Bootstrapping EM via Power EM and Convergence in the Naive Bayes Model

Constantinos Daskalakis
EECS and CSAIL, MIT

Christos Tzamos
Microsoft Research

Manolis Zampetakis
EECS and CSAIL, MIT

Abstract

We study the convergence properties of the Expectation-Maximization algorithm in the Naive Bayes model. We show that EM can get stuck in regions of slow convergence, even when the features are binary and i.i.d. conditioning on the class label, and even under random (i.e. non worst-case) initialization. In turn, we show that EM can be bootstrapped in a pre-training step that computes a good initialization. From this initialization we show theoretically and experimentally that EM converges exponentially fast to the true model parameters. Our bootstrapping method amounts to running the EM algorithm on appropriately centered iterates of small magnitude, which as we show corresponds to effectively performing power iteration on the covariance matrix of the mixture model, although power iteration is performed under the hood by EM itself. As such, we call our bootstrapping approach "power EM." Specifically for the case of two binary features, we show global exponentially fast convergence of EM, even without bootstrapping. Finally, as the Naive Bayes model is quite expressive, we show as corollaries of our convergence results that the EM algorithm globally converges to the true model parameters for mixtures of two Gaussians, recovering recent results of [XHM16, DTZ17].

1 Introduction

The *Expectation-Maximization (EM)* algorithm is one of the most widely used heuristics for maximum likelihood estimation in statistical models with latent variables. Since its introduction by statisticians four decades ago [DLR77, Wu83, RW84], it has found myriad applications with more than 50k citations(!) in Google scholar. Despite its wide use and applicability, we have very limited understanding of its convergence to the maximum likelihood estimate (MLE). Only local convergence guarantees are generally known [Wu83, Tse04, CH08, BWY14], except for some very recent work establishing its global convergence for the case of balanced mixtures of two Gaussian distributions with known covariance matrices [XHM16, DTZ17].

The heart of the challenge in analyzing the convergence of the algorithm to the true MLE stems from the fact that statistical estimation in the presence of latent variables commonly results in non-convex likelihood landscapes. The EM algorithm provides a widely applicable method for navigating such non-convex landscapes,[1] but it can get stuck in local optima or regions of slow convergence, as we discuss shortly. In this light, understanding the convergence properties of the EM algorithm falls under the general theme of understanding non-convex optimization methods in inference and estimation, which have been the focus of renewed interest in recent years; e.g. [NIP16].

In this paper, we study the convergence of the Expectation-Maximization algorithm for one of the most common models where it is used, *the Naive Bayes model*. Introduced in the 1950s the Naive Bayes model has been studied extensively for classification tasks and is competitive to more advanced methods with appropriate preprocessing; see e.g. [RST+03]. We will consider the two-class naive Bayes model of Figure 1, whose latent (a.k.a. class) variable C takes two values 1 and 2, and whose observable (a.k.a. feature) variables $X_1, \ldots, X_n$ take values in some finite set $K = \{1, \ldots, k\}$.[2]

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

[1] For completeness, we provide a detailed description of the algorithm in Appendix A.

[2] Of course, it is not important that $X_1, \ldots, X_n$ all take values in the same set $\{1, \ldots, k\}$. We can always take K to be the union of the domains of individual features and

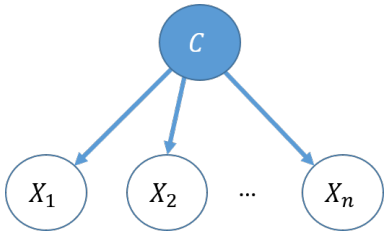

Figure 1: The Naive Bayes Classifier

Throughout the paper we will assume for simplicity that C takes values 1 and 2 uniformly at random, i.e., that the two classes are equally likely. We call such models *balanced*. As the graphical representation of the model implies, conditioning on the value of the class label C, the variables $X_1, \ldots, X_n$, representing the values of different features, are independent.

Our goal is to understand whether the Expectation-Maximization (EM) algorithm of Dempster et al. [DLR77, Wu83] is able to identify the unknown parameters of the model, i.e. the distribution of each feature, conditioning on each possible value of the class variable C. As we have already discussed, the difficulty in analyzing EM stems from the non-convex likelihood landscape that it is trying to navigate. To isolate the complexity arising from the non-convexity of the likelihood landscape from the errors arising from sampling, recent work on EM has focused on the so-called *population* version of the EM algorithm [BWY14, XHM16, DTZ17], where we assume access to infinitely many samples from the mixture distribution. Studying the population version of the algorithm disentangles the study of whether EM is able to navigate the non-convex likelihood landscape from sampling errors. Moreover, as exhibited by recent work, understanding the behavior of the population version can often be leveraged to also bound its error rate in the finite sample regime. Accordingly, we start our investigation in this paper with studying the convergence properties of the population version of the EM algorithm in the Naive Bayes model. We then extend our results to obtain error rates in the finite sample regime. We also perform experimental evaluation of our results, as discussed below.

Our Results. Let us first anchor our expectations about the behavior of EM in the Naive Bayes model. In Section 6, we perform simulations showing that the algorithm may get stuck in regions with very slow convergence, even when the features are binary and i.i.d. conditioning on the class label, but the EM algorithm is agnostic of this symmetry in the features. In fact, as we show, this happens with large probability even from random initialization. Our simulations are discussed in Section 6 and Figures 7–9, although the notation of Section 2.1 is needed to understand the discussion and the figure captions.

