# Learning to model sequences generated by switching distributions

**Yoav Freund**
AT&T Bell Labs
600 Mountain Ave.
Murray Hill, NJ, USA

**Dana Ron**
Computer Science Institute.
Hebrew University
Jerusalem, Israel

## Abstract

We study efficient algorithms for solving the following problem, which we call the *switching distributions* learning problem. A sequence $S = \sigma_1\sigma_2\ldots\sigma_n$, over a finite alphabet $\Sigma$ is generated in the following way. The sequence is a concatenation of $K$ *runs*, each of which is a consecutive subsequence. Each run is generated by independent random draws from a distribution $\vec{p}_i$ over $\Sigma$, where $\vec{p}_i$ is an element in a set of distributions $\{\vec{p}_1,\ldots,\vec{p}_N\}$. The learning algorithm is given this sequence and its goal is to find approximations of the distributions $\vec{p}_1,\ldots,\vec{p}_N$, and give an approximate segmentation of the sequence into its constituting runs. We give an efficient algorithm for solving this problem and show conditions under which the algorithm is guaranteed to work with high probability.

## 1 Introduction

Our work is motivated by the Hidden Markov Model (HMM). The HMM is a model for the distribution of sequences over a finite alphabet $\Sigma$. An HMM consists of a finite number of hidden states $i \in [1..N]$, each of which is associated with a distribution $\vec{p}_i$ over the alphabet $\Sigma$. There is a transition probability $q_{i,j}$ associated with each pair of states. The HMM can be seen as a model which generates infinite sequences as follows. At each time step the model generates a single character from $\Sigma$ according to $\vec{p}_i$ where $i$ is its current hidden state, it then makes a transition to a new state $j$ with probability $q_{i,j}$.

HMMs are a popular model in the context of speech analysis. One can view the hidden state as representing the state of the vocal tract of the speaker, which is not directly observable but controls the distribution of the observable sounds. The Baum-Welch algorithm [2] is the predominant algorithm for learning HMMs from examples and produces. In many real-world cases, this algorithm produces accurate hypotheses after a small number of iterations.[1] There is almost no theory for explaining why Baum-Welch performs so well in some cases and badly in others. The theoretical results regarding the problem of learning HMMs of which we are aware are mostly negative, Abe and Warmuth [1] and Gillman and Sipser [5] show that learning HMMs is NP-hard under various conditions.

The model of sequences that we consider is similar to the HMM with the restriction that the transition probabilities assign a very high value to the transition from each hidden state to itself. In other words, the model tends to stay at the same hidden state for long periods of time and switch from state to state only infrequently. Such an assumption can be justified in the context of speech analysis because the time scale in which speech is sampled is usually an order of magnitude smaller than the time scale of changes in the vocal tract.

The assumption of the infrequent transitions lets us alleviate the problem of estimating the transition probabilities and rephrase the learning problem in a slightly different way. In our new formulation the learning problem consists of two interdependent problems: the *modeling* problem, which is to estimate the distributions $\vec{p}_i$, and the *segmentation* problem, which is to partition the sequence into short *runs* that correspond to the different distributions. Given a solution to the segmentation problem the transition probabilities can be easily estimated.

We define the *switching distributions* learning problem as follows. The learning target consists of $N$ distributions $\{\vec{p}_i\}_{i=1}^N$, and a segmentation sequence $\boldsymbol{\eta} = \eta_1,\ldots,\eta_M$ over the integers $1,\ldots,N$. The segmentation sequence is a concatenation of $K$ *runs* each of which is a repetition of a single index. The element $\eta_i$ is the index of the distribution that generates the $i$th element of the sequence. We assume that we are given a sequence $S = \sigma_1\sigma_2\ldots\sigma_M$ over a finite alphabet $\Sigma$. We assume that $N \ll K$, i.e., that the same distribution is used in many different runs. The learner receives a single sequence $S$ of length $M$, that is generated by the target, and its goal is to generate a hypothesis segmentation and a set of hypothesis distributions which are close to those of the target.

This problem is related to the problem of learning switching concepts studied by Blum and Chalasani [3]. However, in their setup the switching entities are concepts, i.e., mappings from some domain to $\{0,1\}$, while in our setup the switching enti-

[1] For an introduction on HMMs and their use in speech analysis and the use of Hidden Markov Model see Rabiner and Juang [7].

ties are distributions over a single space. In this work we give an efficient algorithm for learning switching distributions. We describe several variants of the algorithm, each of which is guaranteed to succeed under slightly different conditions regarding the process which is generating the sequence.

Our algorithm works in the following general way. It starts by finding rough approximations of the distributions $\vec{p}_1, \ldots, \vec{p}_N$. This is done by finding short subsequences of $S$ which, with high probability, are generated mostly by a single distribution. Starting with these approximations of the distributions, the algorithm iterates the following two steps, which are very similar to the "expectation" and "maximization" steps of the Baum-Welch algorithm.

1. Using the approximate distributions, the algorithm finds an approximate segmentation of the sequence.
2. From the approximate segmentation, the algorithm finds new estimates of the distributions.

Our analysis shows that if the errors in the initial estimates of the distributions are sufficiently small then the re-estimation process described above converges rapidly. Specifically, if the errors in the initial estimates are smaller than a constant factor of the distance between the target distributions, then the number of mistakes in the segmentation is, with high probability, smaller than $O(K \log(NM))$ and the error in the re-estimated distributions is $O(\sqrt{(K+D)\log(NM)/M})$. In other words, if the sequence is long enough with respect to the number of runs (and other parameters of the source which we specify later), then even rough initial estimates of the target distributions are sufficient to achieve very accurate estimates within a single iteration of the algorithm.

The paper is organized as follows. In the Section 2 we give the exact statement of the switching distributions problem. In Section 3 we describe the general algorithm that we use to solve the problem. The details of the algorithm depend on the number of distributions $N$, and on the amount of information given to the algorithm concerning the different parameters of the problem. In Section 4 we present our main results. Our strongest result is for $N = 2$ and is given in Section 6. In Section 7 we give a general algorithm for $N > 2$, and in Section 8 we describe how to treat unspecified parameters.

## 2 Description of the Problem

We are interested in the following problem. Let $\Sigma = \{1 \ldots D\}$ be an alphabet of size $D$. Let $\boldsymbol{\eta} = \eta(1) \ldots \eta(M)$, $\eta(i) \in [1..N]$ be the *target segmentation* sequence containing at most $K$ *runs*, where a *run* is a consecutive subsequence $\eta(i) \ldots \eta(i+s)$ consisting of $s+1$ repetitions of a single index in $[1..N]$. Let $\{\vec{p}_j\}_{j=1}^N$ be the set of *target probability vectors* where for each $j$, $1 \leq j \leq N$, $\vec{p}_j$ is a $D$ dimensional probability vector defined over $\Sigma$. We denote by $\vec{p}_j(\sigma)$ the probability assigned to $\sigma$ by $\vec{p}_j$ (i.e., the $\sigma$ coordinate of $\vec{p}_j$).

We assume that a single sequence $S = \sigma_1 \ldots \sigma_M$ of elements from $\Sigma$ is generated, according to the target $(\boldsymbol{\eta}, \{\vec{p}_j\}_{j=1}^N)$, in the following manner. For each $1 \leq i \leq M$, the element $\sigma_i$ is chosen independently at random according to the distribution defined by $\vec{p}_{\eta(i)}$. We are interested in algorithms which, given such a sequence, construct a hypothesis $(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_j\}_{j=1}^N)$, where $\tilde{\boldsymbol{\eta}}$ is a hypothesis segmentation sequence and $\{\tilde{p}_j\}_{j=1}^N$ is a set of distributions. The error of a hypothesis $(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_j\}_{j=1}^N)$ with respect to the target $(\boldsymbol{\eta}, \{\vec{p}_j\}_{j=1}^N)$ is defined as follows

$$err_d(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_j\}_{j=1}^N) \stackrel{\text{def}}{=} \frac{1}{M} \sum_{i=1}^{M} d\left(\vec{p}_{\eta(i)}, \tilde{p}_{\tilde{\eta}(i)}\right) ,$$

where $d(\cdot, \cdot)$ is a measure of divergence between distribution vectors. We give results with respect to three divergence measures: $d_1(\vec{x}, \vec{y}) = \|\vec{x} - \vec{y}\|_1$, $d_2(\vec{x}, \vec{y}) = \|\vec{x} - \vec{y}\|_2$, and $d_3(\vec{x}, \vec{y}) = \|\vec{x} - \vec{y}\|_2^2$. We use $err_1$, $err_2$, $err_3$ to denote the errors of a hypothesis with respect to $d_1, d_2$ and $d_3$ respectively. As for any distribution vectors $\vec{x}, \vec{y}$, $d_1(\vec{x}, \vec{y}) \geq d_2(\vec{x}, \vec{y}) \geq d_3(\vec{x}, \vec{y})$, $err_1$ is the most sensitive measure of the error of the hypothesis and $err_3$ is the least sensitive measure.

