + \|\mu_1 - \mu_2\|^2 \pm 2\sigma^2\sqrt{n \log n}\end{aligned}$$

Hence if $\|\mu_1 - \mu_2\| = \widetilde{\Omega}(n^{1/4}, \sigma)$, then $\|\mu_1 - \mu_2\|^2$ is larger than the error term and each pair of samples from the same component will be closer than each pair from different components. Indeed, we can find the right threshold τ and correctly cluster all of the samples. Again, we can output the empirical mean, empirical covariance, and relative size of each cluster, and these will be good estimates of the true parameters.

Vempala and Wang [141]: $\widetilde{\Omega}(k^{1/4})$ Separation

Vempala and Wang [141] removed the dependence on n and replaced it with a separation condition that depends on k, the number of components. The idea is that if we could project the mixture into the subspace T spanned by $\{\mu_1, \ldots, \mu_k\}$, we would preserve the separation between each pair of components but reduce the ambient dimension.

So how can we find T, the subspace spanned by the means? We will restrict our discussion to a mixture of spherical Gaussians with a common variance $\sigma^2 I$. Let $x \sim F$ be a random sample from the mixture; then we can write $x = c + z$ where $z \sim N(0, \sigma^2 I_n)$ and c is a random vector that takes the value μ_i with probability w_i for each $i \in [k]$. So:

$$\mathbb{E}[xx^T] = \mathbb{E}[cc^T] + \mathbb{E}[zz^T] = \sum_{i=1}^{k} w_i \mu_i \mu_i^\top + \sigma^2 I_n$$

Hence the top left singular vectors of $\mathbb{E}[xx^T]$, whose singular value is strictly larger than σ^2, exactly span T. We can then estimate $\mathbb{E}[xx^T]$ from sufficiently many random samples, compute its singular value decomposition, and project the mixture onto T and invoke the algorithm of [19].

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Brubaker and Vempala [40]: Separating Hyperplane

What if the largest variance of any component is much larger than the separation between the components? Brubaker and Vempala [40] observed that none of the existing algorithms succeed for a mixture that looks like a pair of *parallel pancakes*. In this example, there is a hyperplane that separates the mixture so that almost all of one component is on one side and almost all of the other component is on the other side. They gave an algorithm that succeeds, provided that such a separating hyperplane exists; however, the conditions are more complex to state for mixtures of three or more Gaussians. With three components, it is easy to construct mixtures that we can hope to learn, but where there are no hyperplanes that separate one component from the others.

7.3 Discussion of Density Estimation

The algorithms we have discussed so far all rely on clustering. But there are some cases where this strategy just won't work, because clustering is information theoretically impossible. More precisely, we will show that if $d_{TV}(F_1, F_2) = 1/2$, then we will quickly encounter a sample where we cannot figure out which component generated it, even if we know the true parameters.

Let's formalize this through the notion of a coupling:

Definition 7.3.1 *A coupling between F and G is a distribution on pairs (x, y) so that the marginal distribution on x is F and the marginal distribution on y is G. The error is the probability that $x \neq y$.*

So what is the error of the best coupling? It is easy to see that it is exactly the total variation distance:

Claim 7.3.2 *There is a coupling with error ε between F and G if and only if $d_{TV}(F, G) \leq \varepsilon$.*

In fact, this is a nice way to think about the total variation distance. Operationally upper-bounding the total variation distance tells us there is a good coupling. In a similar manner, you can interpret the KL divergence as the penalty you pay (in terms of expected coding length) when you optimally encode samples from one distribution using the best code for the other.

Returning to the problem of clustering the samples from a mixture of two Gaussians, suppose we have $d_{TV}(F_1, F_2) = 1/2$ and that

$$F(x) + 1/2F_1(x) + 1/2F_2(x).$$

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Using the above claim, we know that there is a coupling between F_1 and F_2 that agrees with probability $1/2$. Hence, instead of thinking about sampling from a mixture of two Gaussians in the usual way (choose which component, then choose a random sample from it), we can alternatively sample as follows:

1. Choose (x, y) from the best coupling between F_1 and F_2.
2. If $x = y$, output x with probability $1/2$, and otherwise output y.
3. Else output x with probability $1/2$, and otherwise output y.

This procedure generates a random sample from F just as before. What's important is that if you reach the second step, the value you output doesn't depend on which component the sample came from. So you can't predict it better than randomly guessing. This is a useful way to think about the assumptions that clustering-based algorithms make. Some are stronger than others, but at the very least they need to take at least n samples and cluster all of them correctly. In order for this to be possible, we must have

$$d_{TV}(F_1, F_2) \geq 1 - 1/n.$$

But who says that algorithms for learning must first cluster? Can we hope to learn the parameters even when the components almost entirely overlap, such as when $d_{TV}(F_1, F_2) = 1/n$?

Now is a good time to discuss the types of goals we could aim for and how they relate to each other.

(a) **Improper Density Estimation**

This is the weakest learning goal. If we're given samples from some distribution F in some class $\mathcal{C}$ (e.g., $\mathcal{C}$ could be all mixtures of two Gaussians), then we want to find any other distribution $\widehat{F}$ that satisfies $d_{TV}(F, \widehat{F}) \leq \varepsilon$. We do not require $\widehat{F}$ to be in class $\mathcal{C}$ too. What's important to know about improper density estimation is that in one dimension it's easy. You can solve it using a kernel density estimate, provided that F is smooth.

