A Short Tutorial on the Expectation-Maximization Algorithm

Detlef Prescher
Institute for Logic, Language and Computation
University of Amsterdam
prescher@science.uva.nl

1 Introduction

The paper gives a brief review of the expectation-maximization algorithm (Dempster, Laird, and Rubin 1977) in the comprehensible framework of discrete mathematics. In Section 2, two prominent estimation methods, the relative-frequency estimation and the maximum-likelihood estimation are presented. Section 3 is dedicated to the expectation-maximization algorithm and a simpler variant, the generalized expectation-maximization algorithm. In Section 4, two loaded dice are rolled. Enjoy!

2 Estimation Methods

A statistics problem is a problem in which a **corpus**[1] that has been generated in accordance with some unknown **probability distribution** must be analyzed and some type of inference about the unknown distribution must be made. In other words, in a statistics problem there is a choice between two or more probability distributions which might have generated the corpus. In practice, there are often an infinite number of different possible distributions – statisticians bundle these into one single **probability model** – which might have generated the corpus. By analyzing the corpus, an attempt is made to learn about the unknown distribution. So, on the basis of the corpus, an **estimation method** selects one **instance** of the probability model, thereby aiming at finding the original distribution. In this section, two common estimation methods, the relative-frequency and the maximum-likelihood estimation, are presented.

Corpora

Definition 1 *Let $\mathcal{X}$ be a countable set. A real-valued function $f\colon \mathcal{X} \to \mathcal{R}$ is called a **corpus**, if f's values are non-negative numbers*

$$f(x) \geq 0 \quad \text{for all } x \in \mathcal{X}$$

[1] Statisticians use the term *sample* but computational linguists prefer the term *corpus*

Each $x \in \mathcal{X}$ is called a ***type****, and each value of f is called a* ***type frequency****. The* ***corpus size***[2] *is defined as*

$$|f| = \sum_{x \in \mathcal{X}} f(x)$$

Finally, a corpus is called ***non-empty*** *and* ***finite*** *if*

$$0 < |f| < \infty$$

In this definition, type frequencies are defined as *non-negative real numbers*. The reason for not taking *natural numbers* is that some statistical estimation methods define type frequencies as *weighted* occurrence frequencies (which are not natural but non-negative real numbers). Later on, in the context of the EM algorithm, this point will become clear. Note also that a *finite corpus* might consist of an *infinite number of types* with positive frequencies. The following definition shows that Definition 1 covers the standard notion of the term *corpus* (used in Computational Linguistics) and of the term *sample* (used in Statistics).

Definition 2 *Let $x_1, \ldots, x_n$ be a finite sequence of type instances from $\mathcal{X}$. Each x_i of this sequence is called a* ***token****. The* ***occurrence frequency*** *of a type x in the sequence is defined as the following count*

$$f(x) = |\{\ i \mid x_i = x\}|$$

Obviously, f is a corpus in the sense of Definition 1, and it has the following properties: The type x does not occur in the sequence if $f(x) = 0$; In any other case there are $f(x)$ tokens in the sequence which are identical to x. Moreover, the corpus size $|f|$ is identical to n, the number of tokens in the sequence.

Relative-Frequency Estimation

Let us first present the notion of probability that we use throughout this paper.

Definition 3 *Let $\mathcal{X}$ be a countable set of types. A real-valued function $p\colon \mathcal{X} \to \mathcal{R}$ is called a* ***probability distribution on*** *$\mathcal{X}$, if p has two properties: First, p's values are non-negative numbers*

$$p(x) \geq 0 \quad \text{for all } x \in \mathcal{X}$$

and second, p's values sum to 1

$$\sum_{x \in \mathcal{X}} p(x) = 1$$

Readers familiar to probability theory will certainly note that we use the term *probability distribution* in a sloppy way (Duda et al. (2001), page 611, introduce the term *probability mass function* instead). Standardly, probability distributions allocate a probability value $p(A)$ to subsets $A \subseteq \mathcal{X}$, so-called **events** of an **event space** $\mathcal{X}$, such that three specific axioms are satisfied (see e.g. DeGroot (1989)):

Axiom 1 *$p(A) \geq 0$ for any event A.*

[2]Note that the corpus size $|f|$ is well-defined: The order of summation is not relevant for the value of the (possible infinite) series $\sum_{x \in \mathcal{X}} f(x)$, since the types are countable and the type frequencies are non-negative numbers

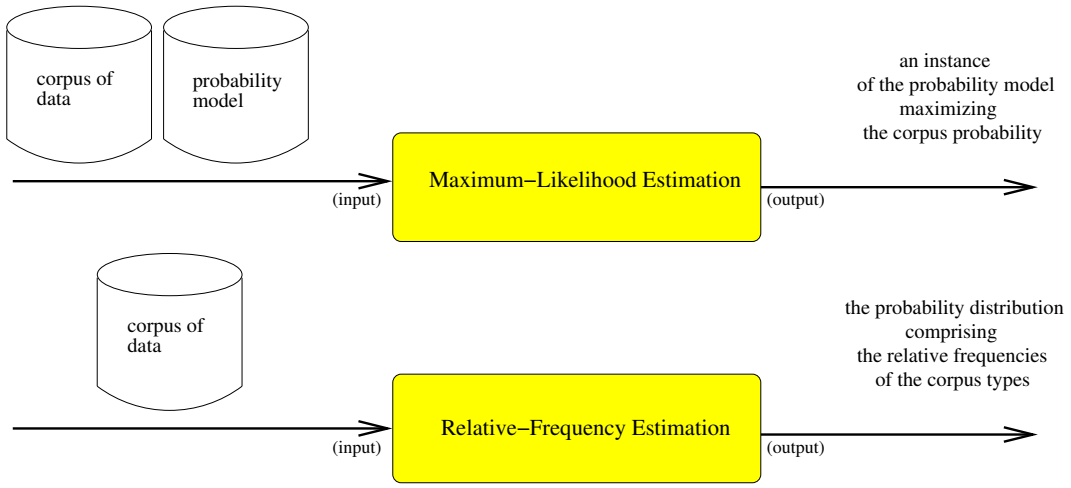

Figure 1: Maximum-likelihood estimation and relative-frequency estimation

Axiom 2 $p(\mathcal{X}) = 1$.

Axiom 3 $p(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} p(A_i)$ *for any infinite sequence of disjoint events* $A_1, A_2, A_3, ...$

Now, however, note that the probability distributions introduced in Definition 3 induce rather naturally the following probabilities for events $A \subseteq \mathcal{X}$

$$p(A) := \sum_{x \in A} p(x)$$

Using the properties of $p(x)$, we can easily show that the probabilities $p(A)$ satisfy the three axioms of probability theory. So, Definition 3 is justified and thus, for the rest of the paper, we are allowed to put axiomatic probability theory out of our minds.

Definition 4 *Let f be a non-empty and finite corpus. The probability distribution*

$$\tilde{p}\colon \mathcal{X} \to [0,1] \qquad \textit{where} \qquad \tilde{p}(x) = \frac{f(x)}{|f|}$$

is called the ***relative-frequency estimate*** *on f.*

The relative-frequency estimation is the most comprehensible estimation method and has some nice properties which will be discussed in the context of the more general maximum-likelihood estimation. For now, however, note that $\tilde{p}$ is well defined, since both $|f| > 0$ and $|f| < \infty$. Moreover, it is easy to check that $\tilde{p}$'s values sum to one: $\sum_{x \in \mathcal{X}} \tilde{p}(x) = \sum_{x \in \mathcal{X}} |f|^{-1} \cdot f(x) = |f|^{-1} \cdot \sum_{x \in \mathcal{X}} f(x) = |f|^{-1} \cdot |f| = 1$.