Given our simulation results we cannot hope for global and fast convergence of the EM algorithm in the Naive Bayes model. So what we do next is (1) identify models in which global and exponentially fast convergence of the EM algorithm holds; (2) identify models where global and exponentially fast convergence can be guaranteed under a good initialization; and (3) identify ways to perform a good initialization.

Here are our findings on the first front.

Theorem 1 (Global Convergence 1)**.** *Consider samples from a balanced two-class Naive Bayes model with 2 binary features. Then population EM converges exponentially fast to maximum likelihood parameters from any initialization. Moreover, the true model parameters are non-identifiable in this case and, in particular, EM converges to a curve of solutions containing the true model parameters.*

Theorem 2 (Global Convergence 2)**.** *Consider samples from a balanced two-class Naive Bayes model with n binary features that are i.i.d. conditioning on the class label. Suppose that EM is cognizant of the symmetry of the features. Then population EM converges exponentially fast to the true model parameters from any initialization.*

Theorem 1 appears as Theorem 7 in Section 3, and Theorem 2 appears as Theorem 8 in Section 4. The notation of Section 2.1 is needed to understand the statements of the latter theorems, but they are restatements of Theorems 1 and 2. All statements apply to the population EM iteration (2.2) derived in Section 2.1.

Exploiting the expressive power of the Naive Bayes model, we derive as corollary of Theorem 2 that the population EM algorithm exhibits exponentially fast and global convergence to the true model parameters for balanced mixtures of two single-dimensional Gaussians, recovering recent results of [XHM16, DTZ17].

Theorem 3 (Global Convergence 3)**.** *Consider samples from a balanced mixture of two single-dimensional Gaussians with known variances. Then population EM converges exponentially fast to the true model parameters from any initialization.*

As we have already discussed, in the setting of Theorem 2, if EM is agnostic to the symmetry of the features, then it may get stuck in regions of very slow convergence. What we study next is whether there exists a canonical initialization of the EM algorithm that lands it in a region of fast convergence. We pro-

rename its elements so that the resulting set is $\{1, \ldots, k\}$.

pose a particular kind of initialization which amounts to bootstrapping EM itself. In particular, we propose to run EM on appropriately centered iterates of small magnitude. In this regime, we show that EM iterations effectively perform power iterations on the covariance matrix of the mixture model.

Theorem 4 (PowerEM)**.** *Consider samples from a balanced two-class Naive Bayes model with n features taking k values. In particular, let us view the Naive Bayes model as sampling vectors $\boldsymbol{x} \in \{0,1\}^{k\cdot n}$, where $\boldsymbol{x}_{ij} = 1$ iff the i-th feature takes the j-th value. As the magnitude of the EM iterate tends to 0, the iteration of population EM converges to the power iteration on the covariance matrix of the distribution.*

See Theorem 10 for a more detailed description of our result. Both statements apply to the population EM iteration 2.1 as derived in Section 2. We call performing EM iterations in the small-magnitude-iterate regime "PowerEM."

Our proposed initialization procedure is to run PowerEM (i.e. run EM with small magnitude iterates) for a few steps, then blow up the magnitude of the resulting iterate and continue with EM iterations. We call this algorithm "Power Pretrained EM," and we detail it in Section 7.1. We show the following.

Theorem 5 (Exponentially Fast Convergence of Power Pretrained EM)**.** *Consider samples from a balanced two-class Naive Bayes model with n binary features that are i.i.d. conditioning on the class label. Suppose that EM is ignorant of the symmetry of the features. Then Power Pretrained EM converges exponentially fast to the true model parameters.*

See Theorem 11 for a more detailed statement. Recall that without the PowerEM iterations, EM would get stuck in regions of slow convergence, as shown by our simulation results discussed earlier. While our theoretical guarantees (of Theorem 5) that the Power Pretrained EM algorithm converges to the true model parameters holds for the case of symmetric features, we perform simulations to study the superiority of Power Pretrained EM in broader settings. Our simulation results are discussed in Sections 7. For example, Figure 13 shows how the Power Pretrained EM (orange graph) performs significantly better compared to randomly initialized EM (blue graph), with the quality of the Power Pretrained EM improving with the number of steps of Power EM performed (left to right).

Finally, while our theoretical results stated above pertain to the population EM algorithm, we can exploit our analysis to obtain finite sample statements.

Theorem 6 (Finite Samples)**.** *In the settings of Theorems 1–5, the error rate of the finite sample EM algorithm is $O(n/\sqrt{N})$, where N is the number of samples.*

2 Preliminaries

We derive the update rule of the population EM algorithm for the two-class Naive Bayes model described in Section 1. We will assume, without loss of generality, that the marginal distribution of each feature is uniform over K. If this is not the case, we can—agnostically with respect to the parameters of the model that we do not know—process the samples we receive to make the feature marginals uniform over K in a way that we also know a one-to-one correspondence between the parameters of the resulting Naive Bayes model and the original Naive Bayes model. This is explained in Appendix B. With this assumption, we can parametrize the Naive Bayes model with n vectors $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_n \in [-1,1]^k$ satisfying $\boldsymbol{\mu}_i \cdot \mathbf{1} = 0$ so that the distribution of feature i conditioning on the class label being 1 and 2 respectively is

$$\frac{\mathbf{1} + \boldsymbol{\mu}_i}{k} \quad \text{and} \quad \frac{\mathbf{1} - \boldsymbol{\mu}_i}{k}.$$

For a Naive Bayes model parametrized by a vector $\boldsymbol{\mu}$, we will denote by $p_{\boldsymbol{\mu}}$ its probability distribution. We will view $p_{\boldsymbol{\mu}}$ as a distribution sampling vectors $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \in \{0,1\}^{k\cdot n}$. Each $\boldsymbol{x}_i$ is a vector in $\{0,1\}^k$ representing the state of feature i; in particular, $x_{ij} = 1$ iff feature i takes value j.