Here's how kernel density estimates work. First you take many samples and construct an empirical point mass distribution G. Now, G is not close to F. It's not even smooth, so how can it be? But you can fix this by convolving with a Gaussian with small variance. In particular, if you set $\widehat{F} = G * \mathcal{N}(0, \sigma^2)$ and choose the parameters and number of samples appropriately, what you get will satisfy $d_{TV}(F, \widehat{F}) \leq \varepsilon$ with high probability. This scheme doesn't use much about the distribution F, but it pays the price in high dimensions. The issue is that you just won't get enough samples that are close to each other. In general, kernel density estimates need the number of samples to be exponential in the dimension in order to work.

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

(b) **Proper Density Estimation**

Proper density estimation is the same but stronger, in that it requires $\widehat{F} \in \mathcal{C}$. Sometimes you can interpolate between improper and proper density estimation by constraining $\widehat{F}$ to be in some larger class that contains $\mathcal{C}$. It's also worth noting that sometimes you can just take a kernel density estimate or anything else that solves the improper density estimation problem and look for the $\widehat{F} \in \mathcal{C}$ that is closest to your improper estimate. This would definitely work, but the trouble is that algorithmically, it's usually not clear how to find the closest distribution in some class to some other unwieldy target distribution. Finally, we reach the strongest type of goal:

(c) **Parameter Learning**

Here we not only require that $d_{TV}(F, \widehat{F}) \leq \varepsilon$ and that $\widehat{F} \in \mathcal{C}$, but we want $\widehat{F}$ to be a good estimate for F *on a component-by-component basis*. For example, our goal specialized to the case of mixtures of two Gaussians is:

Definition 7.3.3 *We will say that a mixture $\widehat{F} = \widehat{w}_1\widehat{F}_1 + \widehat{w}_2\widehat{F}_2$ is ε-close (on a component-by-component basis) to F if there is a permutation $\pi : \{1, 2\} \to \{1, 2\}$ so that for all $i \in \{1, 2\}$*

$$\left| w_i - \widehat{w}_{\pi(i)} \right|, d_{TV}(F_i, \widehat{F}_{\pi(i)}) \leq \varepsilon.$$

Note that F and $\widehat{F}$ must necessarily be close as mixtures too: $d_{TV}(F, \widehat{F}) \leq 4\varepsilon$. However, we can have mixtures F and $\widehat{F}$ that are both mixtures of k Gaussians and are close as distributions, but are not close on a component-by-component basis. So why should we aim for such a challenging goal? It turns out that if $\widehat{F}$ is ε-close to F, then given a typical sample, we can estimate the posterior accurately [94]. What this means is that even if you can't cluster all of your samples into which component they came from, you can still figure out which ones it's possible to be confident about. This is one of the main advantages of parameter learning over some of the weaker learning goals.

It's good to achieve the strongest types of learning goals you can hope for, but you should also remember that lower bounds for these strong learning goals (e.g., parameter learning) do not imply lower bounds for weaker problems (e.g., proper density estimation). We will give algorithms for learning the parameters of a mixture of k Gaussians that run in polynomial time for any $k = O(1)$ but have an exponential dependence on k. But this is necessary, in that there are pairs of mixtures of k Gaussians F and $\widehat{F}$ that are not close on a component-by-component basis but have $d_{TV}(F, \widehat{F}) \leq 2^{-k}$ [114]. So any algorithm for parameter learning would be able to tell them apart, but that

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

takes at least 2^k samples, again by a coupling argument. But maybe for proper density estimation it's possible to get an algorithm that is polynomial in all of the parameters.

Open Question 1 *Is there a* $\text{poly}(n, k, 1/\varepsilon)$ *time algorithm for proper density estimation for mixtures of k Gaussians in n dimensions? What about in one dimension?*

7.4 Clustering-Free Algorithms

Our goal is to learn $\widehat{F}$ that is ε-close to F. Let's first generalize the definition to mixtures of k Gaussians:

Definition 7.4.1 *We will say that a mixture* $\widehat{F} = \sum_{i=1}^{k} \widehat{w}_i \widehat{F}_i$ *is* ε*-close (on a component-by-component basis) to* F *if there is a permutation* $\pi : \{1, 2, \ldots, k\} \to \{1, 2, \ldots, k\}$ *so that for all* $i \in \{1, 2, \ldots, k\}$,

$$\left| w_i - \widehat{w}_{\pi(i)} \right|, d_{TV}(F_i, \widehat{F}_{\pi(i)}) \leq \varepsilon.$$

When can we hope to learn an ε close estimate in $\text{poly}(n, 1/\varepsilon)$ samples? There are two situations where it just isn't possible. Eventually our algorithm will show that these are the only things that go wrong:

(a) If $w_i = 0$, we can never learn $\widehat{F}_i$ that is close to F_i, because we never get any samples from F_i.

In fact, we need a quantitative lower bound on each w_i, say $w_i \geq \varepsilon$, so that if we take a reasonable number of samples, we will get at least one sample from each component.

(b) If $d_{TV}(F_i, F_j) = 0$, we can never learn w_i or w_j, because F_i and F_j entirely overlap.

Again, we need a quantitative lower bound on $d_{TV}(F_i, F_j)$, say $d_{TV}(F_i, F_j) \geq \varepsilon$, for each $i \neq j$ so that if we take a reasonable number of samples, we will get at least one sample from the nonoverlap region between various pairs of components.