Intuitively, a hypothesis with small error is one which defines a sequence of distributions that is very similar to the one which generated the sequence $S$. If the target distributions $\{\vec{p}_j\}_{j=1}^N$ are all far from each other, then the fact that the error of a hypothesis is small implies that to each target distribution there corresponds a hypothesis distribution which is close to it, and that this hypothesis distribution is matched to it on most of the sequence. This means that a hypothesis which has small error solves both the segmentation and the modeling problems described above.

A learning algorithm for the switching distributions problem receives as input a single sequence $S$, together with an accuracy parameter $\epsilon > 0$ and a reliability parameter $\delta > 0$. After time polynomial in $M$, $K$, $N$, $D$, $1/\epsilon$, and $\log(1/\delta)$, the algorithm outputs a hypothesis. We require that there exists a polynomial $q(K, N, D, 1/\epsilon, \log(1/\delta))$, such that if $M \geq q(K, N, D, 1/\epsilon, \log(1/\delta))$, then, with probability at least $1-\delta$ the error of the hypothesis is smaller than $\epsilon$.

## 3 The general algorithm

In this section we describe an efficient algorithm for solving the switching distributions learning problem. Some elements of the algorithm are left unspecified. These elements are implemented differently for the case of two target distributions ($N = 2$) and for the general case of more than two distributions, and are described in detail in the following sections. The reason for the two different implementations is that we were able to derive better results for the case $N = 2$ by using procedures which exploit the fact that the sequence is generated by no more than two distributions.

The algorithm is described in Figure 1. It consists of two parts. In the first part the algorithm finds rough approximations of the target distributions. This is implemented, as will be shown in more detail in the following sections, by locating short subsequences that appear to be generated each by a single distribution. The second part of the algorithm is an iterative part in which the approximate distributions are used to generate an approximate segmentation, and this segmentation is then used to re-estimate the distributions. These two steps are repeated $T$ times, for some $T \geq 1$. On iteration $t$, a *cost vector* $\vec{c}_j^{\,t}$ is associated with each approximate distribution. This cost vector is used to calculate a total cost for any hypothesis segmentation of the sequence as defined in step 2b of the algorithm. The algorithm

**General algorithm for learning switching distributions**

**Input:** A sequence $S = \sigma_1, \sigma_2, \ldots, \sigma_M$ and
$D$ - The size of the alphabet, $N$ - The number of unknown distributions, $K$ - the maximal number of runs, and $\gamma$ - the minimal fraction of the sequence that corresponds to each distribution.

1. **Initialization:** Find initial approximations of the $N$ target distributions: $\tilde{p}_1^0, \ldots, \tilde{p}_N^0$.
2. **Do for** $t = 0, 1, 2, \ldots, T-1$ **:**
   (a) Set the cost vectors $\vec{c}_1^t, \ldots, \vec{c}_N^t$ as functions of the distributions $\tilde{p}_1^t, \ldots, \tilde{p}_N^t$.
   (b) Find the segmentation $\tilde{\boldsymbol{\eta}}^t = \tilde{\eta}^t(1), \ldots, \tilde{\eta}^t(M)$ which minimizes the total cost:
   $$C\left(\{\vec{c}_j^t\}_{j=1}^N, \tilde{\boldsymbol{\eta}}^t, S\right) = \sum_{i=1}^{M} \vec{c}^t_{\tilde{\eta}^t(i)}(\sigma_i)$$
   (c) Calculate the new estimates of the distributions. For $j \in [1, \ldots, N]$ set $\tilde{p}_j^{t+1}$ to the empirical distribution of the elements of $S$ for which $\tilde{\boldsymbol{\eta}}^t(i) = j$.
3. **Output** the segmentation $\tilde{\boldsymbol{\eta}}^{T-1}$, and the distributions $\tilde{p}_1^T, \ldots, \tilde{p}_N^T$.

Figure 1: The general learning algorithm.

selects the segmentation $\tilde{\boldsymbol{\eta}}^t$ that achieves the minimal cost. Using this segmentation the algorithm generates new estimates of the distributions and the process repeats. Finally, after T such iterations, the hypothesis $(\tilde{\boldsymbol{\eta}}^{T-1}, \{\tilde{p}_j^T\}_{j=1}^N)$ is output. We were not able to demonstrate that several iterations achieve substantially better performance than a single one, and so our analysis concentrates on the case $T = 1$. However, in practice it seems likely that additional iterations would improve the accuracy of the hypothesis. We return to this point at the end of Section 6.

Finding the segmentation with at most $K$ runs that minimizes the total cost for a given set of cost vectors can be performed in time $O(\log(K)M^3)$ using a dynamic programming technique, which is essentially the same as the well known Viterbi algorithm [8].

The cost vectors are chosen so that with high probability the segmentation with the lowest total cost does not differ significantly from the target segmentation. More specifically, our goal is to select cost vectors that satisfy the following two properties: (1) The *expected cost* of the target segmentation is smaller than the expected cost of any other segmentation (with at most $K$ runs). (2) With high probability, the segmentation that minimizes the cost on a sample sequence $S$, has a small number of segmentation errors. Once the segmentation with the lowest total cost is found, the $N$ probability distributions are re-estimated, and the process is repeated.

The key property of this iterative process, as we shall show in the following sections, is that if the error of the initial estimates of the distributions is smaller than some threshold, then the iterative process increases the accuracy of the models very rapidly. We shall show that this threshold need not depend on the approximation parameter $\epsilon$, but rather is of the order of the smallest distance between any pair of target distributions.

## 4 Summary of Results

Before we present our results, we add the following notation.

For $j \in [1, \ldots, N]$, we use $n_j$ to denote the number of elements in $S$ corresponding to the distribution $\vec{p}_j$, and define $\gamma \stackrel{\text{def}}{=} \min_j(n_j)/M$. In other words, $\gamma$ is the minimal fraction of elements in $S$ corresponding to any single distributions. We use $L$ to denote the length of the shortest run in $S$.

We summarize our results in four theorems which correspond to different variants of the algorithm described in Section 3. In the first variant we assume that there are only two target distributions, and that the algorithm receives as input $K$, and[2] $\gamma$. The error of the algorithm's hypothesis is measured with respect to $d_1$, which is the most sensitive distance measure we use.

**Theorem 1 (Main Theorem for the case of two distributions)** *There exists a switching distributions learning algorithm such that for any target $(\boldsymbol{\eta}, \{\vec{p}_1, \vec{p}_2\})$, defining sequences whose length satisfies*

$$\frac{M}{\log M} = \Omega\left(\max\left(\frac{K \cdot (D + \log\frac{1}{\delta})}{\max(\Delta_1^2, \epsilon^2) \cdot \gamma}, \frac{K + D + \log\frac{1}{\delta}}{\epsilon^2 \cdot \gamma}\right)\right)$$

*then the algorithm, when receiving a sequence $S$, generated according to the target, together with $K$ and $\gamma$, outputs a hypothesis $(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_1, \tilde{p}_2\})$ such that with probability at least $1-\delta$, $err_1(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_1, \tilde{p}_2\}) \leq \epsilon$.*

In the second variant, we remove the assumption that $N$ is at most two, but we assume that the algorithm is given $K$ and $L$. We would like to note that we do not assume the algorithm knows $N$. In this case we require that the target distributions are well separated. We measure the separation between the target distributions by $\Delta_2 \stackrel{\text{def}}{=} \min_{j \neq j'} ||\vec{p}_{j'} - \vec{p}_j||_2$. Our requirements on $S$ are that $\gamma$ and $\Delta_2$ are polynomially related to $\epsilon$, and that $L$ grows logarithmically with $M$. The error of the algorithm's hypothesis is measured with respect to $d_2$.

[2] It actually suffices that the algorithm receive only an upper bound on $K$ and a lower bound on $\gamma$. In such a case in the theorem below, $K$ and $\gamma$ are simply exchanged by these bounds. A similar statement holds in the case of Theorem 2 where we may assume that the algorithm receives only a lower bound on $L$ (as well as an upper bound on $K$).