The population EM update for the Naive Bayes model is presented in Lemma 1, whose proof is provided in Appendix A.1.

Lemma 1 (Population EM update)**.** *If $\boldsymbol{\lambda}^{(t)} = \left(\boldsymbol{\lambda}_1^{(t)}, \ldots, \boldsymbol{\lambda}_n^{(t)}\right)$ is the current estimate of the parameters $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_n)$ of the Naive Bayes model, the new estimate $\boldsymbol{\lambda}^{(t+1)}$ after one iteration of the population EM algorithm is*

$$\boldsymbol{\lambda}^{(t+1)} = k \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\boldsymbol{\mu}}} \left[\tanh\left(\tanh^{-1}(\boldsymbol{\lambda}^{(t)}) \cdot \boldsymbol{x} \right) \left(\boldsymbol{x} - \mathbf{1}\frac{1}{k} \right) \right] \tag{2.1}$$

where $\tanh^{-1}(\boldsymbol{\lambda}^{(t)})$ denotes the vector resulting from pointwise application of $\tanh^{-1}$ on the coordinates of $\boldsymbol{\lambda}^{(t)}$, i.e. $\tanh^{-1}(\boldsymbol{\lambda}^{(t)}) = \left(\tanh^{-1}\left(\lambda_{ij}^{(t)}\right)\right)_{ij}$.

2.1 The Binary Features Case

When the features are binary ($k = 2$), we can simplify our notation somewhat. First, we can shrink $\boldsymbol{x}$ to a n-dimensional vector as, if we know that feature i does not take value 1 (respectively 2), we can deduce that it takes value 2 (respectively 1). In fact, for convenience we will think of $\boldsymbol{x}$ as a ± 1 vector (rather than a 0/1 vector that we have used in previous sections), where

$x_i = 1$ iff feature i takes value 1. Under our uniform feature marginals assumption, we can similarly shrink vector $\boldsymbol{\mu}$ to a n-dimensional vector in $[-1,1]^n$, with the understanding that $\frac{1}{2}(1+\mu_i)$ is the probability that feature i takes value 1, conditioning on the class label being 1, and $\frac{1}{2}(1-\mu_i)$ is the probability that feature i takes value 1, conditioning on the class label being 2.

With these simplifications in notation, we will think of $p_{\boldsymbol{\mu}}$ as a distribution supported on $\{\pm 1\}^n$ and parametrized by $\boldsymbol{\mu} \in [-1,1]^n$. We will also denote by $d_{\boldsymbol{\mu}}$ the conditional of $p_{\boldsymbol{\mu}}$, conditioning on the class label being 1. This is a product measure over $\{-1,1\}^n$ where the distribution of the ith feature has mean μ_i. It is easy to see that the conditional of $p_{\boldsymbol{\mu}}$, conditioning on the class label being -1, is $d_{-\boldsymbol{\mu}}$, and

$$p_{\boldsymbol{\mu}} = \frac{1}{2} d_{\boldsymbol{\mu}} + \frac{1}{2} d_{-\boldsymbol{\mu}}.$$

The population version of the expectation-maximization algorithm in this setting maintains a guess $\boldsymbol{\lambda} \in [-1,1]^n$ of the parameter $\boldsymbol{\mu}$.

Lemma 2 (Naive Bayes EM with Binary Features)**.** *Consider the population EM algorithm for the Naive Bayes model with n binary features described above. If $\boldsymbol{\lambda}^{(t)} = \left(\lambda_1^{(t)}, \ldots, \lambda_n^{(t)}\right)$ is the current estimate of the parameters $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)$ of the Naive Bayes model, the update that the population EM algorithm will perform is:*

$$\boldsymbol{\lambda}^{(t+1)} = \mathop{\mathbb{E}}_{\boldsymbol{x} \sim d_{\boldsymbol{\mu}}} \left[\tanh\left(\tanh^{-1}(\boldsymbol{\lambda}^{(t)}) \cdot \boldsymbol{x} \right) \boldsymbol{x} \right] \qquad (2.2)$$

where $\tanh^{-1}(\boldsymbol{\lambda}^{(t)})$ denotes the vector resulting from pointwise application of $\tanh^{-1}$ on the coordinates of $\boldsymbol{\lambda}^{(t)}$, i.e. $\tanh^{-1}(\boldsymbol{\lambda}^{(t)}) = \left(\tanh^{-1}\left(\lambda_i^{(t)}\right)\right)_i$.

The proof of Lemma 2 appears in Appendix A.2.