Theorem 7.4.2 *[94], [114] If* $w_i \geq \varepsilon$ *for each* i *and* $d_{TV}(F_i, F_j) \geq \varepsilon$ *for each* $i \neq j$*, then there is an efficient algorithm that learns an* ε*-close estimate* $\widehat{F}$ *to* F *whose running time and sample complexity are* $\text{poly}(n, 1/\varepsilon, \log 1/\delta)$ *and that succeeds with probability* $1 - \delta$.

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Note that the degree of the polynomial depends polynomially on k. Kalai, Moitra, and Valiant [94] gave the first algorithm for learning mixtures of two Gaussians with no separation conditions. Subsequently, Moitra and Valiant [114] gave an algorithm for mixtures of k Gaussians, again with no separation conditions.

In independent and concurrent work, Belkin and Sinha [28] gave a polynomial time algorithm for mixtures of k Gaussians too; however, there is no explicit bound given on the running time as a function of k (since their work depends on Hilbert's basis theorem, which is fundamentally ineffective). Also, the goal in [94] and [114] is to learn $\widehat{F}$ so that its components are close in total variation distance to those of F, which is in general a stronger goal than requiring that the parameters be additively close, which is the goal in [28]. The benefit is that the algorithm in [28] works for more general learning problems in the one-dimensional setting, and we will explain the ideas of their algorithm at the end of this chapter.

Throughout this section we will focus on the $k = 2$ case, since this algorithm is conceptually much simpler. In fact, we will aim for a weaker learning goal: We will say that $\widehat{F}$ is *additively* ε-close to F if $|w_i - \widehat{w}_{\pi(i)}|, \|\mu_i - \widehat{\mu}_{\pi(i)}\|, \|\Sigma_i - \widehat{\Sigma}_{\pi(i)}\|_F \leq \varepsilon$ for all i. We want to find such an $\widehat{F}$. It turns out that we will be able to assume that F is normalized in the following sense:

Definition 7.4.3 *A distribution F is in isotropic position if*

(a) $\mathbb{E}_{x \leftarrow F}[x] = 0$ *and*
(b) $\mathbb{E}_{x \leftarrow F}[xx^T] = I$.

The second condition means that the variance is one in *every* direction. Actually, it's easy to put a distribution in isotropic position, provided that there's no direction where the variance is zero. More precisely:

Claim 7.4.4 *If $\mathbb{E}_{x \leftarrow F}[xx^T]$ is full rank, then there is an affine transformation that places F in isotropic position.*

Proof: Let $\mu = E_{x \leftarrow F}[x]$. Then

$$E_{x \leftarrow F}[(x - \mu)(x - \mu)^T] = M = BB^T$$

which follows because M is positive semidefinite and hence has a Cholesky decomposition. By assumption, M has full rank, and hence B does too. Now if we set

$$y = B^{-1}(x - \mu)$$

it is easy to see that $\mathbb{E}[y] = 0$ and $\mathbb{E}[yy^T] = B^{-1}M(B^{-1})^T = I$ as desired. ■

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Our goal is to learn an additive ε approximation to F, and we will assume that F has been preprocessed so that it is in isotropic position.

Outline

We can now describe the basic outline of the algorithm, although there will be many details to fill in:

(a) Consider a series of projections down to one dimension.
(b) Run a univariate learning algorithm.
(c) Set up a system of linear equations on the high-dimensional parameters and back-solve.

Isotropic Projection Lemma

We will need to overcome a number of obstacles to realize this plan, but let's work through the details of this outline. First let's understand what happens to the parameters of a Gaussian when we project it along some direction r:

Claim 7.4.5 $proj_r[\mathcal{N}(\mu, \Sigma)] = \mathcal{N}(r^T\mu, r^T\Sigma r)$

This simple claim already tells us something important: Suppose we want to learn the parameters μ and Σ of a high-dimensional Gaussian. If we project it onto direction r and learn the parameters of the resulting one-dimensional Gaussian, then what we've really learned are linear constraints on μ and Σ. If we do this many times for many different directions r, we could hope to get enough linear constraints on μ and Σ that we could simply solve for them. Moreover, it's natural to hope that we need only about n^2 directions, because there are that many parameters of Σ. But now we're coming up to the first problem we'll need to find a way around. Let's introduce some notation:

Definition 7.4.6 $d_p(\mathcal{N}(\mu_1, \sigma_1^2), \mathcal{N}(\mu_2, \sigma_2^2)) = |\mu_1 - \mu_2| + |\sigma_1^2 - \sigma_2^2|$

We will refer to this as the parameter distance. Ultimately, we will give a univariate algorithm for learning mixtures of Gaussians, and we would like to run it on $\text{proj}_r[F]$.

Problem 2 *But what if* $d_p(proj_r[F_1], proj_r[F_2])$ *is exponentially small?*

This would be a problem, since we would need to run our univariate algorithm with exponentially fine precision just to see that there are two components and not one! How can we get around this issue? We'll prove that this problem essentially never arises when F is in isotropic position. For intuition, consider two cases:

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

(a) Suppose $\|\mu_1 - \mu_2\| \geq \text{poly}(1/n, \varepsilon)$.