**Theorem 2 (Main Theorem for general $N$)** *There exists a switching distributions learning algorithm such that for any target $(\boldsymbol{\eta}, \{\vec{p}_j\}_{j=1}^N)$ defining sequences whose length satisfies*

$$\frac{M}{\log M} = \Omega\left(\frac{K \log N + D + \log \frac{1}{\delta}}{\gamma \min(\epsilon^2, \Delta_2^3 \epsilon)}\right) ,$$

*and are a concatenation of runs of length at least*

$$\frac{L}{\log L} = \Omega\left(\frac{\log M + D + \log \frac{1}{\delta}}{\Delta_2^2}\right) ,$$

*then the algorithm, when receiving a sequence $S$, generated according to the target, together with $K$ and $L$, outputs a hypothesis $(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_j\}_{j=1}^N)$ such that with probability at least $1 - \delta$, $err_2(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_j\}_{j=1}^N) \leq \epsilon$.*

The third variant of our algorithm needs no input other than $S$ and makes only the assumption that $N \leq 2$. The error of the algorithm's hypothesis is measured with respect to $d_3$.

**Theorem 3 (Main Theorem for two distributions and unspecified parameters)** *There exists a switching distributions learning algorithm such that for any target $(\boldsymbol{\eta}, \{\vec{p}_1, \vec{p}_2\})$, defining sequences whose length satisfies*

$$\frac{M}{\log M} = \Omega\left(\max\left(\frac{K \cdot (D + \log \frac{1}{\delta})}{\max(\Delta_1^2, \epsilon^2) \cdot \gamma}, \frac{K + D + \log \frac{1}{\delta}}{\epsilon^2 \cdot \gamma}\right)\right)$$

*then the algorithm, when receiving a sequence $S$ generated according to the target, outputs a hypothesis $(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_1, \tilde{p}_2\})$ such that with probability at least $1 - \delta$, $err_3(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_1, \tilde{p}_2\}) \leq \epsilon$.*

In our fourth variant we do not assume that the algorithm is given any of the parameters of the problem. However we still require the existence of a lower bound on $L$ which grows logarithmically with $M$, and that $\gamma$ and $\Delta_2$ are polynomially related to $\epsilon$. The error of the algorithm's hypothesis is measured with respect to $d_3$.

**Theorem 4 (Main Theorem for general $N$ and unspecified parameters)** *There exists a switching distributions learning algorithm, such that for any target $(\boldsymbol{\eta}, \{\vec{p}_j\}_{j=1}^N)$ defining sequences whose length satisfies*

$$\frac{M}{\log M} = \Omega\left(\frac{K \log N + D + \log \frac{1}{\delta}}{\gamma \min(\epsilon^2, \Delta_2^3 \epsilon)}\right) ,$$

*and are a concatenation of runs of length at least*

$$\frac{L}{\log L} = \Omega\left(\frac{\log M + D + \log \frac{1}{\delta}}{\Delta_2^2}\right) ,$$

*then the algorithm, when receiving a sequence $S$, generated according to the target, outputs a hypothesis $(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_j\}_{j=1}^N)$ such that with probability at least $1 - \delta$, $err_3(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_j\}_{j=1}^N) \leq \epsilon$.*

The theorems presented above result from an analysis of a *single* iteration of step 2 of the learning algorithm. It is natural to ask whether by increasing the number of iterations, we could significantly weaken the requirements on $M$. Our analysis does not support such a claim, and we shall later discuss this question briefly.

## 5 Useful Inequalities

In the proofs of our theorems and lemmas we apply several well known inequalities that are given here as lemmas. The first is a Chernoff/Hoeffding type bound, derived by Littlestone [6], and the second is due to Sanov ([4],page 292).

**Lemma 5** *For $m > 0$, let $X_1, X_2, \ldots X_m$ be $m$ independent random variables where $a_i \leq X_i \leq b_i$. Let $p = \sum_i E[X_i]/m$. Then, for $w > 0$,*

$$Pr\left[\sum_{i=1}^{m} X_i \geq pm + w\right] \leq exp\left(-\frac{2w^2}{\sum_{i=1}^{m}(b_i - a_i)^2}\right) .$$

**Lemma 6 (Sanov's Inequality)** *For an alphabet $\Sigma$ of size $D$, let $\vec{p}$ be a $D$ dimensional probability vector defined over $\Sigma$. Let $T$ be a random sample of size $m$ generated according to $\vec{p}$, and let $\vec{X}_T$, the* type *of $T$, be a $D$ dimensional probability vector defined as follows: the $d$'th coordinate of $\vec{X}_T$ is the relative frequency of the symbol $d$ in $T$. Then, for any $\alpha > 0$,*

$$Pr\left[D_{KL}[\vec{X}_T || \vec{p}] \geq \alpha\right] \leq (m+1)^D 2^{-\alpha m} ,$$

*where $D_{KL}[\vec{X}_T || \vec{p}]$ is the* **Kullback Leibler (KL) divergence** *between the distributions and is defined as follows:*

$$D_{KL}[\vec{X}_T || \vec{p}] \stackrel{\text{def}}{=} \sum_{\sigma \in \Sigma} \vec{X}_T(\sigma) \log \frac{\vec{X}_T(\sigma)}{\vec{p}(\sigma)} .$$

One more useful inequality is the following:

**Lemma 7** *Let $\vec{p}_1$ and $\vec{p}_2$ be two probability vectors, then:*

$$D_{KL}[\vec{p}_1 || \vec{p}_2] \geq \frac{1}{2 \ln 2} ||\vec{p}_1 - \vec{p}_2||_1^2 \geq \frac{1}{2 \ln 2} ||\vec{p}_1 - \vec{p}_2||_2^2$$

## 6 The Case of Two Distributions

In this section we consider the case in which there are two target probability distributions $\vec{p}_1$ and $\vec{p}_2$ over an alphabet $\Sigma$ of size $D$. The $L_1$ distance between the two vectors, $||\vec{p}_1 - \vec{p}_2||_1$, plays an important role in our analysis, and is denoted by $\Delta_1$. We assume that the algorithm knows $K$, the number of switches in the sequence, and $\gamma$, the minimum between the fraction of elements in the target sequence, $\boldsymbol{\eta}$, generated by $\vec{p}_1$, and the fraction generated by $\vec{p}_2$. As noted in Section 4, it suffices that the algorithm have only an upper bound on $K$ and a lower bound on $\gamma$. In Section 8 we give bounds on the additional error incurred when we remove this assumption

In Figure 2 we describe how we get initial approximations $\tilde{p}_1^0$ and $\tilde{p}_2^0$ of $\vec{p}_1$ and $\vec{p}_2$ respectively, and we define the pair of cost vectors $\vec{c}_1^{\,t}$ and $\vec{c}_2^{\,t}$, given approximations $\tilde{p}_1^t$ and $\tilde{p}_2^t$ of $\vec{p}_1$ and $\vec{p}_2$.

The initial approximation procedure is based on the following two facts. The first is that by definition of $K$ and $\gamma$, both for $\vec{p}_1$ and for $\vec{p}_2$ there exists a subsequence of $S$ of length $\gamma M/K$ that was generated *solely* according to that probability distribution. The second fact is that, in expectation, the distance between pairs of empirical distributions defined based on pairs

**Initial estimates for two distributions.**

1. Set $\ell = \gamma M/K$.
   (If $L$, the minimum length of any run in $S$, is known, set $\ell = \max(\gamma M/K, L)$).
2. For each $i \in [1, \ldots, M - \ell + 1]$ and each $\sigma \in [1, \ldots, D]$, let $f_i(\sigma)$ be the fraction of the elements in $\sigma_i \sigma_{i+1} \ldots \sigma_{i+\ell-1}$ which are equal to $\sigma$. Let $\vec{f}_i$ denote the vector $\langle f_i(1), f_i(2), \ldots, f_i(D) \rangle$.
3. Find the pair of indices $1 \leq i_1 < i_2 \leq M - \ell + 1$, for which $\|\vec{f}_{i_1} - \vec{f}_{i_2}\|_1$ is maximized.
4. Set $\tilde{p}_1^0 = f_{i_1}$ and $\tilde{p}_2^0 = f_{i_2}$.