Remark. Note both the similarity and the dissimilarity of Eq. (2.2) and our iteration for non-binary features of Eq (2.1). In both cases the iteration of the algorithm is compactly expressible in terms of the $\tanh(\cdot)$ and its inverse. On the other hand, the binary case allows for a further simplification seen in Eq (2.1), where the update is expressed as an expectation with respect to one of the two conditionals of the mixture. It is interesting to observe the similarity of this expression with the population EM iteration applied to mixture of two Gaussians with known covariances [DTZ17, XHM16]. We will explore this connection more in Section 5.

3 Two Binary Features Convergence

In this section, we study the convergence of the EM iteration (2.2) as presented in Lemma 2 for the case of two binary features ($n = 2$). Interestingly, we show that in this case it is information theoretically impossible to estimate the true parameters $\boldsymbol{\mu} = (\mu_1, \mu_2)$ exactly. Instead, any vector of parameters $\boldsymbol{\mu}' = (\mu_1', \mu_2')$ that satisfies $\mu_1'\mu_2' = \mu_1\mu_2$ is a valid maximum likelihood solution for the estimation task. We prove that the EM iteration (2.2) converges geometrically to such a maximum likelihood solution $\boldsymbol{\lambda}^*$ with $\lambda_1^*\lambda_2^* = \mu_1\mu_2$.

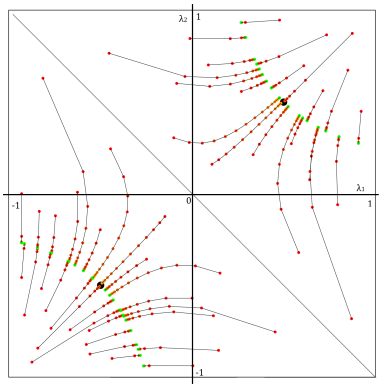

Figure 2: EM paths $(\lambda_1^{(0)}, \lambda_2^{(0)}) \to \ldots \to (\lambda_1^{(t)}, \lambda_2^{(t)})$ under different initializations for $\boldsymbol{\mu} = (1/2, 1/2)$. The iterations converge to the curve(s) $\lambda_1\lambda_2 = 1/4$.

Theorem 7. *Consider the population EM algorithm for the Naive Bayes model with two binary features. Let $\boldsymbol{\lambda}^{(t)} = \left(\lambda_1^{(t)}, \lambda_2^{(t)}\right)$ be the estimate of the parameters $\boldsymbol{\mu} = (\mu_1, \mu_2)$ of the Naive Bayes model at step t. We have that*

$$\left| \lambda_1^{(t+1)} \lambda_2^{(t+1)} - \mu_1\mu_2 \right| \le \sqrt{1 - \mu_1\mu_2} \left| \lambda_1^{(t)} \lambda_2^{(t)} - \mu_1\mu_2 \right|.$$

Moreover, any estimate $\boldsymbol{\lambda}$ such that $\lambda_1\lambda_2 = \mu_1\mu_2$ has the same likelihood and hence it is information theoretically impossible to distinguish between them.

We present the proof of Theorem 7 in Appendix C. Figure 2 illustrates the behavior of the EM algorithm in a typical scenario with two binary features.

4 Many i.i.d Features Convergence

We now focus on a case with many different features that are all independently and identically distributed (i.i.d), i.e. all the means μ_i are equal to μ. Starting from an initial guess $\boldsymbol{\lambda}^{(0)} = (\lambda^{(0)}, ..., \lambda^{(0)})$ with equal coordinates, the EM iteration of Lemma 2 can be writ-

ten as follows

$$\lambda_i^{(t+1)} = M_i(\lambda^{(t)}, \boldsymbol{\mu}) = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim d_{\boldsymbol{\mu}}}\left[\tanh\left(\tanh^{-1}(\lambda^{(t)})\left(\sum_{j=1}^{n} x_j\right)\right) x_i\right].$$

Since for all i, j, $\lambda_i^{(t)} = \lambda_j^{(t)}$ we get that $\lambda_i^{(t+1)} = \lambda_j^{(t+1)}$ as well. Figure 5 in Appendix J shows how the iteration function looks like together with its fixed points for $n = 5$ and $\mu = 1/2$.

We now show that $M_i(\cdot, \mu)$ is an increasing function. Since $M_i(\lambda^{(t)}, \mu) = M_j(\lambda^{(t)}, \mu)$, we can write

$$\lambda_i^{(t+1)} = M_i(\lambda^{(t)}, \boldsymbol{\mu}) = \frac{1}{n}\sum_{j=1}^{n} M_j(\lambda^{(t)}, \boldsymbol{\mu}) = \frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{x}\sim d_{\boldsymbol{\mu}}}\left[\tanh\left(\tanh^{-1}(\lambda^{(t)})\sum_{j=1}^{n} x_j\right)\sum_{j=1}^{n} x_j\right]$$

which implies that

$$\frac{dM_i}{d\lambda}(\lambda^{(t)}, \boldsymbol{\mu}) = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim d_{\boldsymbol{\mu}}}\Bigg[\tanh'\Big(\tanh^{-1}(\lambda^{(t)})\sum_{j=1}^{n} x_j\Big) \cdot \tanh^{-1'}(\lambda^{(t)})\Big(\sum_{j=1}^{n} x_j\Big)^2\Bigg].$$

Since $\tanh'(\cdot)$ and $\tanh^{-1'}(\cdot)$ are both positive functions, we get that M_i is an increasing function of λ.