You can think of this condition as just saying that $\|\mu_1 - \mu_2\|$ is not exponentially small. In any case, we know that projecting a vector onto a random direction typically reduces its norm by a factor of $\sqrt{n}$ and that its projected length is concentrated around this value. This tells us that with high probability $\|r^T\mu_1 - r^T\mu_2\|$ is at least $\text{poly}(1/n, \varepsilon)$ too. Hence $\text{proj}_r[F_1]$ and $\text{proj}_r[F_2]$ will have noticeably different parameters just due to the difference in their means.

(b) Otherwise, $\|\mu_1 - \mu_2\| \leq \text{poly}(1/n, \varepsilon)$.

The key idea is that if $d_{TV}(F_1, F_2) \geq \varepsilon$ and their means are exponentially close, then their covariances Σ_1 and Σ_2 must be noticeably different when projected on a random direction r. In this case, $\text{proj}_r[F_1]$ and $\text{proj}_r[F_2]$ will have noticeably different parameters due to the difference in their variances. This is the intuition behind the following lemma:

Lemma 7.4.7 *If F is in isotropic position and $w_i \geq \varepsilon$ and $d_{TV}(F_1, F_2) \geq \varepsilon$, then with high probability for a direction r chosen uniformly at random*

$$d_p(proj_r[F_1], proj_r[F_2]) \geq \varepsilon_3 = \text{poly}(1/n, \varepsilon).$$

This lemma is false when F is not in isotropic position (e.g., consider the parallel pancakes example)! It also fails when generalizing to mixtures of $k > 2$ Gaussians even when the mixture is in isotropic position. What goes wrong is that there are examples where projecting onto almost all directions r essentially results in a mixture with strictly fewer components! (The approach in [114] is to learn a mixture of fewer Gaussians as a proxy for the true mixture, and later on find a direction that can be used to separate out pairs of components that have been merged.)

Pairing Lemma

Next we will encounter the second problem: Suppose we project onto direction r and s and learn $\widehat{F}^r = \frac{1}{2}\widehat{F}_1^r + \frac{1}{2}\widehat{F}_2^r$ and $\widehat{F}^s = \frac{1}{2}\widehat{F}_1^s + \frac{1}{2}\widehat{F}_2^s$, respectively. Then the mean and variance of $\widehat{F}_1^r$ yield a linear constraint on one of the two high-dimensional Gaussians, and similarly for $\widehat{F}_1^s$.

Problem 3 *How do we know that they yield constraints on the* same *high-dimensional component?*

Ultimately we want to set up a system of linear constraints to solve for the parameters of F_1, but when we project F onto different directions (say,

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

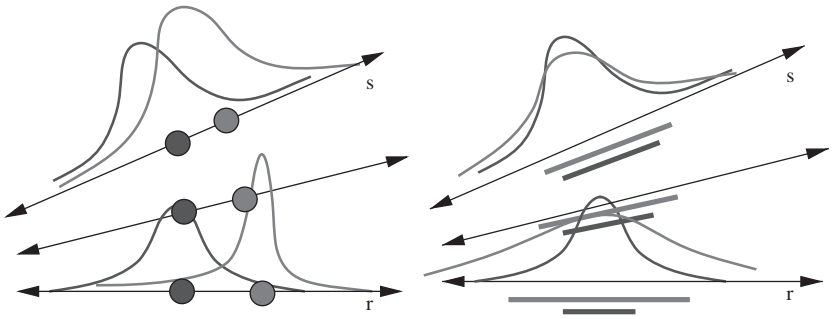

Figure 7.2: The projected mean and projected variance vary continuously as we sweep from r to s.

r and s), we need to pair up the components from these two directions. The key observation is that as we vary r to s, the parameters of the mixture vary continuously. (See Figure 7.2). Hence when we project onto r, we know from the isotropic projection lemma that the two components will have either noticeably different means or variances. Suppose their means are different by ε_3; then if r and s are close (compared to ε_1), the parameters of each component in the mixture do not change much and the component in $\text{proj}_r[F]$ with larger mean will correspond to the same component as the one in $\text{proj}_s[F]$ with larger mean. A similar statement applies when it is the variances that are at least ε_3 apart.

Lemma 7.4.8 *If* $\|r - s\| \leq \varepsilon_2 = \text{poly}(1/n, \varepsilon_3)$*, then:*

(a) If $|r^T \mu_1 - r^T \mu_2| \geq \varepsilon_3$*, then the components in* $proj_r[F]$ *and* $proj_s[F]$ *with the larger mean correspond to the same high-dimensional component.*

(b) Else if $|r^T \Sigma_1 r - r^T \Sigma_2 r| \geq \varepsilon_3$*, then the components in* $proj_r[F]$ *and* $proj_s[F]$ *with the larger variance correspond to the same high-dimensional component.*

Hence if we choose r randomly and only search over directions s with $\|r - s\| \leq \varepsilon_2$, we will be able to pair up the components correctly in the different one-dimensional mixtures.

Condition Number Lemma

Now we encounter the final problem in the high-dimensional case: Suppose we choose r randomly, and for $s_1, s_2, \ldots, s_p$ we learn the parameters of the

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

projection of F onto these directions and pair up the components correctly. We can only hope to learn the parameters on these projections up to some additive accuracy ε_1 (and our univariate learning algorithm will have running time and sample complexity $\text{poly}(1/\varepsilon_1)$).