**Choice of cost vectors for two distributions**

1. Given estimates $\tilde{p}_1^t$ and $\tilde{p}_2^t$, let $\Sigma_1 \stackrel{\text{def}}{=} \{\sigma \in \Sigma \ : \ \tilde{p}_1^t(\sigma) > \tilde{p}_2^t(\sigma)\}$, and for $j \in \{1, 2\}$, let $\tilde{p}_j^t(\Sigma_1) \stackrel{\text{def}}{=} \sum_{\sigma \in \Sigma_1} \tilde{p}_j^t(\sigma)$.
2. Let $\vec{d}^t$ be defined as follows:
$$\forall \sigma \in \Sigma_1, \ \vec{d}^t(\sigma) = \frac{\tilde{p}_1^t(\Sigma_1) + \tilde{p}_2^t(\Sigma_1)}{2} - 1 \ , \quad \forall \sigma \in \Sigma - \Sigma_1, \ \vec{d}^t(\sigma) = \frac{\tilde{p}_1^t(\Sigma_1) + \tilde{p}_2^t(\Sigma_1)}{2} \ .$$
3. Set $\vec{c}_1^t = \vec{d}^t, \ \ \vec{c}_2^t = \vec{0}$ .

Figure 2: The initialization procedure and the choice of the cost vectors for the case of two distributions.

of subsequences is maximized when one of the subsequences was generated according to $\vec{p}_1$ and the second according to $\vec{p}_2$. Using these two facts we show that the pair of distributions chosen are good initial approximations of $\vec{p}_1$ and $\vec{p}_2$.

Our main result for the case of two distributions is stated in Theorem 1 whose proof is divided into three lemmas. We use the following notation. Similar to the definition of $n_j$, which is used with respect to the target segmentation, we use $n_j(\boldsymbol{\eta}')$ to denote the number of sequence locations for which $\eta'(i) = j$, and $n_{j,j'}(\boldsymbol{\eta}')$ to denote the number of sequence locations for which $\eta(i) = j$ and $\eta'(i) = j'$.

Observe that the definition of the distribution of sample sequences is invariant under renaming of the target distribution. It is thus clear that the hypothesis generated by any learning algorithm can be close to the target only up to an arbitrary permutation of the distributions. This issue is side-stepped by our definition of the error of a hypothesis but the lemmas that constitute the proof of Theorem 1 refer to the one-to-one mapping which defines this permutation. However, as the proofs are identical for any permutation, we shall refer to this mapping only in the statements of the lemmas and otherwise consider only the case in which the names given to the distributions by the target and by the hypothesis are identical.

First, we show that if $M$, the length of the sequence, is large enough, then the initial estimates $\tilde{p}_1^0$ and $\tilde{p}_2^0$ of $\vec{p}_1$ and $\vec{p}_2$ are guaranteed to have small error.

**Lemma 8 (Initialization error for two distributions)** *If for some $\rho$, $0 < \rho \leq \Delta_1/2$, the length $M$ of the sequence $S$ satisfies*
$$\frac{M}{\ln M} \geq \frac{2 \cdot K(D + \ln \frac{1}{\delta})}{\rho^2 \cdot \gamma} \ ,$$
*then with probability at least $1 - \delta$, there exists a one-to-one mapping $\phi : \{1, 2\} \rightarrow \{1, 2\}$ such that for $j \in \{1, 2\}$, $\|\tilde{p}^0_{\phi(j)} - \vec{p}_j\|_1 \leq 3\rho$ .*

**Proof:** For a given index $i \in [1, \ldots, M - \ell + 1]$, let $S_{i,i+\ell-1} \stackrel{\text{def}}{=} \sigma_i \ldots \sigma_{i+\ell-1}$, and let $\alpha_i$ be such that the fraction of symbols in $S_{i,i+l-1}$ that were generated by $\vec{p}_1$ and $\vec{p}_2$ are $(1 - \alpha_i)$ and $\alpha_i$ respectively. We say that a vector $\vec{f}_i$ is *pure* if either $\alpha_i = 0$ or $\alpha_i = 1$. For every pair of indices $1 \leq i_1 < i_2 \leq M - \ell + 1$, let $e_{i_1 i_2} = \|\vec{f}_{i_1} - \vec{f}_{i_2}\|_1$. It is not hard to show that the expected value of $e_{i_1 i_2}$ achieves the maximal value of $\Delta_1$ for a pair of pure vectors $\vec{f}_{i_1}$ and $\vec{f}_{i_2}$ satisfying $\alpha_{i_1} = 0$ (1) and $\alpha_{i_2} = 1$ (0).

We shall show that for some $\rho < \Delta_1/2$, (which we set subsequently), and for every $i$,
$$\|\vec{f}_i - ((1 - \alpha_i)\vec{p}_1 + \alpha_i \vec{p}_2)\|_1 \leq \rho \ . \tag{1}$$

Let $\vec{f}_{i_a}$ and $\vec{f}_{i_b}$ be the pair of vectors for which $e_{i_a i_b}$ is maximized. Without loss of generality we assume that $\alpha_{i_a} \leq (1 - \alpha_{i_b})$. Based on our choice of $\ell$, there must exist a pair of pure vectors $\vec{f}_{i_1}$ and $\vec{f}_{i_2}$ having $\alpha_{i_1} = 0$, and $\alpha_{i_2} = 1$. Since for this pure pair $e_{i_1 i_2} \geq \Delta_1 - 2\rho$, for the maximizing pair, $\vec{f}_{i_a}$ and $\vec{f}_{i_a}$, $e_{i_a i_b} \geq \Delta_1 - 2\rho$ as well. It follows that $\|\vec{f}_{i_a} - \vec{p}_1\|_1 \leq 3\rho$ and $\|\vec{f}_{i_b} - \vec{p}_2\|_1 \leq 3\rho$.

It remains to show that the selection of $\rho$ assumed in Equation 1 exists. Applying Lemma 6 and using the bound on the KL divergence given in Lemma 7, we have that for every $1 \leq i \leq M - \ell + 1$,
$$Pr_S\left[\left\|\vec{f}_i - ((1 - \alpha_i)\vec{p}_1 + \alpha_i \vec{p}_2)\right\|_1 > \rho\right]$$
$$< \ \ (\ell + 1)^D \exp\left(-\frac{1}{2}\rho^2 \ell\right) \ . \tag{2}$$

There are at most $M$ such vectors, and hence, by setting $M$ as in the statement of the lemma, we find that, with probability at least $1 - \delta$, $\|\vec{f}_i - ((1 - \alpha_i)\vec{p}_1 + \alpha_i \vec{p}_2)\|_1 \leq \rho$, for every $i$, as required. ■

Secondly, we show that if the errors of the estimates $\tilde{p}_1^t$ and $\tilde{p}_2^t$ are not too large then, with high probability, only a small number of mistakes exist in the segmentation with the lowest cost, defined using the corresponding cost vectors.

**Lemma 9 (Segmentation error for two distributions)** *Let*

$$\rho = \min_{\psi} \max_{j \in \{1,2\}} \|\tilde{p}^t_{\psi(j)} - \vec{p}_j\|_1 \ ,$$

*where $\psi$ ranges over the (two) one-to-one mappings from $\{1,2\}$ to $\{1,2\}$. Let $\phi$ be a mapping that achieves the minimum. If $\rho < \frac{1}{6}\Delta_1$, then with probability at least $1-\delta$*

$$n_{1,\phi(2)}(\tilde{\boldsymbol{\eta}}^t) + n_{2,\phi(1)}(\tilde{\boldsymbol{\eta}}^t) < \frac{32(\ln(1/\delta) + K\ln(2M))}{(\Delta_1 - 6\rho)^2} \ ,$$

*where $\tilde{\boldsymbol{\eta}}^t$ is the $t$'th hypothesis segmentation.*

**Proof:** Assume, without loss of generality, that $\phi$ is the identity mapping. By our definition of the cost vectors $\vec{c}^t_1$ and $\vec{c}^t_2$, we have that for any given segmentation $\boldsymbol{\eta}'$,

$$E_S\left[C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}, S\right) - C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}', S\right)\right]$$

$$= n_{1,2}(\boldsymbol{\eta}')\vec{p}_1 \cdot \vec{d}^t - n_{2,1}(\boldsymbol{\eta}')\vec{p}_2 \cdot \vec{d}^t \ , \quad (3)$$

where $\vec{d}^t = \vec{c}^t_1 - \vec{c}^t_2$ is as defined in Figure 2. We first verify that $\vec{p}_1 \cdot \vec{d}^t < 0$ and $\vec{p}_2 \cdot \vec{d}^t > 0$, and hence for every segmentation $\boldsymbol{\eta}' \neq \boldsymbol{\eta}$, $E_S\left[C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}, S\right)\right] < E_S\left[C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}', S\right)\right]$. For $j \in \{1,2\}$, let $\vec{e}^t_j = \tilde{p}^t_j - \vec{p}_j$, thus $\|\vec{e}^t_j\|_1 \le \rho \ (< \frac{1}{6}\|\vec{p}_1 - \vec{p}_2\|_1)$, and we get