Now we proceed to bound the convergence rate of the EM iteration. To derive the convergence rate bound we follow the *sensitivity method* developed in [DTZ17] and we get the following theorem.

Theorem 8. *Consider the population EM algorithm for the Naive Bayes model with $n > 2$ i.i.d features, where $\mu = \mu_i = \mu_j > 0$. When initialized at a point $\boldsymbol{\lambda}^{(0)}$ such that $\lambda^{(0)} = \lambda_i^{(0)} = \lambda_j^{(0)}$ with $\lambda^{(0)} > 0$, the parameters $\lambda^{(t)}$ satisfy*

$$\left|\lambda^{(t+1)} - \mu\right| \le \kappa^{(t)}\left|\lambda^{(t)} - \mu\right|,$$

where

$$\kappa^{(t)} = \left(1 - \min(\lambda^{(t)}, \mu)^2\right)^{\frac{n-2}{2}}.$$

Moreover $\kappa^{(t)}$ is a decreasing function of t. In particular $\kappa^{(t)} \le \kappa^{(0)}$ and hence the above relation implies geometric convergence of EM in this case.

The proof of Theorem 8 can be found in Appendix D.

Figure 6 in Appendix J shows experimentally how the distance $\left|\lambda^{(t)} - \mu\right|$ evolves during the EM iterations for $n = 5$, $\mu = 1/2$ and $\lambda^{(0)} = 1/10$. The convergence speed matches the geometric convergence guaranteed by Theorem 8.

5 Application to Mixture of Single Dimensional Gaussians

In the case of mixture of two Gaussians with known variances and unknown means, the latent variable C corresponds to the class number and it takes values 1 and 2 as in the Naive Bayes case as we defined in Section 2. The observable random variable X takes real values and the distribution of X conditioning on the class label being 1 and 2 respectively is μ and $-\mu$. Therefore the mixture distribution of the observable variable X is

$$p_\mu(x) = 0.5 \cdot \mathcal{N}(x; \mu, \sigma^2) + 0.5 \cdot \mathcal{N}(x; -\mu, \sigma^2),$$

where $\mathcal{N}(x; \mu, \sigma^2)$ is the density at point x of the normal distribution with mean μ and variance σ^2. We assume that we know the variance parameter σ^2. For this case [DTZ17] have shown that the EM update takes the following form (Equation (3.1) [DTZ17])

$$\lambda^{(t+1)} = M(\lambda^{(t)}, \mu) = \mathbb{E}_{x\sim\mathcal{N}(\mu,\sigma^2)}\left[\tanh\left(\frac{\lambda^{(t)}x}{\sigma^2}\right)x\right]. \tag{5.3}$$

In this section we reprove the convergence theorems of [XHM16, DTZ17], (1D version of Theorem 1 [XHM16], Theorem 1 [DTZ17]), using as main tool the statement of Theorem 8. This way we illustrate the expressive power of the Naive Bayes model and we show that the convergence analysis of EM in this case can serve as a tool for analyzing the convergence properties of EM in other models that can even involve continuous distributions. More precisely the theorem that we prove is the following.

Theorem 9. *In the single dimensional case of balanced mixture of two Gaussian distributions with means μ and $-\mu$ when $\lambda^{(0)}, \mu > 0$, the parameters $\lambda^{(t)}$ defined by (5.3) satisfy*

$$\left|\lambda^{(t+1)} - \mu\right| \le \kappa^{(t)}\left|\lambda^{(t)} - \mu\right|$$

where

$$\kappa^{(t)} = \exp\left(-\frac{\min(\lambda^{(t)}, \mu)^2}{2\sigma^2}\right)$$

Moreover $\kappa^{(t)}$ is a decreasing function of t. In particular $\kappa^{(t)} \le \kappa^{(0)}$ and hence the above relation implies geometric convergence of EM in this case.

We present the proof of Theorem 9 in Appendix G.

Remark. The reduction that is used in the proof can be generalized to mixture of Gaussians in higher dimensions following a similar path. Hence the more general convergence properties of EM in the Naive Bayes model can be used to prove corresponding convergence properties of EM for the case of multidimensional Gaussians.

6 Cases of Slow Convergence

In the previous section, we considered the scenario where both the mean vector and our initial guess have identical coordinates, i.e. $\mu_i = \mu_j$ and $\lambda_i^{(0)} = \lambda_j^{(0)}$. We saw that in this case the EM algorithm converges to the true parameters $\boldsymbol{\mu}$ geometrically and one would expect that such geometric convergence generalizes to other cases where coordinates may not be equal.

Unfortunately, this is not true as we show in this Section. We provide a counter-example where the convergence of EM is extremely slow even when all means are identical, i.e. $\mu_i = \mu_j$ and the initial vector $\lambda^{(0)}$ is such that all coordinates but one are equal. Figure 7 in Appendix J shows the execution of EM algorithm for $n = 5$, $\boldsymbol{\mu} = (1/10, 1/10, 1/10, 1/10, 1/10)$ and different initial guesses of the form $(\lambda_1, \lambda_2, \lambda_2, \lambda_2, \lambda_2)$.