Maximum-Likelihood Estimation

Maximum-likelihood estimation was introduced by R. A. Fisher in 1912, and will typically yield an excellent estimate if the given corpus is large. Most notably, maximum-likelihood estimators fulfill the so-called **invariance principle** and, under certain conditions which

are typically satisfied in practical problems, they are even **consistent estimators** (DeGroot 1989). For these reasons, maximum-likelihood estimation is probably the most widely used estimation method.

Now, unlike relative-frequency estimation, maximum-likelihood estimation is a fully-fledged estimation method that aims at selecting an instance of a **given probability model** which might have originally generated the given corpus. By contrast, the relative-frequency estimate is defined on the basis of a corpus only (see Definition 4). Figure 1 reveals the conceptual difference of both estimation methods. In what follows, we will pay some attention to describe the single setting, in which we are exceptionally allowed to mix up both methods (see Theorem 1). Let us start, however, by presenting the notion of a probability model.

Definition 5 *A non-empty set $\mathcal{M}$ of probability distributions on a set $\mathcal{X}$ of types is called a* ***probability model on*** *$\mathcal{X}$. The elements of $\mathcal{M}$ are called* ***instances of the model*** *$\mathcal{M}$. The* ***unrestricted probability model*** *is the set $\mathcal{M}(\mathcal{X})$ of all probability distributions on the set of types*

$$\mathcal{M}(\mathcal{X}) = \left\{ p\colon \mathcal{X} \to [0,1] \;\middle|\; \sum_{x \in \mathcal{X}} p(x) = 1 \right\}$$

A probability model $\mathcal{M}$ is called ***restricted*** *in all other cases*

$$\mathcal{M} \subseteq \mathcal{M}(\mathcal{X}) \quad \text{and} \quad \mathcal{M} \neq \mathcal{M}(\mathcal{X})$$

In practice, most probability models are restricted since their instances are often defined on a set $\mathcal{X}$ comprising multi-dimensional types such that certain parts of the types are statistically independent (see examples 4 and 5). Here is another side note: We already checked that the relative-frequency estimate is a probability distribution, meaning in terms of Definition 5 that *the relative-frequency estimate is an instance of the unrestricted probability model.* So, from an extreme point of view, the relative-frequency estimation might be also regarded as a fully-fledged estimation method exploiting a corpus **and** a probability model (namely, the unrestricted model).

In the following, we define maximum-likelihood estimation as a method that aims at finding an instance of a given model which maximizes the probability of a given corpus. Later on, we will see that maximum-likelihood estimates have an additional property: They are the instances of the given probability model that have a "minimal distance" to the relative frequencies of the types in the corpus (see Theorem 2). So, indeed, maximum-likelihood estimates can be intuitively thought of in the intended way: They are the instances of the probability model that might have originally generated the corpus.

Definition 6 *Let f be a non-empty and finite corpus on a countable set $\mathcal{X}$ of types. Let $\mathcal{M}$ be a probability model on $\mathcal{X}$. The* ***probability of the corpus*** *allocated by an instance p of the model $\mathcal{M}$ is defined as*

$$L(f;p) = \prod_{x \in \mathcal{X}} p(x)^{f(x)}$$

An instance $\hat{p}$ of the model $\mathcal{M}$ is called a ***maximum-likelihood estimate of*** *$\mathcal{M}$* ***on*** *f, if and only if the corpus f is allocated a maximum probability by $\hat{p}$*

$$L(f;\hat{p}) = \max_{p \in \mathcal{M}} L(f;p)$$

(Based on continuity arguments, we use the convention that $p^0 = 1$ and $0^0 = 1$.)

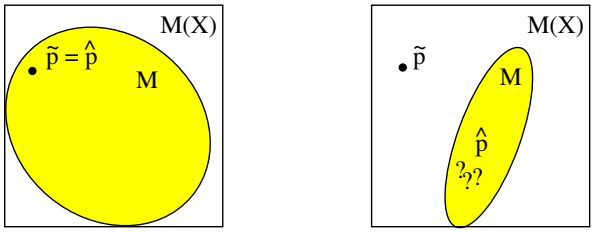

Figure 2: Maximum-likelihood estimation and relative-frequency estimation yield for some "exceptional" probability models the *same estimate*. These models are lightly restricted or even unrestricted models that contain an instance comprising the relative frequencies of all corpus types (left-hand side). In practice, however, most probability models will not behave like that. So, maximum-likelihood estimation and relative-frequency estimation yield in most cases *different estimates*. As a further and more serious consequence, the maximum-likelihood estimates have then to be searched for by genuine optimization procedures (right-hand side).

By looking at this definition, we recognize that maximum-likelihood estimates are the solutions of a quite complex optimization problem. So, some nasty questions about maximum-likelihood estimation arise:

> **Existence** Is there for any probability model and any corpus a maximum-likelihood estimate of the model on the corpus?
>
> **Uniqueness** Is there for any probability model and any corpus a unique maximum-likelihood estimate of the model on the corpus?
>
> **Computability** For which probability models and corpora can maximum-likelihood estimates be efficiently computed?

For some probability models $\mathcal{M}$, the following theorem gives a positive answer.

Theorem 1 *Let f be a non-empty and finite corpus on a countable set $\mathcal{X}$ of types. Then:*

(i) The relative-frequency estimate $\tilde{p}$ is a unique maximum-likelihood estimate of the unrestricted probability model $\mathcal{M}(\mathcal{X})$ on f.

(ii) The relative-frequency estimate $\tilde{p}$ is a maximum-likelihood estimate of a (restricted or unrestricted) probability model $\mathcal{M}$ on f, if and only if $\tilde{p}$ is an instance of the model $\mathcal{M}$. In this case, $\tilde{p}$ is a unique maximum-likelihood estimate of $\mathcal{M}$ on f.

Proof Ad (i): Combine theorems 2 and 3. Ad (ii): "$\Rightarrow$" is trivial. "$\Leftarrow$" by (i) **q.e.d.**

On a first glance, proposition (ii) seems to be more general than proposition (i), since proposition (i) is about one single probability model, the unrestricted model, whereas proposition (ii) gives some insight about the relation of the relative-frequency estimate to a maximum-likelihood estimate of arbitrary restricted probability models (see also Figure 2). Both propositions, however, are equivalent. As we will show later on, proposition (i) is equivalent to the famous information inequality of information theory, for which various proofs have been given in the literature.

Example 1 *On the basis of the following corpus*

$$f(a) = 2,\ f(b) = 3,\ f(c) = 5$$

we shall calculate the maximum-likelihood estimate of the unrestricted probability model $\mathcal{M}(\{a, b, c\})$*, as well as the maximum-likelihood estimate of the restricted probability model*

$$\mathcal{M} = \Big\{ p \in \mathcal{M}(\{a, b, c\}) \ \Big|\ p(a) = 0.5 \Big\}$$

The solution is instructive, but is left to the reader.

The Information Inequality of Information Theory

Definition 7 *The* ***relative entropy*** $D(p \mid\mid q)$ *of the probability distribution* p *with respect to the probability distribution* q *is defined by*

$$D(p \mid\mid q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)}$$

(Based on continuity arguments, we use the convention that $0 \log \frac{0}{q} = 0$ *and* $p \log \frac{p}{0} = \infty$ *and* $0 \log \frac{0}{0} = 0$*. The logarithm is calculated with respect to the base 2.)*

Connecting maximum-likelihood estimation with the concept of relative entropy, the following theorem gives the important insight that the relative-entropy of the relative-frequency estimate is minimal with respect to a maximum-likelihood estimate.