$$
\begin{aligned}
\vec{p}_1 \cdot \vec{d}^t &= \tilde{p}^t_1 \cdot \vec{d}^t - \vec{e}^t_1 \cdot \vec{d}^t & (4)\\
&= \sum_{\sigma \in \Sigma_1} \tilde{p}^t_1(\sigma)\left(\frac{\tilde{p}^t_1(\Sigma_1) + \tilde{p}^t_2(\Sigma_1)}{2} - 1\right) & (5)\\
&+ \sum_{\sigma \in \Sigma - \Sigma_1} \tilde{p}^t_1(\sigma)\frac{\tilde{p}^t_1(\Sigma_1) + \tilde{p}^t_2(\Sigma_1)}{2} - \vec{e}^t_1 \cdot \vec{d}^t & \\
&= \tilde{p}^t_1(\Sigma_1)\left(\frac{\tilde{p}^t_1(\Sigma_1) + \tilde{p}^t_2(\Sigma_1)}{2} - 1\right) & (6)\\
&+ \left(1 - \tilde{p}^t_1(\Sigma_1)\right)\frac{\tilde{p}^t_1(\Sigma_1) + \tilde{p}^t_2(\Sigma_1)}{2} - \vec{e}^t_1 \cdot \vec{d}^t & \\
&= \frac{\tilde{p}^t_2(\Sigma_1) - \tilde{p}^t_1(\Sigma_1)}{2} - \vec{e}^t_1 \cdot \vec{d}^t & (7)\\
&\le -\frac{1}{4}\|\tilde{p}^t_1 - \tilde{p}^t_2\|_1 + \rho & (8)\\
&\le -\frac{1}{4}\Delta_1 + \frac{3}{2}\rho \ < \ 0 \ . & (9)
\end{aligned}
$$

Similarly

$$\vec{p}_2 \cdot \vec{d}^t \ge \frac{1}{4}\Delta_1 - \frac{3}{2}\rho > 0 \ . \quad (10)$$

It remains to bound the probability that $C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}', S\right) < C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}, S\right)$ for a segmentation $\boldsymbol{\eta}'$ such that $n_{1,2}(\boldsymbol{\eta}') + n_{2,1}(\boldsymbol{\eta}') \ge \frac{8(\ln(1/\delta) + K\ln(2M))}{(\Delta_1 - 6\rho)^2}$. The difference in the total cost, $C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}', S\right) - C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}, S\right)$, is a sum of the contributions of the elements of $S$ for which $\eta(i) \neq \tilde{\eta}(i)$. These contributions are independent, and the difference between the two values that are possible for each element is exactly 1. Thus, applying Lemma 5, the probability that the total cost of $\tilde{\boldsymbol{\eta}}$ is smaller than that of $\boldsymbol{\eta}$ is upper bounded by

$$
\begin{aligned}
&Pr_S\left[C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}', S\right) < C\left(\{\vec{c}^t_1, \vec{c}^t_2\}, \boldsymbol{\eta}, S\right)\right] \\
&\le exp\left(-\frac{\left(n_{1,2}\vec{p}_1 \cdot \vec{d}^t - n_{2,1}\vec{p}_2 \cdot \vec{d}^t\right)^2}{2(n_{1,2} + n_{2,1})}\right) \quad (11)\\
&= exp\left(-\frac{n_{1,2} + n_{2,1}}{2}\, D^2\right) \ , \quad (12)
\end{aligned}
$$

where $D$ is a weighted average of $\vec{p}_1 \cdot \vec{d}^t$ and $-\vec{p}_2 \cdot \vec{d}^t$:

$$D = \left(\frac{n_{1,2}}{n_{1,2} + n_{2,1}}\vec{p}_1 \cdot \vec{d}^t - \frac{n_{2,1}}{n_{1,2} + n_{2,1}}\vec{p}_2 \cdot \vec{d}^t\right) \ . \quad (13)$$

From our assumption on the size of $n_{1,2} + n_{2,1}$, and by substituting our bounds on $\vec{p}_1\vec{d}^t$ and $\vec{p}_2\vec{d}^t$, the probability above is bounded by $\delta/((2M)^K)$. Since the total number of possible segmentation containing at most $K$ runs, is at most $M^K \cdot 2^K$, we get the statement of the lemma. ■

In the third lemma we show that if the segmentation $\tilde{\boldsymbol{\eta}}^t$ has a small number of errors, then good new estimates of $\vec{p}_1$ and $\vec{p}_2$ can be computed using $\tilde{\boldsymbol{\eta}}^t$.

**Lemma 10 (Reevaluation error for two distributions)** *Suppose $\phi : [1,2] \to [1,2]$ is a one to one mapping such that*

$$\beta = \max(n_{1,\phi(2)}(\tilde{\boldsymbol{\eta}}^t), n_{2,\phi(1)}(\tilde{\boldsymbol{\eta}}^t))/M \ ,$$

*and $\beta < \gamma$. Then for $j \in \{1,2\}$*

$$\|\tilde{p}^{t+1}_{\phi(j)} - \vec{p}_j\|_1 \le \frac{\beta}{\gamma - \beta}\Delta_1 + var \ ,$$

*where*

$$var = \sqrt{2\frac{K\ln(2M) + D\ln(M+1) + \ln(1/\delta)}{(\gamma - \beta)M}} \ .$$

**Proof:** Assume, without loss of generality, that $\phi$ is the identity mapping. For a given segmentation $\boldsymbol{\eta}'$, for $j \in \{1,2\}$, let $\tilde{p}^{\boldsymbol{\eta}'}_j$ be the empirical distribution of the elements $\sigma_i$ in $S$ for which $\boldsymbol{\eta}'(i) = j$. We define the deviation, $e_j(\boldsymbol{\eta}')$, of $\tilde{p}^{\boldsymbol{\eta}'}_j$ from its mean as follows:

$$e_j(\boldsymbol{\eta}') \stackrel{\text{def}}{=} \left\| \tilde{p}^{\boldsymbol{\eta}'}_j - \left(\frac{n_{1,j}(\boldsymbol{\eta}')\vec{p}_1 + n_{2,j}(\boldsymbol{\eta}')\vec{p}_2}{n_{1,j}(\boldsymbol{\eta}') + n_{2,j}(\boldsymbol{\eta}')}\right)\right\|_1 \ . \quad (14)$$

We show that there exists $\rho < 1$ (which is set subsequently), such that for every $\boldsymbol{\eta}'$, having $\max(n_{1,2}(\boldsymbol{\eta}'), n_{2,1}(\boldsymbol{\eta}'))/M \le \beta$, $e_1(\boldsymbol{\eta}'), e_2(\boldsymbol{\eta}') \le \rho$. Thus, in particular, $e_1(\tilde{\boldsymbol{\eta}}^t), e_2(\tilde{\boldsymbol{\eta}}^t) \le \rho$ and

$$
\begin{aligned}
\|\tilde{p}^t_1 - \vec{p}_1\|_1 &\le \left\|\frac{n_{1,1}(\tilde{\boldsymbol{\eta}}^t)\vec{p}_1 + n_{2,1}(\tilde{\boldsymbol{\eta}}^t)\vec{p}_2}{n_{2,1}(\tilde{\boldsymbol{\eta}}^t) + n_{1,1}(\tilde{\boldsymbol{\eta}}^t)} - \vec{p}_1 -\right\|_1 + \rho \\
&= \frac{n_{2,1}(\tilde{\boldsymbol{\eta}}^t)}{n_{1,1}(\tilde{\boldsymbol{\eta}}^t) + n_{2,1}(\tilde{\boldsymbol{\eta}}^t)}\Delta_1 + \rho & (15)\\
&\le \frac{n_{2,1}(\tilde{\boldsymbol{\eta}}^t)}{n_1 - n_{1,2}(\tilde{\boldsymbol{\eta}}^t)}\Delta_1 + \rho & (16)\\
&\le \frac{\beta}{\gamma - \beta}\Delta_1 + \rho \ , & (17)
\end{aligned}
$$

where the last inequality follows from our assumptions on $n_{2,1}(\tilde{\boldsymbol{\eta}}^t)$ and $n_1$. The same bound on $\|\tilde{p}^t_2 - \vec{p}_2\|_1$ is obtained analogously.