Such slow convergence cannot be avoided simply by random initialization. Figure 8 presented in Appendix J, shows that even with random initialization there is high probability that the EM algorithm will reach a region of slow convergence even when all μ_i's are equal. In the more typical case (a), the algorithm seems to get stuck in local optima and not move from those at all. Even in the less typical scenario that the execution of the algorithm behaves like case (b) we can see that after a couple of iterations the convergence to the fixed point still becomes slow. This illustrates that the EM algorithm might not converge very quickly when initialized with a guess that belongs to some bad region of the space. Moreover, the size of this bad region is big enough so that with random initialization we end up in this region with high probability.

To further argue about the size of that region. We run in parallel 100 EM executions with random initialization and we plot the distribution of the distances of $\boldsymbol{\lambda}^{(t)}$ from $\boldsymbol{\mu}$ for several time steps t. The results are shown in Figure 9 presented in Appendix J.

As we can see in Figure 9 there is a small decrease on the distance from the optimum, when we go from step $t = 4$ to the step $t = 10$ as shown in part (a) of the figure. But as we can see in part (b) of the figure the improvement is very small from step $t = 10$ to step $t = 20$ and most of the mass of histogram has not been moved at all.

In the next section, we show how to properly initialize the EM algorithm to achieve fast convergence.

7 Pre-Training EM via "Power EM"

As we saw in Section 6 the Expectation-Maximization algorithm may get stuck in regions of very slow convergence even in very simple cases of Naive Bayes models. One of our main contributions in this paper is proposing a canonical initialization procedure that brings EM to a region of fast convergence. We do this by bootstrapping EM itself to identify a good initial point. We show that running EM starting from a point of small norm brings the algorithm to a subspace where convergence is fast. In particular, we show that, in the general Naive Bayes model that we are studying, the EM iteration (2.1) is in fact, asymptotically as the norm of the iterate $\boldsymbol{\lambda}^{(t)}$ goes to 0, a power iteration on the covariance matrix of the distribution. We show how this can serve as a crucial initialization procedure identifying a good subspace on which to run EM, avoiding regions of slow convergence. We call the pretraining approach of running EM with a small-norm-iterate "Power EM."

Theorem 10 (Power EM)**.** *Consider the population EM update of equation* (2.1) *for the Naive Bayes model. For notational convenience denote by* $\boldsymbol{\lambda}$ *and* $\boldsymbol{\lambda}'$ *the iterates before and after the update. Then:*

$$\boldsymbol{\lambda}' = k \cdot \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\boldsymbol{\mu}}} \left[\left(\boldsymbol{x} - \frac{1}{k} \cdot \mathbf{1} \right) \cdot \boldsymbol{x}^{\mathrm{T}} \right] \cdot \boldsymbol{\lambda} \pm O(k \cdot \|\boldsymbol{\lambda}\|_1^3), \tag{7.4}$$

as $\|\boldsymbol{\lambda}\|_1 \to 0$. *Notice that* $\mathbb{E}_{\boldsymbol{x}\sim p_{\boldsymbol{\mu}}} \left[\left(\boldsymbol{x} - \frac{1}{k} \cdot \mathbf{1} \right) \cdot \boldsymbol{x}^{\mathrm{T}} \right]$ *is the covariance matrix of* $p_{\boldsymbol{\mu}}$*, and that Eq* (7.4) *becomes a power iteration as* $\|\boldsymbol{\lambda}\|_1 \to 0$.

We present the proof of Theorem 10 in Appendix E.

Next we show that the principal eigenvector of the covariance matrix $\mathbb{E}_{\boldsymbol{x}\sim p_{\boldsymbol{\mu}}} \left[\left(\boldsymbol{x} - \frac{1}{k} \cdot \mathbf{1} \right) \cdot \boldsymbol{x}^{\mathrm{T}} \right]$ is $\boldsymbol{\mu}$.

Lemma 3. *Whenever* $\|\boldsymbol{\mu}_i\|_2 = \|\boldsymbol{\mu}_j\|_2 > 0$ *for all* i, j*, the principal eigenvector of the covariance matrix* $\mathbb{E}_{\boldsymbol{x}\sim p_{\boldsymbol{\mu}}} \left[\left(\boldsymbol{x} - \frac{1}{k} \cdot \mathbf{1} \right) \cdot \boldsymbol{x}^{\mathrm{T}} \right]$ *is* $\boldsymbol{\mu}$*. Moreover, the ratio of the largest to second-largest eigenvalue is at least* $1 + \frac{1}{k}(n-1)\|\boldsymbol{\mu}_1\|_2^2$.

The proof of Lemma 3 can be found in Appendix F.

7.1 Power Pretrained EM

In this section we propose a pretraining step of EM that is based on Theorem 10 and Lemma 3. We prove that this pretraining step brings EM to a region of geometric convergence, avoiding the slow convergence region presented in Section 6. The pretraining steps of our algorithm is a bootstrapping step of EM itself. We call the bootstrapped EM algorithm "Power Pretrained EM".

Power Pretrained EM (L)

1. Choose $\boldsymbol{v}$ uniformly at random from $[0,1]^n$ and set $\tilde{\boldsymbol{\lambda}}^{(0)} = s\boldsymbol{v}$ where $s = \frac{1}{k\|\boldsymbol{v}\|_1}$.
2. Run EM starting from $\tilde{\boldsymbol{\lambda}}^{(0)}$ for L steps to get an estimate $\tilde{\boldsymbol{\lambda}}^{(L)}$.
3. Set $\boldsymbol{\lambda}^{(0)} = \frac{1}{s}\tilde{\boldsymbol{\lambda}}^{(L)}$ and run EM starting from this point until it converges.