Theorem 2 *Let* $\tilde{p}$ *be the relative-frequency estimate on a non-empty and finite corpus* f*, and let* $\mathcal{M}$ *be a probability model on the set* $\mathcal{X}$ *of types. Then: An instance* $\hat{p}$ *of the model* $\mathcal{M}$ *is a maximum-likelihood estimate of* $\mathcal{M}$ *on* f*, if and only if the relative-entropy of* $\tilde{p}$ *is minimal with respect to* $\hat{p}$

$$D(\tilde{p} \mid\mid \hat{p}) = \min_{p \in \mathcal{M}} D(\tilde{p} \mid\mid p)$$

Proof First, the relative entropy $D(\tilde{p} \mid\mid p)$ is simply the difference of two further entropy values, the so-called **cross-entropy** $H(\tilde{p}; p) = -\sum_{x \in X} \tilde{p}(x) \log p(x)$ and the **entropy** $H(\tilde{p}) = -\sum_{x \in X} \tilde{p}(x) \log \tilde{p}(x)$ of the relative-frequency estimate

$$D(\tilde{p} \mid\mid p) = H(\tilde{p}; p) - H(\tilde{p})$$

(Based on continuity arguments and in full agreement with the convention used in Definition 7, we use here that $\tilde{p} \log 0 = -\infty$ and $0 \log 0 = 0$.) It follows that minimizing the relative entropy is equivalent to minimizing the cross-entropy (as a function of the instances p of the given probability model $\mathcal{M}$). The cross-entropy, however, is proportional to the negative log-probability of the corpus f

$$H(\tilde{p}; p) = -\frac{1}{|f|} \log L(f; p)$$

So, finally, minimizing the relative entropy $D(\tilde{p} \mid\mid p)$ is equivalent to maximizing the corpus probability $L(f;p)$. [3]

Together with Theorem 2, the following theorem, the so-called information inequality of information theory, proves Theorem 1. The information inequality states simply that the relative entropy is a non-negative number – which is zero, if and only if the two probability distributions are equal.

Theorem 3 (Information Inequality) *Let p and q be two probability distributions. Then*

$$D(p \mid\mid q) \geq 0$$

with equality if and only if $p(x) = q(x)$ for all $x \in \mathcal{X}$.

Proof See, e.g., Cover and Thomas (1991), page 26.

*Maximum-Entropy Estimation

Readers only interested in the expectation-maximization algorithm are encouraged to omit this section. For completeness, however, note that the relative entropy is **asymmetric**. That means, in general

$$D(p||q) \neq D(q||p)$$

It is easy to check that the triangle inequality is not valid too. So, the relative entropy $D(.||.)$ is not a "true" distance function. On the other hand, $D(.||.)$ has some of the properties of a distance function. In particular, it is always non-negative and it is zero if and only if $p = q$ (see Theorem 3). So far, however, we aimed at minimizing the relative entropy with respect to its **second** argument, filling the first argument slot of $D(.||.)$ with the relative-frequency estimate $\tilde{p}$. Obviously, these observations raise the question, whether it is also possible to derive other "good" estimates by minimizing the relative entropy with respect to its **first** argument. So, in terms of Theorem 2, it might be interesting to ask for model instances $p^* \in \mathcal{M}$ with

$$D(p^*||\tilde{p}) = \min_{p \in \mathcal{M}} D(p||\tilde{p})$$

For at least two reasons, however, this initial approach of relative-entropy estimation is too simplistic. First, it is tailored to probability models that lack any generalization power. Second, it does not provide deeper insight when estimating constrained probability models. Here are the details:

[3] For completeness, note that the **perplexity** of a corpus f allocated by a model instance p is defined as $\text{perp}(f;p) = 2^{H(\tilde{p};p)}$. This yields $\text{perp}(f;p) = \sqrt[|f|]{\frac{1}{L(f;p)}}$ and $L(f;p) = \left(\frac{1}{\text{perp}(f;p)}\right)^{|f|}$ as well as the common interpretation that **the perplexity value measures the complexity of the given corpus from the model instance's view**: the perplexity is equal to the size of an imaginary word list from which the corpus can be generated by the model instance – assuming that all words on this list are equally probable. Moreover, the equations state that minimizing the corpus perplexity $\text{perp}(f;p)$ is equivalent to maximizing the corpus probability $L(f;p)$.

- A closer look at Definition 7 reveals that the relative entropy $D(p||\tilde{p})$ is finite for those model instances $p \in \mathcal{M}$ only that fulfill

$$\tilde{p}(x) = 0 \Rightarrow p(x) = 0$$

  So, the initial approach would lead to model instances that are completely unable to generalize, since they are not allowed to allocate positive probabilities to at least some of the types not seen in the training corpus.

- Theorem 2 guarantees that the relative-frequency estimate $\tilde{p}$ is a solution to the initial approach of relative-entropy estimation, whenever $\tilde{p} \in \mathcal{M}$. Now, Definition 8 introduces the constrained probability models M_{constr}, and indeed, it is easy to check that $\tilde{p}$ is always an instance of these models. In other words, estimating constrained probability models by the approach above does not result in interesting model instances.

Clearly, all the mentioned drawbacks are due to the fact that the relative-entropy minimization is performed with respect to the relative-frequency estimate. As a resource, we switch simply to a more convenient reference distribution, thereby generalizing formally the initial problem setting. So, as the final request, we ask for model instances $p^* \in \mathcal{M}$ with

$$D(p^*||p_0) = \min_{p \in \mathcal{M}} D(p||p_0)$$

In this setting, the **reference distribution** $p_0 \in \mathcal{M}(\mathcal{X})$ is a given instance of the unrestricted probability model, and from what we have seen so far, p_0 should allocate all types of interest a positive probability, and moreover, p_0 should not be itself an instance of the probability model $\mathcal{M}$. Indeed, this request will lead us to the interesting maximum-entropy estimates. Note first, that

$$D(p||p_0) = H(p; p_0) - H(p)$$

So, *minimizing* $D(p||p_0)$ as a function of the model instances p is equivalent to *minimizing* the cross entropy $H(p; p_0)$ and *simultaneously maximizing* the model entropy $H(p)$. Now, simultaneous optimization is a hard task in general, and this gives reason to focus firstly on maximizing the entropy $H(p)$ in isolation. The following definition presents maximum-entropy estimation in terms of the well-known maximum-entropy principle (Jaynes 1957). Sloppily formulated, the maximum-entropy principle recommends to maximize the entropy $H(p)$ as a function of the instances p of certain "constrained" probability models.

Definition 8 *Let $f_1, \ldots, f_d$ be a finite number of real-valued functions on a set $\mathcal{X}$ of types, the so-called **feature functions**[4]. Let $\tilde{p}$ be the relative-frequency estimate on a non-empty*

[4]Each of these feature functions can be thought of as being constructed by inspecting the set of types, thereby measuring a specific property of the types $x \in \mathcal{X}$. For example, if working in a formal-grammar framework, then it might be worthy to look (at least) at some feature functions f_r directly associated to the rules r of the given formal grammar. The "measure" $f_r(x)$ of a specific rule r for the analyzes $x \in \mathcal{X}$ of the grammar might be calculated, for example, in terms of the occurrence frequency of r in the sequence of those rules which are necessary to produce x. For instance, Chi (1999) studied this approach for the context-free grammar formalism. Note, however, that there is in general no recipe for constructing "good" feature functions: Often, it is really an intellectual challenge to find those feature functions that describe the given data as best as possible (or at least in a satisfying manner).

and finite corpus f on $\mathcal{X}$. Then, the ***probability model constrained by the expected values of*** $f_1 \dots f_d$ ***on*** f *is defined as*

$$\mathcal{M}_{constr} = \left\{ p \in \mathcal{M}(\mathcal{X}) \;\middle|\; E_p f_i = E_{\tilde{p}} f_i \;\; \textit{for } i = 1, \dots, d \right\}$$

Here, each $E_p f_i$ is the ***model instance's expectation*** *of f_i*

$$E_p f_i = \sum_{x \in \mathcal{X}} p(x) f_i(x)$$

constrained to match $E_{\tilde{p}} f_i$, the ***observed expectation*** *of f_i*

$$E_{\tilde{p}} f_i = \sum_{x \in \mathcal{X}} \tilde{p}(x) f_i(x)$$

Furthermore, a model instance $p^* \in \mathcal{M}_{constr}$ is called a ***maximum-entropy estimate of*** $\mathcal{M}_{constr}$ *if and only if*

$$H(p^*) = \max_{p \in \mathcal{M}_{constr}} H(p)$$

It is well-known that the maximum-entropy estimates have some nice properties. For example, as Definition 9 and Theorem 4 show, they can be identified to be the unique maximum-likelihood estimates of the so-called exponential models (which are also known as log-linear models).