Problem 4 *How do these errors in our univariate estimates translate to errors in our high-dimensional estimates for $\mu_1, \Sigma_1, \mu_2, \Sigma_2$?*

Recall that the *condition number* controls this. The final lemma we need in the high-dimensional case is:

Lemma 7.4.9 *The condition number of the linear system to solve for μ_1, Σ_1 is $\text{poly}(1/\varepsilon_2, n)$ where all pairs of directions are ε_2 apart.*

Intuitively, as r and $s_1, s_2, \ldots, s_p$ are closer together, the condition number of the system will be worse (because the linear constraints are closer to redundant), but the key fact is that the condition number is bounded by a fixed polynomial in $1/\varepsilon_2$ and n, and hence if we choose $\varepsilon_1 = \text{poly}(\varepsilon_2, n)\varepsilon$, then our estimates of the high-dimensional parameters will be within an additive ε. Note that each parameter $\varepsilon, \varepsilon_3, \varepsilon_2, \varepsilon_1$ is a fixed polynomial in the earlier parameters (and $1/n$), and hence we need only run our univariate learning algorithm with inverse polynomial precision on a polynomial number of mixtures to learn an ε-close estimate $\widehat{F}$!

But we still need to design a univariate algorithm, and next we return to Pearson's original problem!

7.5 A Univariate Algorithm

Here we will give a univariate algorithm for learning the parameters of a mixture of two Gaussians up to additive accuracy ε whose running time and sample complexity is $\text{poly}(1/\varepsilon)$. Our first observation is that all of the parameters are bounded:

Claim 7.5.1 *Let $F = w_1F_1 + w_2F_2$ be a mixture of two Gaussians that is in isotropic position. Suppose that $w_1, w_2 \geq \varepsilon$. Then*

(a) $\mu_1, \mu_2 \in [-1/\sqrt{\varepsilon}, 1/\sqrt{\varepsilon}]$ *and*
(b) $\sigma_1^2, \sigma_2^2 \in [0, 1/\varepsilon]$.

The idea is that if either of the conditions is violated, it would imply that the mixture has variance strictly larger than one. Once we know that the parameters are bounded, the natural approach is to try a grid search:

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Grid Search

Input: Samples from $F(\Theta)$
Output: Parameters $\widehat{\Theta} = (\widehat{w}_1, \widehat{\mu}_1, \widehat{\sigma}_1^2, \widehat{\mu}_2, \widehat{\sigma}_2^2)$

For all valid $\widehat{\Theta}$ where the parameters are multiples of ε^C
    Test $\widehat{\Theta}$ using the samples, if it passes output $\widehat{\Theta}$
End

There are many ways we could think about testing the closeness of our estimate with the true parameters of the model. For example, we could empirically estiamte the first six moments of $F(\Theta)$ from our samples, and pass $\widehat{\Theta}$ if its first six moments are each within some additive tolerance τ of the empirical moments. (This is really a variant on Pearson's sixth moment test.) It is easy to see that if we take enough samples and set τ appropriately, then if we round the true parameters Θ to any valid grid point whose parameters are multiples of ε^C, the resulting $\widehat{\Theta}$ will with high probability pass our test. This is called the *completeness*. The much more challenging part is establishing the *soundness*; after all, why is there no other set of parameters $\widehat{\Theta}$ except for ones close to Θ that pass our test?

Alternatively, we want to prove that any two mixtures F and $\widehat{F}$ whose parameters *do not* match within an additive ε must have one of their first six moments noticeably different. The main lemma is:

Lemma 7.5.2 (Six Moments Suffice) *For any F and $\widehat{F}$ that are not ε-close in parameters, there is an $r \in \{1, 2, \ldots, 6\}$ where*

$$\left| M_r(\Theta) - M_r(\widehat{\Theta}) \right| \geq \varepsilon^{O(1)}$$

where Θ and $\widehat{\Theta}$ are the parameters of F and $\widehat{F}$, respectively, and M_r is the r^{th} raw moment.

Let $\widetilde{M}_r$ be the empirical moments. Then

$$\left| M_r(\widehat{\Theta}) - M_r(\Theta) \right| \leq \underbrace{\left| \widetilde{M}_r(\widehat{\Theta}) - \widetilde{M}_r \right|}_{\leq \tau} + \underbrace{\left| \widetilde{M}_r - M_r(\Theta) \right|}_{\leq \tau} \leq 2\tau$$

where the first term is at most τ because the test passes, and the second term is small because we can take enough samples (but still poly$(1/\tau)$) so that the empirical moments and the true moments are close. Hence we can apply the above lemma in the contrapositive, and conclude that if the grid search outputs

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

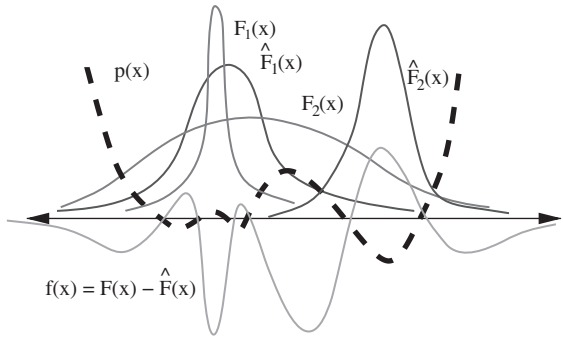

Figure 7.3: If $f(x)$ has at most six zero crossings, we can find a polynomial of degree at most six that agrees with its sign.

$\widehat{\Theta}$, then Θ and $\widehat{\Theta}$ must be ε-close in parameters, which gives us an efficient univariate algorithm!