Combining Theorem 10 with some robust version of Theorem 8 we can get the following theorem for the convergence of Power Pretrained EM to the correct parameters in the case of many i.i.d. features.

Theorem 11. *Consider a balanced two-class Naive Bayes model with n binary features having the same mean $\mu_i = \mu \in (\delta, 1-\delta)$. Suppose we run the Power Pretrained EM with $L \geq \frac{\log(n/\varepsilon)}{\log(1+(n-1)\delta)}$ for some $\varepsilon > 0$ and initial estimate $\tilde{\boldsymbol{\lambda}}^{(0)}$ chosen uniformly at random from $[-1,1]^n$. Let also $\boldsymbol{\lambda}^{(t)}$ the estimates of $\boldsymbol{\mu}$ while iterating step 3. of Power Pretrained EM. Then, with high probability at least $1/\mathrm{poly}(n)$, it holds that*

$$\left\|\boldsymbol{\lambda}^{(t+1)} - \boldsymbol{\mu}\right\| \leq \kappa \cdot \left\|\boldsymbol{\lambda}^{(t)} - \boldsymbol{\mu}\right\| + \varepsilon$$

with $\kappa = (1-\delta)^{\frac{n-2}{2}}$. This implies that $\left\|\boldsymbol{\lambda}^{(t)} - \boldsymbol{\mu}\right\| - \frac{\varepsilon}{1-\kappa}$ geometrically converges to 0 with rate κ.

The proof of Theorem 11 is deferred to Appendix H.

We now also verify Theorem 11 experimentally by running the Power Pretrained EM starting from exactly the same initial guesses that we used to derive the bad instances of the EM algorithm before. For the comparison to be fair we use the same total number of EM iteration as we used before. That is the number of iterations of both step 2. and step 3. of Power Pretrained EM that we use are equal to the total number of iterations of the original EM. We only need to decide the parameter L of Power Pretrained EM for which we try several values. The results are summarized in Figure 10 in Appendix J

These results are exactly what we would expect from the Theorem 11. The effect of the pretraining step is pretty clear from the figures (a) - (c). In (a), when $L = 10$ the step 2. of Power Pretrained EM hasn't converge yet to the direction of $\boldsymbol{\mu}$ and we observe the same bad behavior as with the original EM algorithm. Although the situation is better for $L = 30$ still the vector $\tilde{\boldsymbol{\lambda}}^{(L)}$ is not yet perfectly aligned with $\boldsymbol{\mu}$ and the progress towards the fixed point vanishes very quickly. Finally for $L = 60$ step 2. of Power Pretrained EM has converge and hence our iteration is equivalent with the single dimensional iteration presented in Section 4.

The same effect takes place in the figures (d)-(f) too although in this case step 2. of Power Pretrained EM seems to have converged even for $L = 30$.

From the simulations we conclude that even in the easier instances of the original EM algorithm the Power Pretrained EM increases the efficiency and hence it is a good policy to do this pretraining step of EM in any case.

To also validate the robustness of the effects presented in Figure 10 we compute the distribution of estimates after some steps of both the original EM and the Power Pretrained EM and we compare them in Figure 11.

7.2 Convergence for Non-Identical Means

In the case of more than two binary features the theoretical analysis we followed so far does not apply exactly the same way. The main difficulty that we face is that now the principal eigenvector of the covariance matrix of the mixture is not $\boldsymbol{\mu}$ and hence the advantage that we get from the pretraining step is not clear and easy to prove. For this reason we experimentally justify the improvement of Power Pretrained EM versus the original EM. As we will see the efficiency of the algorithm increases very much and similar phenomena with the identical mean - non identical guess case happen. This verifies also our intuition that the analysis of the later case gives a lot of information for the behaviour of EM in this more general case.

Since the case with non-identical means is strictly more general and difficult than the case with identical means we don't need to present more experimental evidence that the EM performs poorly in this case. Indeed we can see similar execution with the identical mean case. This is illustrated in the next figure

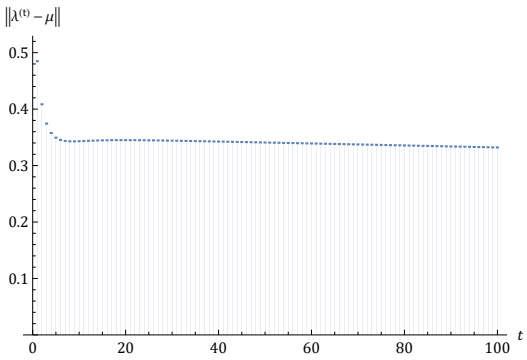

Figure 3: This figure shows the most typical scenario that can be observed when running EM with random initialization for $n = 5$ and $\boldsymbol{\mu} = (0.053, 0.16, 0.09, 0.13, 0.06)$. When we say random initialization we mean again that $\boldsymbol{\lambda}^{(0)}$ is picked uniformly at random from $[0,1]^n$.

We now directly apply the Power Pretrained EM algorithm and the results are shown in the next figure and in more detail in Figure 13 presented in Appendix J, where we used different values of the parameter L the same way as we did in the previous section.