Definition 9 *Let $f_1, \dots, f_d$ be a finite number of feature functions on a set $\mathcal{X}$ of types. The* ***exponential model of*** $f_1, \dots, f_d$ *is defined by*

$$\mathcal{M}_{exp} = \left\{ p \in \mathcal{M}(\mathcal{X}) \;\middle|\; p(x) = \frac{1}{Z_\lambda} e^{\lambda_1 f_1(x) + \dots + \lambda_d f_d(x)} \;\; \textit{with } \lambda_1, \dots, \lambda_d, Z_\lambda \in \mathcal{R} \right\}$$

Here, the ***normalizing constant*** Z_λ *(with λ as a short form for the sequence $\lambda_1, \dots, \lambda_d$) guarantees that $p \in \mathcal{M}(\mathcal{X})$, and it is given by*

$$Z_\lambda = \sum_{x \in \mathcal{X}} e^{\lambda_1 f_1(x) + \dots + \lambda_d f_d(x)}$$

Theorem 4 *Let f be a non-empty and finite corpus, and $f_1, \dots, f_d$ be a finite number of feature functions on a set $\mathcal{X}$ of types. Then*

(i) The maximum-entropy estimates of $\mathcal{M}_{constr}$ are instances of $\mathcal{M}_{exp}$, and the maximum-likelihood estimates of $\mathcal{M}_{exp}$ on f are instances of $\mathcal{M}_{constr}$.

(ii) If $p^* \in \mathcal{M}_{constr} \cap \mathcal{M}_{exp}$, then p^* is both a unique maximum-entropy estimate of $\mathcal{M}_{constr}$ and a unique maximum-likelihood estimate of $\mathcal{M}_{exp}$ on f.

Part (i) of the theorem simply suggests the form of the maximum-entropy or maximum-likelihood estimates we are looking for. By combining both findings of (i), however, the search space is drastically reduced for both estimation methods: We simply have to look at the intersection of the involved probability models. In turn, exactly this fact makes the second part of the theorem so valuable. If there is a maximum-entropy **or** a maximum-likelihood

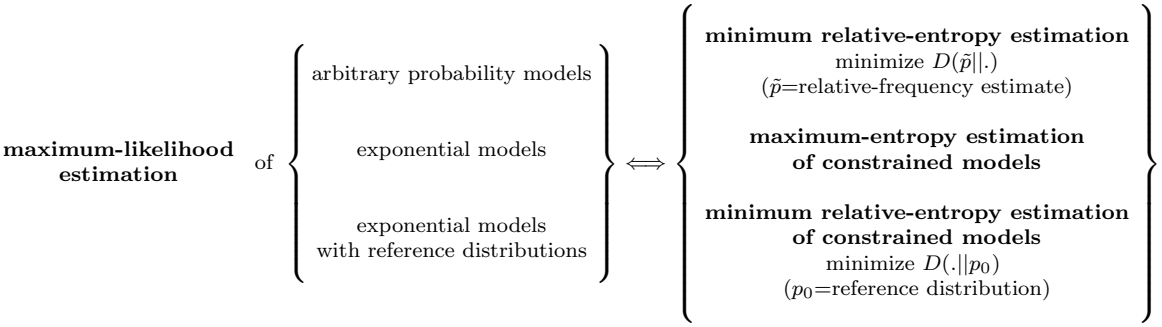

Figure 3: Maximum-likelihood estimation generalizes maximum-entropy estimation, as well as both variants of minimum relative-entropy estimation (where either the first or the second argument slot of $D(.||.)$ is filled by a given probability distribution).

estimate, then it is in the intersection of both models, and thus according to Part (ii), it is a unique estimate, and even more, it is both a maximum-entropy **and** a maximum-likelihood estimate.

Proof See e.g. Cover and Thomas (1991), pages 266-278. For an interesting alternate proof of (ii), see Ratnaparkhi (1997). Note, however, that the proof of Ratnaparkhi's Theorem 1 is incorrect, whenever the set $\mathcal{X}$ of types is infinite. Although Ratnaparkhi's proof is very elegant, it relies on the existence of a uniform distribution on $\mathcal{X}$ that simply does not exist in this special case. By contrast, Cover and Thomas prove Theorem 11.1.1 without using a uniform distribution on $\mathcal{X}$, and so, they achieve indeed the more general result.

Finally, we are coming back to our request of minimizing the relative entropy with respect to a given reference distribution $p_0 \in \mathcal{M}(\mathcal{X})$. For constrained probability models, the relevant results differ not much from the results described in Theorem 4. So, let

$$\mathcal{M}_{exp \cdot ref} = \left\{ p \in \mathcal{M}(\mathcal{X}) \;\middle|\; p(x) = \frac{1}{Z_\lambda} e^{\lambda_1 f_1(x) + \ldots + \lambda_d f_d(x)} \cdot p_0(x) \text{ with } \lambda_1, \ldots, \lambda_d, Z_\lambda \in \mathcal{R} \right\}$$

Then, along the lines of the proof of Theorem 4 it can be also proven that the following propositions are valid.

(i) The minimum relative-entropy estimates of $\mathcal{M}_{constr}$ are instances of $\mathcal{M}_{exp \cdot ref}$, and the maximum-likelihood estimates of $\mathcal{M}_{exp \cdot ref}$ on f are instances of $\mathcal{M}_{constr}$.

(ii) If $p^* \in \mathcal{M}_{constr} \cap \mathcal{M}_{exp \cdot ref}$, then p^* is both a unique minimum relative-entropy estimate of $\mathcal{M}_{constr}$ and a unique maximum-likelihood estimate of $\mathcal{M}_{exp \cdot ref}$ on f.

All results are displayed in Figure 3.

3 The Expectation-Maximization Algorithm

The expectation-maximization algorithm was introduced by Dempster et al. (1977), who also presented its main properties. In short, the EM algorithm aims at finding maximum-likelihood estimates for settings where this appears to be difficult if not impossible. The trick

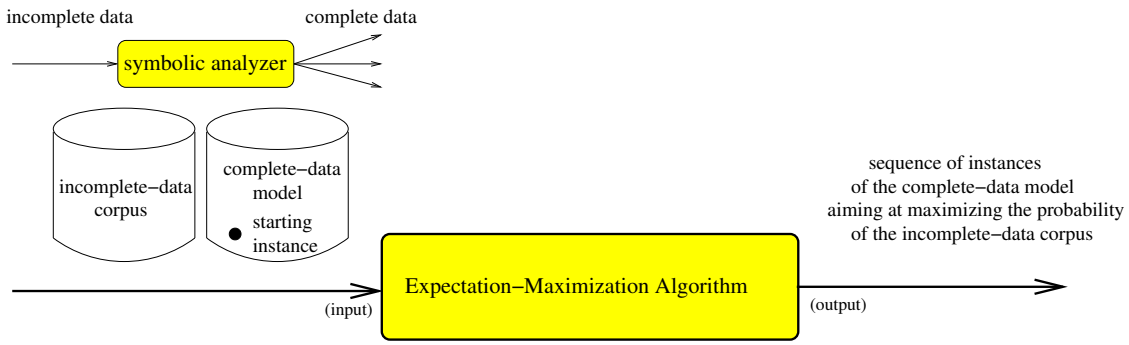

Figure 4: Input and output of the EM algorithm.

of the EM algorithm is to map the *given data* to *complete data* on which it is well-known how to perform maximum-likelihood estimation. Typically, the EM algorithm is applied in the following setting:

- Direct maximum-likelihood estimation of the given probability model on the given corpus is not feasible. For example, if the likelihood function is too complex (e.g. it is a product of sums).
- There is an obvious (but one-to-many) mapping to complete data, on which maximum-likelihood estimation can be easily done. The prototypical example is indeed that maximum-likelihood estimation on the complete data is already a solved problem.