So our main goal is to prove that if F and $\widehat{F}$ are not ε-close, one of their first six moments is noticeably different. In fact, even the case of $\varepsilon = 0$ is challenging: If F and $\widehat{F}$ are different mixtures of two Gaussians, why is one of their first six moments necessarily different? Our main goal is to prove this statement using the *heat equation*.

In fact, let us consider the following thought experiment. Let $f(x) = F(x) - \widehat{F}(x)$ be the pointwise difference between the density functions F and $\widehat{F}$. Then the heart of the problem is: Can we prove that $f(x)$ crosses the x-axis at most six times? (See Figure 7.3.)

Lemma 7.5.3 *If $f(x)$ crosses the x-axis at most six times, then one of the first six moments of F and $\widehat{F}$ is different.*

Proof: In fact, we can construct a (nonzero) degree at most six polynomial $p(x)$ that agrees with the sign of $f(x)$; i.e., $p(x)f(x) \geq 0$ for all x. Then

$$0 < \left| \int_x p(x)f(x)dx \right| = \left| \int_x \sum_{r=1}^{6} p_r x^r f(x)dx \right|$$
$$\leq \sum_{r=1}^{6} |p_r| \left| M_r(\Theta) - M_r(\widehat{\Theta}) \right|.$$

And if the first six moments of F and $\widehat{F}$ match exactly, the right-hand side is zero, which is a contradiction. ■

So all we need to prove is that $F(x) - \widehat{F}(x)$ has at most six zero crossings. Let us prove a stronger lemma by induction:

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Lemma 7.5.4 *Let $f(x) = \sum_{i=1}^{k} \alpha_i \mathcal{N}(\mu_i, \sigma_i^2, x)$ be a linear combination of k Gaussians (α_i can be negative). Then if $f(x)$ is not identically zero, $f(x)$ has at most $2k - 2$ zero crossings.*

We will rely on the following tools:

Theorem 7.5.5 *Given $f(x) : \mathbb{R} \to \mathbb{R}$ that is analytic and has n zero crossings, then for any $\sigma^2 > 0$, the function $g(x) = f(x) * \mathcal{N}(0, \sigma^2)$ has at most n zero crossings.*

This theorem has a physical interpretation. If we think of $f(x)$ as the heat profile of an infinite one-dimensional rod, then what does the heat profile look like at some later time? In fact, it is precisely $g(x) = f(x) * \mathcal{N}(0, \sigma^2)$ for an appropriately chosen σ^2. Alternatively, the Gaussian is the *Green's function* of the heat equation. And hence many of our physical intuitions for diffusion have consequences for convolution – convolving a function by a Gaussian has the effect of smoothing it, and it cannot create new local maxima (and relatedly, it cannot create new zero crossings).

Finally, we recall the elementary fact:

Fact 7.5.6 $\mathcal{N}(0, \sigma_1^2) * \mathcal{N}(0, \sigma_2^2) = \mathcal{N}(0, \sigma_1^2 + \sigma_2^2)$

Now, we are ready to prove the above lemma and conclude that if we knew the first six moments of a mixture of two Gaussians, *exactly*, then we would know its parameters exactly too. Let us prove the above lemma by induction, and assume that for any linear combination of $k = 3$ Gaussians, the number of zero crossings is at most four. Now consider an arbitrary linear combination of four Gaussians, and let σ^2 be the smallest variance of any component. (See Figure 7.4a.) We can consider a related mixture where we subtract σ^2 from the variance of each component. (See Figure 7.4b.)

Now, if we ignore the delta function, we have a linear combination of three Gaussians, and by induction we know that it has at most four zero crossings. But how many zero crossings can we add when we add back in the delta function? We can add at most two, one on the way up and one on the way down (here we are ignoring some real analysis complications of working with delta functions for ease of presentation). (See Figure 7.4c.) And now we can convolve the function by $\mathcal{N}(0, \sigma^2)$ to recover the original linear combination of four Gaussians, but this last step does not increase the number of zero crossings! (See Figure 7.4d.)

This proves that

$$\left\{ M_r(\widehat{\Theta}) = M_r(\Theta) \right\}, \; r = 1, 2, \ldots, 6$$

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

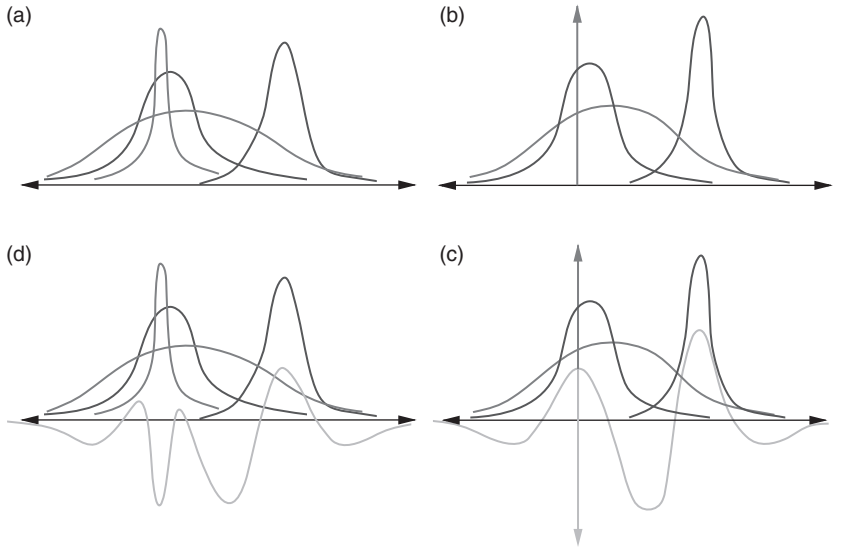

Figure 7.4: (a) Linear combination of four Gaussians; (b) subtracting σ^2 from each variance; (c) adding back in the delta function; (d) convolving by $\mathcal{N}(0, \sigma^2)$ to recover the original linear combination.

has only two solutions (the true parameters, and we can also interchange which is component is which). In fact, this system of polynomial equations is also *stable*, and there is an analogue of condition numbers for systems of polynomial equations that implies a quantitative version of what we have just proved: if F and $\widehat{F}$ are not ε-close, then one of their first six moments is noticeably different. This gives us our univariate algorithm.