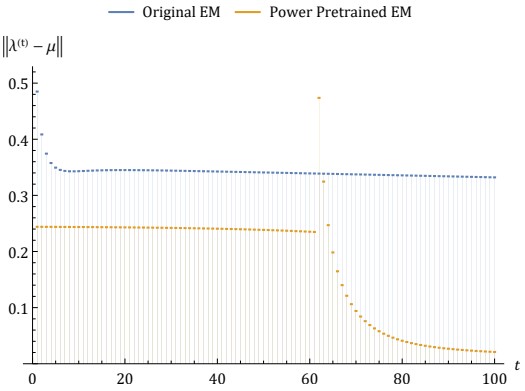

Figure 4: This figure shows the comparison between the execution of the original EM and the Power Prentrained EM for the same initial estimation. The Power Pretrained EM is used with paramter $L = 60$, the number of features is $n = 5$ and the real means of the model are $\boldsymbol{\mu} = (0.053, 0.16, 0.09, 0.13, 0.06)$.

The situation in this case is pretty similar to the corresponding case in the previous section and we can derive the same conclusions as before. The only difference is that although the Power Pretrained EM reduces the error of the algorithm very much it does not succeeds to vanish it completely. The reason is that even when the power method converges as we said the principal eigenvector of the covariance matrix is not parallel to $\boldsymbol{\mu}$. Hence the direction that we ended up with is not perfect as it was in the previous section but has a small error that appears also in our final error. But still the a huge part of the error was reduced and the main difficulty of the instance is captured and that's why we get this great improvement using Power Pretrained EM instead of the original algorithm. Finally, we illustrate the robustness of our results even in this case of non-identical means in Figure 14 of Appendix J.

8 Convergence with Finite Samples

While our theoretical results stated above pertain to the population EM algorithm, we can exploit our analysis to obtain finite sample statements of our results. We show a general statement that allows us to convert result for the population model to a result with finite samples that achieves error rate $O(\frac{n}{\sqrt{N}})$ with N samples.

When we have access to finite number of samples from a balanced Naive Bayes model with two-classes and n binary features having means $\boldsymbol{\mu}$ and $-\boldsymbol{\mu}$, the finite sample EM iteration takes the following form

$$\tilde{\boldsymbol{\lambda}}^{(t+1)} = \bar{M}(\tilde{\boldsymbol{\lambda}}) = \frac{1}{N}\sum_{i=1}^{N} \tanh\left(\tanh^{-1}(\tilde{\boldsymbol{\lambda}}^{(t)}) \cdot \boldsymbol{x}_i\right) \boldsymbol{x}_i. \tag{8.5}$$

For this case we show the following theorem.

Theorem 12. *Consider the finite sample Naive Bayes EM algorithm for a setting with two-classes and n binary features having means $\boldsymbol{\mu}$ and $-\boldsymbol{\mu}$, respectively. Moreover, suppose that there exists a region Λ and constants $\kappa, \varepsilon \geq 0$ such that for all $\boldsymbol{\lambda}^{(0)} \in \Lambda$, it holds that the population iteration of EM given by (2.2) satisfies $\boldsymbol{\lambda}^{(t)} \in \Lambda$ and*

$$\left\|\boldsymbol{\lambda}^{(t+1)} - \boldsymbol{\mu}\right\| \leq \kappa \left\|\boldsymbol{\lambda}^{(t)} - \boldsymbol{\mu}\right\| + \varepsilon.$$

Then, the EM algorithm with N samples, whose iteration is given by (8.5), converges to a point $\boldsymbol{\mu}^*$, such that $\|\boldsymbol{\mu} - \boldsymbol{\mu}^*\| \leq \frac{\varepsilon}{1-\kappa} + \tilde{O}(\frac{n}{\sqrt{N}})$.

The proof of Theorem 12 is given in Appendix I.

References

[BWY14] Sivaraman Balakrishnan, Martin J Wainwright, and Bin Yu. Statistical guarantees for the EM algorithm: From population to sample-based analysis. *arXiv preprint arXiv:1408.2156*, 2014.

[CH08] Stéphane Chrétien and Alfred O Hero. On EM algorithms and their proximal generalizations. *ESAIM: Probability and Statistics*, 12:308–326, 2008.

[DLR77] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[DTZ17] Constantinos Daskalakis, Christos Tzamos, and Manolis Zampetakis. Ten Steps of EM Suffice for Mixtures of Two Gaussians. In *the 30th Annual Conference on Learning Theory (COLT)*, 2017.

[NIP16] NIPS. Workshop on nonconvex optimization for machine learning: Theory and practice. Berlin, Germany, 2016.

[RST+03] Jason D Rennie, Lawrence Shih, Jaime Teevan, David R Karger, et al. Tackling the poor assumptions of naive bayes text classifiers. In *ICML*, volume 3, pages 616–623. Washington DC), 2003.

[RW84] Richard A Redner and Homer F Walker. Mixture densities, maximum likelihood and the EM algorithm. *SIAM review*, 26(2):195–239, 1984.

[Tse04] Paul Tseng. An analysis of the EM algorithm and entropy-like proximal point methods. *Mathematics of Operations Research*, 29(1):27–44, 2004.

[Wu83] CF Jeff Wu. On the convergence properties of the EM algorithm. *The Annals of statistics*, pages 95–103, 1983.

[XHM16] Ji Xu, Daniel Hsu, and Arian Maleki. Global analysis of Expectation Maximization for mixtures of two Gaussians. In *the 30th Annual Conference on Neural Information Processing Systems (NIPS)*, 2016.