Both relative-frequency and maximum-likelihood estimation are common estimation methods with a two-fold input, a corpus and a probability model[5] such that the instances of the model might have generated the corpus. The output of both estimation methods is simply an instance of the probability model, ideally, the unknown distribution that generated the corpus. In contrast to this setting, in which we are almost *completely informed* (the only thing that is not known to us is the unknown distribution that generated the corpus), the expectation-maximization algorithm is designed to estimate an instance of the probability model for settings, in which we are *incompletely informed*.

To be more specific, instead of a **complete-data corpus**, the input of the expectation-maximization algorithm is an **incomplete-data corpus** together with a so-called **symbolic analyzer**. A symbolic analyzer is a device assigning to each **incomplete-data type** a **set of analyzes**, each analysis being a **complete-data type**. As a result, the missing complete-data corpus can be partly compensated by the expectation-maximization algorithm: The application of the the symbolic analyzer to the incomplete-data corpus leads to an *ambiguous complete-data corpus*. The ambiguity arises as a consequence of the inherent **analytical ambiguity** of the symbolic analyzer: the analyzer can replace each token of the incomplete-data corpus by a set of complete-data types – the set of its analyzes – but clearly, the symbolic analyzer is not able to resolve the analytical ambiguity.

The expectation-maximization algorithm performs a sequence of runs over the resulting ambiguous complete-data corpus. Each of these runs consists of an **expectation step** followed by a **maximization step**. In the **E step**, the expectation-maximization algorithm combines the symbolic analyzer with an instance of the probability model. The results of

[5]We associate the relative-frequency estimate with the unrestricted probability model

this combination is a **statistical analyzer** which is able to **resolve the analytical ambiguity** introduced by the symbolic analyzer. In the **M step**, the expectation-maximization algorithm calculates an ordinary maximum-likelihood estimate on the resolved complete-data corpus.

In general, however, a sequence of such runs is necessary. The reason is that we never know which instance of the given probability model leads to a good statistical analyzer, and thus, which instance of the probability model shall be used in the E-step. The expectation-maximization algorithm provides a simple but somehow surprising solution to this serious problem. At the beginning, a randomly generated **starting instance** of the given probability model is used for the first E-step. In further iterations, the estimate of the M-step is used for the next E-step. Figure 4 displays the input and the output of the EM algorithm. The procedure of the EM algorithm is displayed in Figure 5.

Symbolic and Statistical Analyzers

Definition 10 *Let $\mathcal{X}$ and $\mathcal{Y}$ be non-empty and countable sets. A function*

$$\mathcal{A}\colon \mathcal{Y} \to 2^{\mathcal{X}}$$

*is called a **symbolic analyzer** if the (possibly empty) **sets of analyzes** $\mathcal{A}(y) \subseteq \mathcal{X}$ are pair-wise disjoint, and the union of all sets of analyzes $\mathcal{A}(y)$ is complete*

$$\mathcal{X} = \sum_{y \in \mathcal{Y}} \mathcal{A}(y)$$

*In this case, $\mathcal{Y}$ is called the set of **incomplete-data types**, whereas $\mathcal{X}$ is called the set of **complete-data types**. So, in other words, the analyzes $\mathcal{A}(y)$ of the incomplete-data types y form a partition of the complete-data $\mathcal{X}$. Therefore, for each $x \in \mathcal{X}$ exists a unique $y \in \mathcal{Y}$, the so-called **yield** of x, such that x is an analysis of y*

$$y = \mathit{yield}(x) \quad \textit{if and only if} \quad x \in \mathcal{A}(y)$$

For example, if working in a formal-grammar framework, the *grammatical sentences* can be interpreted as the incomplete-data types, whereas the *grammatical analyzes of the sentences* are the complete-data types. So, in terms of Definition 10, a so-called *parser* – a device assigning a set of grammatical analyzes to a given sentence – is clearly a symbolic analyzer: The most important thing to check is that the parser does not assign a given grammatical analysis to two different sentences – which is pretty obvious, if the *sentence words* are part of the grammatical analyzes.

Definition 11 *A pair $<\mathcal{A},\ p>$ consisting of a symbolic analyzer $\mathcal{A}$ and a probability distribution p on the complete-data types $\mathcal{X}$ is called a **statistical analyzer**. We use a statistical analyzer to **induce probabilities** for the incomplete-data types $y \in \mathcal{Y}$*

$$p(y) := \sum_{x \in \mathcal{A}(y)} p(x)$$

*Even more important, we use a statistical analyzer to **resolve the analytical ambiguity** of an incomplete-data type $y \in \mathcal{Y}$ by looking at the **conditional probabilities of the analyzes** $x \in \mathcal{A}(y)$*

$$p(x|y) := \frac{p(x)}{p(y)} \qquad \textit{where } y = \mathit{yield}(x)$$

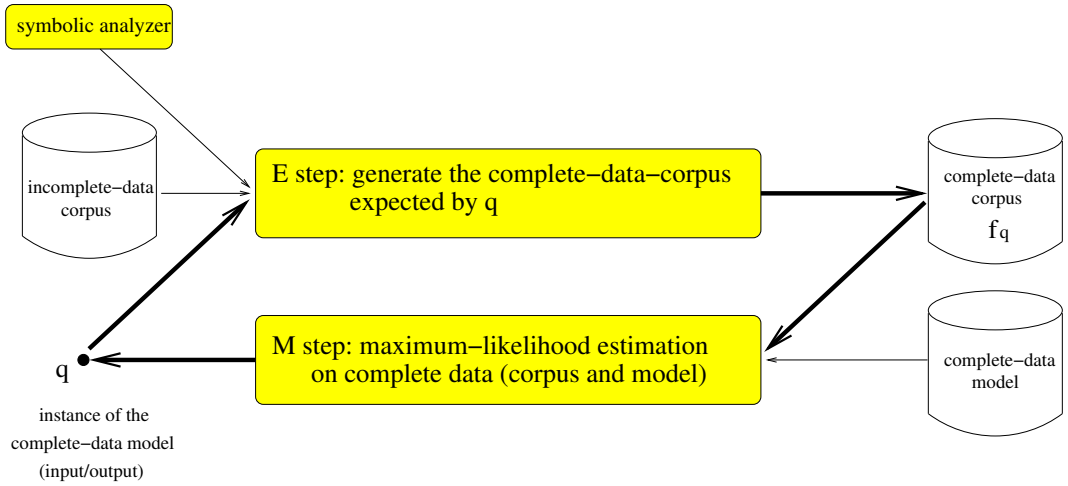

Figure 5: Procedure of the EM algorithm. An incomplete-data corpus, a symbolic analyzer (a device assigning to each incomplete-data type a set of complete-data types), and a complete-data model are given. In the E step, the EM algorithm combines the symbolic analyzer with an instance q of the probability model. The results of this combination is a statistical analyzer that is able to resolve the ambiguity of the given incomplete data. In fact, the statistical analyzer is used to generate an expected complete-data corpus f_q. In the M step, the EM algorithm calculates an ordinary maximum-likelihood estimate of the complete-data model on the complete-data corpus generated in the E step. In further iterations, the estimates of the M-steps are used in the subsequent E-steps. The output of the EM algorithm are the estimates that are produced in the M steps.

It is easy to check that the statistical analyzer induces a proper probability distribution on the set $\mathcal{Y}$ of incomplete-data types

$$\sum_{y \in \mathcal{Y}} p(y) = \sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{A}(y)} p(x) = \sum_{x \in \mathcal{X}} p(x) = 1$$

Moreover, the statistical analyzer induces also proper conditional probability distributions on the sets of analyzes $\mathcal{A}(y)$

$$\sum_{x \in \mathcal{A}(y)} p(x|y) = \sum_{x \in \mathcal{A}(y)} \frac{p(x)}{p(y)} = \frac{\sum_{x \in \mathcal{A}(y)} p(x)}{p(y)} = \frac{p(y)}{p(y)} = 1$$

Of course, by defining $p(x|y) = 0$ for $y \neq \text{yield}(x)$, $p(.|y)$ is even a probability distribution on the full set $\mathcal{X}$ of analyzes.