It remains to bound $\rho$. Applying Lemma 6 and using the bound on the KL divergence given in Lemma 7, we have that with probability at least $1-\delta$, for every segmentation $\boldsymbol{\eta}'$ having $\max(n_{1,2}(\boldsymbol{\eta}'), n_{2,1}(\boldsymbol{\eta}'))/M \leq \beta$, and for $j \in \{1,2\}$,

$$e_j(\boldsymbol{\eta}') \leq \sqrt{2\frac{K\ln(2M) + D\ln(M+1) + \ln(1/\delta)}{(\gamma-\beta)M}} \ . \quad (18)$$

■

**Proof Sketch of Theorem 1:** In order to show that

$$err_1(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_1, \tilde{p}_2\}) \leq \epsilon$$

we separate our argument into three parts, according to the values of $\gamma$ and $\Delta_1$.

First, we consider the case in which both $\gamma$ and $\Delta_1$ are greater than $\epsilon/8$. In this case we show that the segmentation error is $O(\epsilon/\Delta_1)$ and that the estimation error is $O(\epsilon)$. It is not hard to verify that for the right choice of constants the total error is then bounded by $\epsilon$. The condition on the length of the sequence, $M$, together with Lemma 8 guarantee that, with high probability, the initialization procedure generates estimates, $\tilde{p}_j^0$, whose error is smaller than $\Delta_1/12$. Using this in Lemma 9 we get that the segmentation error, $\tilde{\boldsymbol{\eta}}^0$, is $O((\epsilon^2\gamma)/\Delta_1^2)$. From our assumption on $\Delta_1$ we have that this expression is upper bounded by $O(\epsilon/\Delta_1)$ as desired. Using our assumption on $\gamma$ and the bound in Lemma 10, we get that the error of the re-estimated probabilities, $\tilde{p}_j^1$, is $O(\epsilon)$ as required.

If $\gamma \leq \epsilon/8$ then it is not hard to verify that under our assumption on the size of $M$, with probability at least $1-\delta$ the hypothesis which constitutes of a single run, together with the corresponding probability distribution (defined based on the complete sequence), has error bounded by $\epsilon$. As $\gamma$ is part of the input to the algorithm, the algorithm checks for this condition and output the single-run hypothesis.

If $\Delta_1 \leq \epsilon/8$ then the single-run hypothesis has small error as well. However, as $\Delta_1$ is not part of the input, the algorithm cannot check for this condition. Nonetheless, it is easy to check that in this situation, *any* hypothesis segmentation $\boldsymbol{\eta}'$, for which $\min(n_1(\boldsymbol{\eta}'), n_2(\boldsymbol{\eta}')) \geq \gamma M/2$ (together with the corresponding estimated probability distributions), has error at most $\epsilon$. Thus, as a last step of the algorithm we check if $\min(n_1(\tilde{\boldsymbol{\eta}}^0), n_2(\tilde{\boldsymbol{\eta}}^0)) \geq \gamma M/2$. If this condition holds then we simply output $\tilde{\boldsymbol{\eta}}^0$. Otherwise, we output the single-run hypothesis. It follows from the first part of this proof that if $\Delta_1 \geq \epsilon/8$ then the condition holds with high probability. ■

Theorem 1 summarizes the convergence properties of the algorithm when step 3 of the algorithm is executed once. It is interesting to consider the convergence properties that are implied by Lemmas 9 and 10 in a little more detail.

According to Lemma 9, if the errors of the estimates of $\vec{p}_1$ and $\vec{p}_2$ are smaller than a constant fraction of $\Delta_1 = ||\vec{p}_1 - \vec{p}_2||_1$, then the number of segmentation errors can be decreased to an arbitrarily small fraction of $M$ by increasing $M$, this, in turn, decreases the error of the new estimates of the distributions that result from the segmentation. Intuitively, this means that there is a "basin of attraction" of the estimates of the distributions, whose "size" is proportional to the distance $\Delta_1$. If the algorithm starts with an estimate of the distribution that is within this basin of attraction then the estimate it gives in the following iteration is very accurate. On the other hand, our analysis predicts that iterating the algorithm more than once will not significantly improve the segmentation error. This is because even if the estimates of the distributions are perfect, there will be segmentation errors as a result of the randomness of the sequence $S$.

Next we consider the error in the estimates of the distributions. From our analysis we see that, given our lower bound on the length of the sequence, the estimation error after the initialization step is[3] $O(\epsilon\sqrt{KD/(K+D)})$ and the error after a single re-estimation step is $O(\epsilon)$.

## 7 A General Number of Distributions

In this section we consider the case in which there are $N > 2$ target probability distributions $\vec{p}_1, \ldots, \vec{p}_N$. In this case the problem of finding good cost vectors to be associated with the different distributions is more complicated. We have to choose $N$ cost vectors, one per distribution, but we also have to satisfy $\binom{N}{2}$ sets of requirements, because each *pair* of distributions has to be distinguished well by the corresponding pair of cost vectors. The choice of the cost vectors that we have found is described in Figure 3. This choice is a generalization of the squared loss in the binary case ($D = 2$), and allows us to bound the error of the algorithm according to $err_2$ (which is weaker, i.e., less sensitive, than $err_1$).

The initialization procedure that is used in the two-distribution case cannot be applied to the general case. The initialization procedure that we suggest requires that the segmentation sequence $\boldsymbol{\eta}$ is such that *all* of the runs in $\boldsymbol{\eta}$ are longer than some integer parameter $L$. This allows the algorithm to assume that in each segment of $S$ of length $L$ there is at most one switch between distributions. Consequently, it can identify if both parts were generated almost solely by the same probability distribution. The initialization procedure is described in Figure 3.

We assume that the algorithm receives the parameters $K$ and $L$ as input. This assumption is removed (at some additional cost) in the next section.

The Proof of Theorem 2 is very similar to the proof of Theorem 1 and follows from the lemmas given below.

**Lemma 11 (Initialization error for general $N$)** *If for some $\rho < \Delta_2/14$, the minimal length $L$ of runs in $S$ satisfies*

$$\frac{L}{\ln L} \geq \frac{4\left(\ln M + D + \ln\frac{1}{\delta}\right)}{\rho^2} \ ,$$

*then, with probability at least $1-\delta$, the initialization procedure described in Figure 3 generates a set of estimates $B = \{\tilde{p}_1^0, \ldots, \tilde{p}_N^0\}$ which approximates $\{\vec{p}_1, \ldots, \vec{p}_N\}$ in the following sense. There exists a one-to-one mapping $\phi$ from $1..N$ to $1..N$ such that for all $1 \leq j \leq N$*

$$||\vec{p}_j - \tilde{p}_{\phi(j)}^0||_2 \leq 6\rho$$

**Proof:** The initialization procedure starts by considering all pairs of windows of the form $\sigma_i, \sigma_{i+1}, \ldots, \sigma_{i+\ell-1}$ and

[3] Ignoring the dependence on $\delta$.

**Initial estimates of the distributions. (general case)**

1. Set $\ell = \frac{1}{2}L$.
2. Set $\theta = 2\sqrt{\frac{2(\ln M + D\ln(\ell+1) + \ln\frac{1}{\delta})}{\ell}}$.
3. For each $i \in 1 \ldots M-\ell+1$ and each $\sigma \in [1, \ldots, D]$, let $f_i(\sigma)$ be the fraction of the elements in $\sigma_i, \sigma_{i+1}, \ldots, \sigma_{i+\ell-1}$ which are equal to $\sigma$. Let $\vec{f_i}$ denote the vector $\langle f_i(1), f_i(2), \ldots, f_i(D)\rangle$.
4. Let $A$ be the set of vectors $(\vec{f_i} + \vec{f}_{i+\ell})/2$ for all $1 \le i \le M-\ell$ such that $\|\vec{f_i} - \vec{f}_{i+\ell}\|_2 \le \theta$.
5. Start with $B$ as an empty set and repeat the following until $A$ becomes empty:
   - Select an element $\vec{f}$ from $A$ and add it to $B$.
   - Remove from $A$ all elements $\vec{g}$ such that $\|\vec{f} - \vec{g}\|_2 \le 3\theta$.
6. Output the set $B$ as the set of initial distribution estimates.