7.6 A View from Algebraic Geometry

Here we will present an alternative univariate learning algorithm of Belkin and Sinha [28] that also makes use of the method of moments, but gives a much more general analysis using tools from algebraic geometry.

Polynomial Families

We will analyze the method of moments for the following class of distributions:

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Definition 7.6.1 *A class of distributions $F(\Theta)$ is called a polynomial family if*

$$\forall r,\ \mathbb{E}_{X\in F(\Theta)}\left[X^r\right] = M_r(\Theta)$$

where $M_r(\Theta)$ is a polynomial in $\Theta = (\theta_1, \theta_2, \ldots, \theta_k)$.

This definition captures a broad class of distributions, such as mixture models whose components are uniform, exponential, Poisson, Gaussian, or gamma functions. We will need another (tame) condition on the distribution that guarantees it is characterized by all of its moments.

Definition 7.6.2 *The moment-generating function (mgf) of a random variable X is defined as*

$$f(t) = \sum_{n=0}^{\infty} \mathbb{E}\left[X^n\right] \frac{t^n}{n!}.$$

Fact 7.6.3 *If the moment-generating function of X converges in a neighborhood of zero, it uniquely determines the probability distribution; i.e.,*

$$\forall r,\ M_r(\Theta) = M_r(\widehat{\Theta}) \implies F(\Theta) = F(\widehat{\Theta}).$$

Our goal is to show that for any polynomial family, a *finite* number of its moments suffice. First we introduce the relevant definitions:

Definition 7.6.4 *Given a ring R, an ideal I generated by $g_1, g_2, \cdots, g_n \in R$ denoted by $I = \langle g_1, g_2, \cdots, g_n \rangle$ is defined as*

$$I = \left\{ \sum_i r_i g_i \text{ where } r_i \in R \right\}.$$

Definition 7.6.5 *A Noetherian ring is a ring such that for any sequence of ideals*

$$I_1 \subseteq I_2 \subseteq I_3 \subseteq \cdots,$$

there is N such that $I_N = I_{N+1} = I_{N+2} = \cdots$.

Theorem 7.6.6 *[Hilbert's Basis Theorem] If R is a Noetherian ring, then $R[X]$ is also a Noetherian ring.*

It is easy to see that $\mathbb{R}$ is a Noetherian ring, and hence we know that $\mathbb{R}[x]$ is also Noetherian. Now we can prove that for any polynomial family, a finite number of moments suffice to uniquely identify any distribution in the family:

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

Theorem 7.6.7 *Let $F(\Theta)$ be a polynomial family. If the moment-generating function converges in a neighborhood of zero, there exists N such that*

$$F(\Theta) = F(\widehat{\Theta}) \text{ if and only if } M_r(\Theta) = M_r(\widehat{\Theta}) \; \forall r \in 1, 2, \cdots, N.$$

Proof: Let $Q_r(\Theta, \widehat{\Theta}) = M_r(\Theta) - M_r(\widehat{\Theta})$. Let $I_1 = \langle Q_1 \rangle$, $I_2 = \langle Q_1, Q_2 \rangle, \cdots$. This is our ascending chain of ideals in $\mathbb{R}[\Theta, \widehat{\Theta}]$. We can invoke Hilbert's basis theorem and conclude that $\mathbb{R}[X]$ is a Noetherian ring, and hence there is N such that $I_N = I_{N+1} = \cdots$. So for all $N + j$, we have

$$Q_{N+j}(\Theta, \widehat{\Theta}) = \sum_{i=1}^{N} p_{ij}(\Theta, \widehat{\Theta}) Q_i(\Theta, \widehat{\Theta})$$

for some polynomial $p_{ij} \in \mathbb{R}[\Theta, \widehat{\Theta}]$. Thus, if $M_r(\Theta) = M_r(\widehat{\Theta})$ for all $r \in 1, 2, \cdots, N$, then $M_r(\Theta) = M_r(\widehat{\Theta})$ for all r, and from Fact 7.6.3 we conclude that $F(\Theta) = F(\widehat{\Theta})$.

The other side of the theorem is obvious. ■

The theorem above does not give any finite bound on N, since the basis theorem does not either. This is because the basis theorem is proved by contradiction, but more fundamentally, it is not possible to give a bound on N that depends only on the choice of the ring. Consider the following example:

Example 2 *Consider the Noetherian ring $\mathbb{R}[x]$. Let $I_i = \langle x^{N-i} \rangle$ for $i = 0, \cdots, N$. It is a strictly ascending chain of ideals for $i = 0, \cdots, N$. Therefore, even if the ring $\mathbb{R}[x]$ is fixed, there is no universal bound on N.*

Bounds such as those in Theorem 7.6.7 are often referred to as *ineffective*. Consider an application of the above result to mixtures of Gaussians: from the above theorem, we have that any two mixtures F and $\widehat{F}$ of k Gaussians are identical if and only if these mixtures agree on their first N moments. Here N is a function of k and N is finite, but we cannot write down any explicit bound on N as a function of k using the above tools. Nevertheless, these tools apply much more broadly than the specialized ones based on the heat equation that we used in the previous section to prove that $4k - 2$ moments suffice for mixtures of k Gaussians.