Input, Procedure, and Output of the EM Algorithm

Definition 12 *The input of the expectation-maximization (EM) algorithm is*

(i) a ***symbolic analyzer****, i.e., a function $\mathcal{A}$ which assigns a* ***set of analyzes*** *$\mathcal{A}(y) \subseteq \mathcal{X}$ to each* ***incomplete-data type*** *$y \in \mathcal{Y}$, such that all sets of analyzes form a partition of the set $\mathcal{X}$ of* ***complete-data types***

$$\mathcal{X} = \sum_{y \in \mathcal{Y}} \mathcal{A}(y)$$

(ii) a non-empty and finite ***incomplete-data corpus****, i.e., a frequency distribution f on the set of incomplete-data types*

$$f\colon \mathcal{Y} \to \mathcal{R} \quad \textit{such that} \quad f(y) \geq 0 \textit{ for all } y \in \mathcal{Y} \quad \textit{and} \quad 0 < |f| < \infty$$

(iii) a ***complete-data model*** *$\mathcal{M} \subseteq \mathcal{M}(\mathcal{X})$, i.e., each instance $p \in \mathcal{M}$ is a probability distribution on the set of complete-data types*

$$p\colon \mathcal{X} \to [0,1] \quad \textit{and} \quad \sum_{x \in \mathcal{X}} p(x) = 1$$

(*) ***implicit input:*** *an* ***incomplete-data model*** *$\mathcal{M} \subseteq \mathcal{M}(\mathcal{Y})$ induced by the symbolic analyzer and the complete-data model. To see this, recall Definition 11. Together with a given instance of the complete-data model, the symbolic analyzer constitutes a statistical analyzer which, in turn, induces the following instance of the incomplete-data model*

$$p\colon \mathcal{Y} \to [0,1] \quad \textit{and} \quad p(y) = \sum_{x \in \mathcal{A}(y)} p(x)$$

(Note: For both complete and incomplete data, the same notation symbols $\mathcal{M}$ and p are used. The sloppy notation, however, is justified, because the incomplete-data model is a marginal of the complete-data model.)

(iv) a (randomly generated) ***starting instance*** *p_0 of the complete-data model $\mathcal{M}$.*
(Note: If permitted by $\mathcal{M}$, then p_0 should not assign to any $x \in \mathcal{X}$ a probability of zero.)

Definition 13 *The procedure of the EM algorithm is*

(1) *for each $i = 1,\ 2,\ 3,\ \ldots$ do*
(2) $q := p_{i-1}$
(3) ***E-step:*** *compute the* ***complete-data corpus*** *$f_q\colon \mathcal{X} \to \mathcal{R}$* ***expected by*** *$q$*

$$f_q(x) \ := f(y) \cdot q(x|y) \qquad \textit{where } y = \textit{yield}(x)$$

(4) ***M-step:*** *compute a maximum-likelihood estimate $\hat{p}$ of $\mathcal{M}$ on f_q*

$$L(f_q; \hat{p}) = \max_{p \in \mathcal{M}} L(f_q, p)$$

(Implicit pre-condition of the EM algorithm: it exists!)

(5) $p_i := \hat{p}$
(6) *end // for each i*
(7) *print $p_0, p_1, p_2, p_3, \ldots$*

In line (3) of the EM procedure, a complete-data corpus $f_q(x)$ has to be generated on the basis of the incomplete-data corpus $f(y)$ and the conditional probabilities $q(x|y)$ of the analyzes of y (conditional probabilities are introduced in Definition 11). In fact, this generation procedure is conceptually very easy: according to the conditional probabilities $q(x|y)$, the frequency $f(y)$ has to be distributed among the complete-data types $x \in \mathcal{A}(y)$. Figure 6 displays the procedure. Moreover, there exists a simple reversed procedure (summation of all frequencies

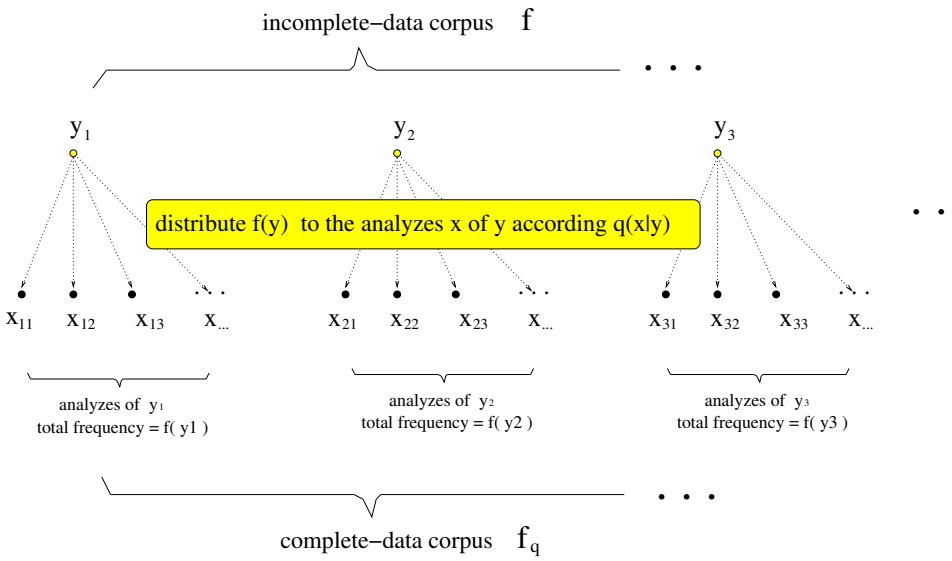

Figure 6: The E step of the EM algorithm. A complete-data corpus $f_q(x)$ is generated on the basis of the incomplete-data corpus $f(y)$ and the conditional probabilities $q(x|y)$ of the analyzes of y. The frequency $f(y)$ is distributed among the complete-data types $x \in \mathcal{A}(y)$ according to the conditional probabilities $q(x|y)$. A simple reversed procedure guarantees that the original incomplete-data corpus $f(y)$ can be recovered from the generated corpus $f_q(x)$: Sum up all frequencies $f_q(x)$ with $x \in \mathcal{A}(y)$. So the size of both corpora is the same $|f_q| = |f|$. *Memory hook*: f_q is the $q$omplete data corpus.

$f_q(x)$ with $x \in \mathcal{A}(y)$) which guarantees that the original corpus $f(y)$ can be recovered from the generated corpus $f_q(x)$. Finally, the size of both corpora is the same

$$|f_q| = |f|$$

In line (4) of the EM procedure, it is stated that a maximum-likelihood estimate $\hat{p}$ of the complete-data model has to be computed on the complete-data corpus f_q expected by q. Recall for this purpose that the probability of f_q allocated by an instance $p \in \mathcal{M}$ is defined as

$$L(f_q; p) = \prod_{x \in \mathcal{X}} p(x)^{f_q(x)}$$

In contrast, the probability of the incomplete-data corpus f allocated by an instance p of the incomplete-data model is much more complex. Using Definition 12.*, we get an expression involving a product of sums

$$L(f; p) = \prod_{y \in \mathcal{Y}} \left(\sum_{x \in \mathcal{A}(y)} p(x) \right)^{f(y)}$$

Nevertheless, the following theorem reveals that the EM algorithm aims at finding an instance of the incomplete-data model which possibly maximizes the probability of the incomplete-data corpus.