**Choice of cost vectors. (general case)**

- With each estimate $\tilde{p}_j^t$ associate the cost vector

$$\vec{c}_j^t(\sigma) \quad = \quad \sum_{\sigma' \neq \sigma} (\tilde{p}_j^t(\sigma'))^2 + (1 - \tilde{p}_j^t(\sigma))^2 = \sum_{\sigma'=1}^{D} (\tilde{p}_j^t(\sigma'))^2 + 1 - 2\tilde{p}_j^t(\sigma)$$

Figure 3: The initialization procedure and the choice of the cost vectors for the general case

$\sigma_{i+\ell}, \sigma_{i+\ell+1}, \ldots, \sigma_{i+2\ell-1}$. The assumption on the minimal length of runs in $\boldsymbol{\eta}$ implies that each such pair overlaps with at most one switch between runs in $\boldsymbol{\eta}$. We concentrate on some particular pair of windows, whose corresponding estimates are $\vec{f_i}$ and $\vec{f}_{i+\ell}$ and assume, without loss of generality, that the switch is in the second window. Let $\vec{p}_a$ be the distribution which generates the elements before the switch and $\vec{p}_b$ be the distribution after the switch. Then the expected value of $\vec{f_i}$ is $\vec{p}_a$ and the expected value of $\vec{f}_{i+\ell}$ is $(1-\alpha_i)\vec{p}_a + \alpha_i\vec{p}_b$ for some $0 \le \alpha_i \le 1$. Using Lemma 6 and our requirement on $L$ we show that, for all $1 \le i \le M-\ell$, the actual value of $\vec{f_i}$ which is associated with each window is close to its expected value. Specifically, with probability at least $1-\delta$, the $L_2$ distance between each estimate $\vec{f_i}$ and its expected value is at most $\rho = \theta/2$. We thus get that the pair $\vec{f_i}, \vec{f}_{i+\ell}$ is added to the set $A$ only if

$$\theta \ge \|\vec{f_i} - \vec{f}_{i+\ell}\|_2 \ \ge \alpha_i\|\vec{p}_a - \vec{p}_b\|_2 - \theta\ , \qquad (19)$$

so that

$$\alpha_i \le \frac{2\theta}{\|\vec{p}_a - \vec{p}_b\|_2}\ . \qquad (20)$$

This implies that the estimate that is added to the set $A$ in this case satisfies

$$\|(\vec{f_i} + \vec{f}_{i+\ell})/2 - \vec{p}_a\|_2 \le (\alpha_i/2)\|\vec{p}_a - \vec{p}_b\|_2 + \theta/2 \le \frac{3\theta}{2}\ .$$

Thus the estimates in the set $B$ are all within $(3/2)\theta$ of actual target distributions.

On the other hand, we are guaranteed that each run contains at least one pair of estimates that are both pure, and it is easy to check that the accuracy of the estimates $\vec{f_i}$ guarantees that this pair will be accepted into $A$. Thus each target distribution has a least one representative in $A$.

The goal of step 4 of the initialization procedure is to find a *single* representative for each target distribution. Simple arguments show that the distance between two different representatives of the same distribution is at most $3\theta$ and the distance between two representatives of two different distributions is at least $\|\vec{p}_a - \vec{p}_b\|_2 - 3\theta$. From the requirement on $L$ in the statement of the lemma we get that $\|\vec{p}_a - \vec{p}_b\|_2 - 3\theta \ge 4\theta$. Thus step 4 generates a set of distributions with one representative per target distribution, as required in the statement of the lemma. ■

**Lemma 12 (Segmentation error for general $N$)** *Let*

$$\rho = \min_{\psi} \max_{j \in [1,\ldots,N]} \|\tilde{\vec{p}}^t_{\psi(j)} - \vec{p}_j\|_2\ ,$$

*where $\psi$ ranges over all one-to-one mappings from $[1, \ldots, N]$ to $[1, \ldots, N]$. Assume that $\phi$ is the mapping that achieves the minimum. If $\rho < \Delta_2/14$, then with probability at least $1-\delta$ the segmentation $\tilde{\boldsymbol{\eta}}^t$ that minimizes the total cost satisfies*

$$\sum_{\phi(j) \neq j'} n_{j,j'}(\tilde{\boldsymbol{\eta}}^t) \le \frac{8(\ln\frac{1}{\delta} + K\ln(NM))}{(\Delta_2 - 2\rho)^4}$$

**Proof:** Assume, without loss of generality, that $\phi$ is the identity mapping. Assume some element in the sequence $S$ is generated by the distribution $\vec{p}_j$ and then assigned a cost $c$ from the cost vector $\vec{c}_{j'}^t$ which corresponds to the approximated distribution $\tilde{p}^t_{j'}$. Define $\vec{e} = \tilde{p}^t_{j'} - \vec{p}_j$. Then the expected value of $c$ is

$$\sum_{\sigma=1}^{D} \vec{p}_j(\sigma)\vec{c}_{j'}(\sigma)$$

$$= \sum_{\sigma=1}^{D} \left\{ \vec{p}_j(\sigma) \left( \sum_{\sigma'=1}^{D} (\tilde{p}^t_{j'}(\sigma'))^2 + 1 - 2\tilde{p}^t_{j'}(\sigma) \right) \right\}$$

$$= \sum_{\sigma=1}^{D} (\vec{p}_j(\sigma) + \vec{e}(\sigma))^2 + 1$$

$$
\begin{aligned}
&- \quad 2\sum_{\sigma=1}^{D} \vec{p}_j(\sigma)(\vec{p}_j(\sigma)+\vec{e}(\sigma)) \\
&= \quad 1-\sum_{\sigma=1}^{D}(\vec{p}_j(\sigma))^2+\sum_{\sigma=1}^{D}\vec{e}(\sigma)^2 \\
&= \quad 1-\|\vec{p}_j\|_2^2+\|\vec{e}\|_2^2 \ . \qquad (21)
\end{aligned}
$$

Thus the expected contribution of any element to the total cost is a sum of two terms. The first term depends only on the underlying target distribution, and the second term depends on the $L_2$ norm of the approximation error $\tilde{p}^t_{j'} - \vec{p}_j$.

Similarly to the analysis in the proof of Lemma 9 we now consider the difference between the total costs, corresponding to the approximate cost vectors, of two different segmentations. The first is the correct segmentation and the second is the best segmentation which minimizes the total cost on $S$. Clearly, only the elements on which $\boldsymbol{\eta}$ and $\tilde{\boldsymbol{\eta}}$ differ contribute to the difference in the total cost. From Equation (21) we get that the expected total difference is

$$
\begin{aligned}
&E_S\left[C\left(\{\vec{c}_j\}_{j=1}^N, \boldsymbol{\eta}, S\right) - C\left(\{\vec{c}_j\}_{j=1}^N, \boldsymbol{\eta}', S\right)\right] \\
&\quad = \sum_{j'\neq j} n_{j,j'}\left(\|\tilde{p}^t_j - \vec{p}_j\|_2^2 - \|\tilde{p}^t_{j'} - \vec{p}_j\|_2^2\right) \ . \qquad (22)
\end{aligned}
$$

Thus if $\|\tilde{p}^t_{j'} - \vec{p}_j\|_2 > \|\tilde{p}^t_j - \vec{p}_j\|_2$ for all $j' \neq j$, then the expected cost difference is guaranteed to be negative. Using the triangle inequality for the $L_2$ norm we find that the conditions on the minimal separation and on the maximal error imply that, $\forall j' \neq j, \|\tilde{p}^t_{j'} - \vec{p}_j\|_2 - \|\tilde{p}^t_j - \vec{p}_j\|_2 \geq \Delta_2 - 2\rho$. We thus get that the expected total difference in costs is bounded by

$$
\begin{aligned}
&E_S\left[C\left(\{\vec{c}_j\}_{j=1}^N, \boldsymbol{\eta}, S\right) - C\left(\{\vec{c}_j\}_{j=1}^N, \boldsymbol{\eta}', S\right)\right] \\
&\quad \leq -(\Delta_2 - 2\rho)^2 \sum_{j'\neq j} n_{j,j'} \ . \qquad (23)
\end{aligned}
$$

We want to bound the probability that a segmentation with many errors has a total cost which is smaller than that of the correct segmentation. The cost difference is a sum of the cost differences in the places where the segmentations disagree. These are independent random variables. It is easy to check that the coordinates of any cost vector are bounded in the range $[0, 1]$ and thus the cost difference is bounded in $[-1, 1]$. We can thus apply Lemma 5 and get that for any individual segmentation, the probability that the segmentation achieves smaller total cost than the correct segmentation is upper bounded by

$$
\begin{aligned}
&Pr_S\left[C\left(\{\vec{c}_j\}_{j=1}^N, \boldsymbol{\eta}', S\right) < C\left(\{\vec{c}_j\}_{j=1}^N, \boldsymbol{\eta}, S\right)\right] \\
&\quad \leq \exp\left(-\frac{1}{8}(\Delta_2 - 2\rho)^4 \sum_{j'\neq j} n_{j,j'}\right) \ . \qquad (24)
\end{aligned}
$$

There are less than $M^K N^K$ segmentations with $K$ runs. Combining this with the last equation we get the statement of the lemma. ■

**Lemma 13 (Reevaluation error for general $N$)** *Suppose $\phi : [1, \ldots, N] \to [1, \ldots, N]$ is a one to one mapping such that*

$$
\beta = \frac{\max_{\phi(j)\neq j'}(n_{j,j'}(\tilde{\boldsymbol{\eta}^t}))}{M} \ .
$$

*If $\beta < \gamma$, then with probability at least $1 - \delta$, there exists a one-to-one mapping $\phi : [1, \ldots, N] \to [1, \ldots, N]$, such that for every $j \in [1, \ldots, N]$*

$$
\|\tilde{p}^{t+1}_{\phi(j)} - \vec{p}_j\|_2 \leq \frac{\beta}{\gamma - \beta}\Delta_2 + var \ ,
$$

*where*

$$
var = \sqrt{\frac{2(K\ln(NM) + D\ln(M+1) + \ln(1/\delta)}{(\gamma - \beta)M}} \ .
$$

The Proof of Lemma 13 is the same as the proof of Lemma 10 except for the use of the $L_2$ norm in place of the $L_1$ norm.