Systems of Polynomial Inequalities

In general, we do not have exact access to the moments of a distribution, but only noisy approximations. Our main goal is to prove a quantitative version of the previous result that shows that any two distributions F and $\widehat{F}$ that are close on their first N moments are close in their parameters too. The key fact is

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

that we can bound the condition number of systems of polynomial inequalities; there are a number of ways to do this, but we will use *quantifier elimination*. Recall:

Definition 7.6.8 *A set S is semialgebraic if there exist multivariate polynomials $p_1, \ldots, p_n$ such that*

$$S = \{x_1, \ldots, x_r | p_i(x_1, \ldots, x_r) \geq 0\}$$

or if S is a finite union or intersection of such sets.

When a set can be defined through polynomial equalities, we call it *algebraic*.

Theorem 7.6.9 *[Tarski] The projection of a semialgebraic set is semialgebraic.*

Interestingly, the projection of an algebraic set is not necessarily algebraic. Can you come up with an example? A projection corresponds to defining a set not just through polynomial inequalities, but also a $\exists$ operator. It turns out that you can even take a sequence of $\exists$ and $\forall$ operators and the resulting set is still semialgebraic.

With this tool in hand, we define the following helper set:

$$H(\varepsilon, \delta) = \left\{ \forall(\Theta, \widehat{\Theta}) \ : \ |M_r(\Theta) - M_r(\widehat{\Theta})| \leq \delta \text{ for } r = 1, 2, \ldots N \Longrightarrow d_p(\Theta, \widehat{\Theta}) \leq \varepsilon \right\}$$

Here $d_p(\Theta, \widehat{\Theta})$ is some parameter distance between Θ and $\widehat{\Theta}$. It is not important exactly what we choose, just that it can be expressed through polynomials in the parameters and that it treats parameters that produce the same distribution as the same; e.g., by taking the minimum over all matchings of components in $F(\Theta)$ to components in $F(\widehat{\Theta})$ and summing the componentwise parameter distances.

Now let $\varepsilon(\delta)$ be the smallest ε as a function of δ. Using Tarski's theorem, we can prove the following stability bound for the method of moments:

Theorem 7.6.10 *There are fixed constants C_1, C_2, s such that if $\delta \leq 1/C_1$, then $\varepsilon(\delta) \leq C_2 \delta^{1/s}$.*

Proof: It is easy to see that we can define $H(\varepsilon, \delta)$ as the projection of a semialgebraic set, hence using Tarski's theorem, we conclude that $H(\varepsilon, \delta)$ is also semialgebraic. The crucial observation is that because $H(\varepsilon, \delta)$ is semialgebraic, the smallest we can choose ε to be as a function of δ is itself a polynomial function of δ. There are some caveats here, because we need to prove that for a fixed δ we can choose ε to be strictly greater than zero

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press

and, moreover, the polynomial relationship between ε and δ only holds if δ is sufficiently small. However, these technical issues can be resolved without much more work (see [28]). ■

Now we arrive at the main result:

Corollary 7.6.11 *If* $|M_r(\Theta) - M_r(\widehat{\Theta})| \leq \left(\frac{\varepsilon}{C_2}\right)^s$, *then* $d_p(\Theta, \widehat{\Theta}) \leq \varepsilon$.

Hence there is a polynomial time algorithm to learn the parameters of any univariate polynomial family (whose mgf converges in a neighborhood of zero) within an additive accuracy of ε whose running time and sample complexity is poly$(1/\varepsilon)$; we can take enough samples to estimate the first N moments within ε^s and search over a grid of the parameters, and any set of parameters that matches each of the moments is necessarily close in parameter distance to the true parameters.

7.7 Exercises

Problem 7-1: Suppose we are given a mixture of two Gaussians where the variances of each component are equal:

$$F(x) = w_1 \mathcal{N}(\mu_1, \sigma^2, x) + (1 - w_1)\mathcal{N}(\mu_2, \sigma^2, x)$$

Show that four moments suffice to uniquely determine the parameters of the mixture.

Problem 7-2: Suppose we are given access to an oracle that, for any direction r, returns the projected means and variances; i.e., $r^T \mu_1$ and $r^T \Sigma_1 r$ for one component and $r^T \mu_2$ and $r^T \Sigma_2 r$. The trouble is that you do not know which parameters correspond to which component.

(a) Design an algorithm to recover μ_1 and μ_2 (up to permuting which component is which) that makes at most $O(d^2)$ queries to the oracle where d is the dimension. *Hint:* Recover the entries of $(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$.

(b) **Challenge:** Design an algorithm to recover Σ_1 and Σ_2 (up to permuting which component is which) that makes $O(1)$ queries to the oracle when $d = 2$.

Note that here we are not assuming anything about how far apart the projected means or variances are on some direction r.

https://doi.org/10.1017/9781316882177.008 Published online by Cambridge University Press