Theorem 5 *The output of the EM algorithm is: A sequence of instances of the complete-data model $\mathcal{M}$, the so-called* ***EM re-estimates****,*

$$p_0,\ p_1,\ p_2,\ p_3,\ \ldots$$

such that the sequence of probabilities allocated to the incomplete-data corpus is monotonic increasing

$$L(f;p_0) \leq L(f;p_1) \leq L(f;p_2) \leq L(f;p_3) \leq \ldots$$

It is common wisdom that the sequence of EM re-estimates will converge to a (local) maximum-likelihood estimate of the incomplete-data model on the incomplete-data corpus. As proven by Wu (1983), however, the EM algorithm will do this only in specific circumstances. Of course, it is guaranteed that the sequence of corpus probabilities (allocated by the EM re-estimates) must converge. However, we are more interested in the behavior of the EM re-estimates itself. Now, intuitively, the EM algorithm might get stuck in a saddle point or even a local minimum of the corpus-probability function, whereas the associated model instances are hopping uncontrolled around (for example, on a circle-like path in the "space" of all model instances).

Proof See theorems 6 and 7.

The Generalized Expectation-Maximization Algorithm

The EM algorithm performs a sequence of maximum-likelihood estimations on complete data, resulting in good re-estimates on incomplete-data ("good" in the sense of Theorem 5). The following theorem, however, reveals that the EM algorithm might overdo it somehow, since there exist alternative M-steps which can be easier performed, and which result in re-estimates having the same property as the EM re-estimates.

Definition 14 *A generalized expectation-maximization (GEM) algorithm has exactly the same input as the EM-algorithm, but an easier M-step is performed in its procedure:*

(4) ***M-step (GEM):*** *compute an instance $\hat{p}$ of the complete-data model $\mathcal{M}$ such that*

$$L(f_q;\hat{p}) \geq L(f_q;q)$$

Theorem 6 *The output of a GEM algorithm is: A sequence of instances of the complete-data model $\mathcal{M}$, the so-called* ***GEM re-estimates****, such that the sequence of probabilities allocated to the incomplete-data corpus is monotonic increasing.*

Proof Various proofs have been given in the literature. The first one was presented by Dempster et al. (1977). For other variants of the EM algorithm, the book of McLachlan and Krishnan (1997) is a good source. Here, we present something along the lines of the original proof. Clearly, a proof of the theorem requires somehow that we are able to express *the probability of the given incomplete-data corpus f* in terms of the *the probabilities of complete-data corpora f_q* which are involved in the M-steps of the GEM algorithm (where both types of corpora are allocated a probability by the same instance p of the model $\mathcal{M}$). A certain entity, which we would like to call the **expected cross-entropy on the analyzes**, plays a major role for solving this task. To be specific, the expected cross-entropy on the analyzes is

defined as the expectation of certain cross-entropy values $H_{\mathcal{A}(y)}(q,p)$ which are calculated on the different sets $\mathcal{A}(y)$ of analyzes. Then, of course, the "expectation" is calculated on the basis of the relative-frequency estimate $\tilde{p}$ of the given incomplete-data corpus

$$H_{\mathcal{A}}(q;p) = \sum_{y \in \mathcal{Y}} \tilde{p}(y) \cdot H_{\mathcal{A}(y)}(q;p)$$

Now, for two instances q and p of the complete-data model, their conditional probabilities $q(x|y)$ and $p(x|y)$ form proper probability distributions on the set $\mathcal{A}(y)$ of analyzes of y (see Definition 11). So, the cross-entropy $H_{\mathcal{A}(y)}(q;p)$ on the set $\mathcal{A}(y)$ is simply given by

$$H_{\mathcal{A}(y)}(q;p) = -\sum_{x \in \mathcal{A}(y)} q(x|y) \log p(x|y)$$

Recalling the central task of this proof, a bunch of relatively straight-forward calculations leads to the following interesting equation[6]

$$L(f;p) = \left(2^{H_{\mathcal{A}}(q;p)}\right)^{|f|} \cdot L(f_q;p)$$

Using this equation, we can state that

$$\frac{L(f;p)}{L(f;q)} = \left(2^{H_{\mathcal{A}}(q;p) - H_{\mathcal{A}}(q,q)}\right)^{|f|} \cdot \frac{L(f_q;p)}{L(f_q;q)}$$

In what follows, we will show that, after each M-step of a GEM algorithm (i.e. for p being a GEM re-estimate $\hat{p}$), both of the factors on the right-hand side of this equation are not less than one. First, an iterated application of the information inequality of information theory (see Theorem 3) yields

$$\begin{aligned} H_{\mathcal{A}}(q;p) - H_{\mathcal{A}}(q,q) &= \sum_{y \in \mathcal{Y}} \tilde{p}(y) \cdot \left(H_{\mathcal{A}(y)}(q;p) - H_{\mathcal{A}(y)}(q;q)\right) \\ &= \sum_{y \in \mathcal{Y}} \tilde{p}(y) \cdot D_{\mathcal{A}(y)}(q||p) \\ &\geq 0 \end{aligned}$$

So, the first factor is never (i.e. for no model instance p) less than one

$$\left(2^{H_{\mathcal{A}}(q;p) - H_{\mathcal{A}}(q,q)}\right)^{|f|} \geq 1$$

[6] It is easier to show that

$$H(\tilde{p};p) = H(\tilde{p}_q;p) - H_{\mathcal{A}}(q;p).$$

Here, $\tilde{p}$ is the relative-frequency estimate on the incomplete-data corpus f, whereas $\tilde{p}_q$ is the relative-frequency estimate on the complete-data corpus f_q. However, by defining an "average perplexity of the analyzes", $\mathrm{perp}_{\mathcal{A}}(q;p) := 2^{H_{\mathcal{A}}(q;p)}$ (see also Footnote 3), the true spirit of the equation can be revealed:

$$L(f_q;p) = L(f;p) \cdot \left(\frac{1}{\mathrm{perp}_{\mathcal{A}}(q;p)}\right)^{|f|}$$

This equation states that the probability of a complete-data corpus (generated by a statistical analyzer) is the product of the probability of the given incomplete-data corpus and $|f|$-times the average probability of the different corpora of analyzes (as generated for each of the $|f|$ tokens of the incomplete-data corpus).

Second, by definition of the M-step of a GEM algorithm, the second factor is also not less than one

$$\frac{L(f_q;\hat{p})}{L(f_q;q)} \geq 1$$

So, it follows

$$\frac{L(f;\hat{p})}{L(f;q)} \geq 1$$

yielding that the probability of the incomplete-data corpus allocated by the GEM re-estimate $\hat{p}$ is not less than the probability of the incomplete-data corpus allocated by the model instance q (which is either the starting instance p_0 of the GEM algorithm or the previously calculated GEM re-estimate)

$$L(f;\hat{p}) \geq L(f;q)$$

Theorem 7 *An EM algorithm is a GEM algorithm.*

Proof In the M-step of an EM algorithm, a model instance $\hat{p}$ is selected such that

$$L(f_q;\hat{p}) = \max_{p \in \mathcal{M}} L(f_q, p)$$

So, especially

$$L(f_q;\hat{p}) \geq L(f_q, q)$$

and the requirements of the M-step of a GEM algorithm are met.