## 8 Treating Unspecified parameters

So far we have assumed that our learning algorithms receive as input parameters that describe properties of the sequence $S$. In the two distribution case the algorithm receives as input $K$ and $\gamma$, and in the multiple distribution case the algorithm receives $K$ and $L$. In this section we show that these assumptions can be removed, with some increase in the error, if we measure the error of the hypothesis with $err_3$. Recall that $err_3$ is always smaller than $err_1$ and $err_2$, thus, the bounds we previously got for cases where the algorithm receives additional input, can be used here.

The idea is to try all possible settings of the unknown parameters, and then select the best resulting hypothesis. While the different algorithms receive as input different subsets of $\gamma$, $K$, and $L$, the only variable that is controlled by these parameters is $\ell$, the length of the segments that are used for initialization. As there are at most $M$ possible settings for $\ell$, we need to run the algorithm $M$ times and then compare the hypotheses. The different hypotheses are compared in terms of the total cost defined in Figure 3 and the hypothesis with the minimal total cost is selected.

More formally, assume that we have $m$ hypotheses

$$
\left\{\left(\tilde{\boldsymbol{\eta}}_r, \{\tilde{p}_{r,j}\}_{j=1}^N\right)\right\}_{r=1}^m \ .
$$

For each $1 \leq r \leq m$ we calculate the total cost

$$
C\left(\{\vec{c}_{r,j}\}_{j=1}^N, \tilde{\boldsymbol{\eta}}_r, S\right)
$$

according to the cost vectors $\{\vec{c}_{r,j}\}_{j=1}^N$ as defined in Figure 3 for the case of more than two distributions. We then select the hypothesis with the minimal total cost as our final hypothesis.

The following lemma shows that the error of the selected hypothesis is, with high probability, only slightly larger than that of the best hypothesis in the set.

**Lemma 14** *Let $S$ be a sequence generated according to the segmentation $\boldsymbol{\eta}$ and the distributions $\vec{p}_1, \ldots, \vec{p}_N$. Let*

$$
\left\{\left(\tilde{\boldsymbol{\eta}}_r, \{\tilde{p}_{r,j}\}_{j=1}^N\right)\right\}_{r=1}^m
$$

*be a set of hypotheses and let $u = \mathrm{argmin}_r C(\{\vec{c}_{r,j}\}_{j=1}^N, \tilde{\boldsymbol{\eta}}_r, S)$. Then, with probability at least $1 - \delta$ with respect to the distribution of $S$*

$$
err_3(\tilde{\boldsymbol{\eta}}_u, \{\tilde{p}_{u,j}\}_{j=1}^N) \leq \min_{r=1..m} err_3(\tilde{\boldsymbol{\eta}}_r, \{\tilde{p}_{r,j}\}_{j=1}^N) + var \ ,
$$

*where*

$$
var = \sqrt{\frac{2\ln m + \ln(1/\delta) + K\ln N + K\ln M}{M}} \ .
$$

**Proof:** We first consider the expected value of the total cost of the sequence $S$ for some fixed hypothesis $(\tilde{\boldsymbol{\eta}}, \{\tilde{p}_j\}_{j=1}^N)$. Let $\sigma_i$ be the $i$th element in $S$. Thus $\sigma_i$ is generated by the distribution $\vec{p}_{\eta(i)}$ and then assigned a cost $c(i)$ according to the cost vector $\vec{c}_{\tilde{\eta}(i)}$ which corresponds to the approximated distribution $\tilde{p}_{\tilde{\eta}(i)}$. Define $\vec{e}_i = \vec{p}_{\eta(i)} - \tilde{p}_{\tilde{\eta}(i)}$. Then similarly to Equation (21), the expected value of $c(i)$ is

$$E[c(i)] = 1 - ||\vec{p}_{\eta(i)}||_2^2 + ||\vec{e}_i||_2^2 \,. \qquad (25)$$

Summing the expected value of $c(i)$ over $i = 1 \ldots n$, we get that the expected total cost of $S$ is $A + \sum_{i=1}^n ||\vec{e}_i||_2^2$. Where $A$ is a constant independent of the hypothesis. The range of all of the $c(i)$'s is $[0,1]$ and they are independent random variables. Thus the deviation of the cost of the sequence from its expected value can be bounded using Lemma 5 as follows

$$Pr\left[\left|\sum_{i=1}^M c(i) - E(\sum_{i=1}^M c(i))\right| > a\right] \leq \exp\left(-\frac{2a^2}{M}\right) \,.$$

We now return to the set of $m$ hypotheses. As each of these hypotheses is selected from a set of at most $(NM)^K$ segmentations, we find that by setting $a$ to be $\sqrt{(M/2)\ln(m(NM)^K/\delta)}$ we get that with probability at least $1-\delta$ the total costs of all $m(NM)^K$ possible hypotheses are within $a$ of their respective expected values. Thus the expected total cost of the segmentation that appears to be best is within $2a$ of the expected value of the actual best segmentation. Which gives the statement of the lemma. ■

Applying Lemma 14 to Theorem 1 we get Theorem 3. In this case we add to the $M$ candidate hypotheses that correspond to the choices of $\ell$ the hypothesis $(\boldsymbol{\eta}_{single}, \{\tilde{p}\})$, where $\boldsymbol{\eta}_{single}$ consists of a single run, and $\tilde{p}$ is the empirical distribution of the sequence. This hypothesis is accurate if either $\gamma$ or $\Delta_1$ are smaller than $\epsilon/8$, and thus we can remove any assumption on $\gamma$ and $\Delta_1$. Applying Lemma 14 to Theorem 2 we get Theorem 4. In this case we still need to assume a lower bound on $L$ in order to assure that at least one of the executions of the algorithm succeeds.

In both cases the requirement on $M$ that we have to make in order that the extra error be smaller than $\epsilon$, is

$$M = \Omega\left(1/\epsilon^2(K\ln N + \ln(1/\delta))\right)$$

. This requirement is subsumed by the requirements in Theorems 1 and 2.

## 9 Open Problems

It is clear that improvements in our upper bounds should be possible. In addition it would be interesting to search for matching lower bounds, since no lower bounds on the achievable error are known.

It would be interesting to show whether our bounds can be improved if we use the likelihood cost as done in the Baum-Welch algorithm rather than the costs we invented.

A natural scaling of the parameters is to fix $K/M$, the switching rate, and to let $M \to \infty$. Our analysis breaks down in this case.


### Acknowledgments

We would like to thank Yoram Singer and Tali Tishby for discussions that motivated this work. Dana Ron would like to thank the Eshkol Fellowship for its support. Part of this work was done while Dana Ron was visiting AT&T Bell Laboratories.


## References


[1] N. Abe and M. K. Warmuth. On the computational complexity of approximating distributions by probabilistic automata. *Machine Learning*, 9(2/3), 1992. Special issue for COLT90.

[2] Leonard E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology. *Bulletin of the American Mathematical Society*, 73:360–363, 1967.

[3] Avrim Blum and Prasad Chalasani. Learning switching concepts. In *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, pages 231–242, July 1992.

[4] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 1991.

[5] David Gillman and Michael Sipser. Inference and minimization of hidden markov chains. In *Proceedings of the Seventh Annual ACM Conference on Computational Learning Theory*, pages 147–158, July 1994.

[6] Nick Littlestone. Notes on the derivation of chernoff-type bounds for sums of random variables. Unpublished Manuscript, 1990.

[7] L. R. Rabiner and B. H. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(1):4–16, January 1986.

[8] A.J. Viterbi. Error bounds for convulutional codes and an asymptotically optimal decoding algorithm. *IEEE Trans. Inform. Theory*, 13:260–269, 1967.