4 Rolling Two Dice

Example 2 *We shall now consider an experiment in which two loaded dice are rolled, and we shall compute the relative-frequency estimate on a corpus of outcomes.*

If we assume that the two dice are distinguishable, each outcome can be represented as a pair of numbers (x_1, x_2), where x_1 is the number that appears on the first die and x_2 is the number that appears on the second die. So, for this experiment, an appropriate set $\mathcal{X}$ of types comprises the following 36 outcomes:

(x_1, x_2)	$x_2 = 1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 4$	$x_2 = 5$	$x_2 = 6$
$x_1 = 1$	$(1,1)$	$(1,2)$	$(1,3)$	$(1,4)$	$(1,5)$	$(1,6)$
$x_1 = 2$	$(2,1)$	$(2,2)$	$(2,3)$	$(2,4)$	$(2,5)$	$(2,6)$
$x_1 = 3$	$(3,1)$	$(3,2)$	$(3,3)$	$(3,4)$	$(3,5)$	$(3,6)$
$x_1 = 4$	$(4,1)$	$(4,2)$	$(4,3)$	$(4,4)$	$(4,5)$	$(4,6)$
$x_1 = 5$	$(5,1)$	$(5,2)$	$(5,3)$	$(5,4)$	$(5,5)$	$(5,6)$
$x_1 = 6$	$(6,1)$	$(6,2)$	$(6,3)$	$(6,4)$	$(6,5)$	$(6,6)$

If we throw the two dice a 100 000 times, then the following occurrence frequencies might arise

$f(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 4$	$x_2 = 5$	$x_2 = 6$
$x_1 = 1$	3790	3773	1520	1498	2233	2298
$x_1 = 2$	3735	3794	1497	1462	2269	2184
$x_1 = 3$	4903	4956	1969	2035	2883	3010
$x_1 = 4$	2495	2519	1026	1049	1487	1451
$x_1 = 5$	3820	3735	1517	1498	2276	2191
$x_1 = 6$	6369	6290	2600	2510	3685	3673

The size of this corpus is $|f| = 100\ 000$. So, the relative-frequency estimate $\tilde{p}$ on f can be easily computed (see Definition 4)

$\tilde{p}(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 4$	$x_2 = 5$	$x_2 = 6$
$x_1 = 1$	0.03790	0.03773	0.01520	0.01498	0.02233	0.02298
$x_1 = 2$	0.03735	0.03794	0.01497	0.01462	0.02269	0.02184
$x_1 = 3$	0.04903	0.04956	0.01969	0.02035	0.02883	0.03010
$x_1 = 4$	0.02495	0.02519	0.01026	0.01049	0.01487	0.01451
$x_1 = 5$	0.03820	0.03735	0.01517	0.01498	0.02276	0.02191
$x_1 = 6$	0.06369	0.06290	0.02600	0.02510	0.03685	0.03673

Example 3 *We shall consider again Experiment 2 in which two loaded dice are rolled, but we shall now compute the relative-frequency estimate on the corpus of outcomes of the first die, as well as on the corpus of outcomes of the second die.*

If we look at the same corpus as in Example 2, then the corpus f_1 of outcomes of the first die can be calculated as $f_1(x_1) = \sum_{x_2} f(x_1, x_2)$. An analog summation yields the corpus of outcomes of the second die, $f_2(x_2) = \sum_{x_1} f(x_1, x_2)$. Obviously, the sizes of all corpora are identical $|f_1| = |f_2| = |f| = 100\ 000$. So, the relative-frequency estimates $\tilde{p_1}$ on f_1 and $\tilde{p_2}$ on f_2 are calculated as follows

$f_1(x_1)$	x_1
15112	1
14941	2
19756	3
10027	4
15037	5
25127	6

$\tilde{p}_1(x_1)$	x_1
0.15112	1
0.14941	2
0.19756	3
0.10027	4
0.15037	5
0.25127	6

$f_2(x_2)$	x_2
25112	1
25067	2
10129	3
10052	4
14833	5
14807	6

$\tilde{p}_2(x_2)$	x_2
0.25112	1
0.25067	2
0.10129	3
0.10052	4
0.14833	5
0.14807	6

Example 4 *We shall consider again Experiment 2 in which two loaded dice are rolled, but we shall now compute a maximum-likelihood estimate of the probability model which assumes that the numbers appearing on the first and second die are statistically independent.*

First, recall the definition of statistical independence (see e.g. Duda et al. (2001), page 613).

Definition 15 *The variables x_1 and x_2 are said to be **statistically independent** given a joint probability distribution p on $\mathcal{X}$ if and only if*

$$p(x_1, x_2) = p_1(x_1) \cdot p_2(x_2)$$

*where p_1 and p_2 are the **marginal distributions** for x_1 and x_2*

$$p_1(x_1) = \sum_{x_2} p(x_1, x_2)$$
$$p_2(x_2) = \sum_{x_1} p(x_1, x_2)$$

So, let $\mathcal{M}_{1/2}$ be the probability model which assumes that the numbers appearing on the first and second die are statistically independent

$$\mathcal{M}_{1/2} = \{p \in \mathcal{M}(\mathcal{X}) \mid x_1 \text{ and } x_2 \text{ are statistically independent given } p\}$$

In Example 2, we have calculated the relative-frequency estimator $\tilde{p}$. Theorem 1 states that $\tilde{p}$ is the unique maximum-likelihood estimate of the unrestricted model $\mathcal{M}(\mathcal{X})$. Thus, $\tilde{p}$ is also a candidate for a maximum-likelihood estimate of $\mathcal{M}_{1/2}$. Unfortunately, however, x_1 and x_2 are **not** statistically independent given $\tilde{p}$ (see e.g. $\tilde{p}(1,1) = 0.03790$ and $\tilde{p}_1(1) \cdot \tilde{p}_2(1) = 0.0379\mathbf{493}$). This has two consequences for the experiment in which two (loaded) dice are rolled:

- the probability model, which assumes that the numbers appearing on the first and second die are statistically independent, is a **restricted model** (see Definition 5), and
- **the relative-frequency estimate is in general not a maximum-likelihood estimate of the standard probability model** assuming that the numbers appearing on the first and second die are statistically independent.

Therefore, we are now following Definition 6 to compute the maximum-likelihood estimate of $\mathcal{M}_{1/2}$. Using the independence property, the probability of the corpus f allocated by an instance p of the model $\mathcal{M}_{1/2}$ can be calculated as

$$L(f;p) = \left(\prod_{x_1=1,\ldots,6} p_1(x_1)^{f_1(x_1)} \right) \cdot \left(\prod_{x_2=1,\ldots,6} p_2(x_2)^{f_2(x_2)} \right) = L(f_1;p_1) \cdot L(f_2;p_2)$$

Definition 6 states that the maximum-likelihood estimate $\hat{p}$ of $\mathcal{M}_{1/2}$ on f must maximize $L(f;p)$. A product, however, is maximized, if and only if its factors are simultaneously maximized. Theorem 1 states that the corpus probabilities $L(f_i;p_i)$ are maximized by the relative-frequency estimators $\tilde{p}_i$. Therefore, the product of the relative-frequency estimators $\tilde{p_1}$ and $\tilde{p_2}$ (on f_1 and f_2 respectively) might be a candidate for the maximum-likelihood estimate $\hat{p}$ we are looking for

$$\hat{p}(x_1,x_2) = \tilde{p_1}(x_1) \cdot \tilde{p_2}(x_2)$$

Now, note that the marginal distributions of $\hat{p}$ are identical with the relative-frequency estimators on f_1 and f_2. For example, $\hat{p}$'s marginal distribution for x_1 is calculated as

$$\hat{p}_1(x_1) = \sum_{x_2} \hat{p}(x_1,x_2) = \sum_{x_2} \tilde{p}_1(x_1) \cdot \tilde{p}_2(x_2) = \tilde{p_1}(x_1) \cdot \sum_{x_2} \tilde{p_2}(x_2) = \tilde{p}_1(x_1) \cdot 1 = \tilde{p}_1(x_1)$$

A similar calculation yields $\hat{p}_2(x_2) = \tilde{p}_2(x_2)$. Both equations state that x_1 and x_2 are indeed statistically independent given $\hat{p}$

$$\hat{p}(x_1,x_2) = \hat{p_1}(x_1) \cdot \hat{p_2}(x_2)$$

So, finally, it is guaranteed that $\hat{p}$ is an instance of the probability model $\mathcal{M}_{1/2}$ as required for a maximum-likelihood estimate of $\mathcal{M}_{1/2}$. *Note: $\hat{p}$ is even an unique maximum-likelihood estimate since the relative-frequency estimates $\tilde{p}_i$ are unique maximum-likelihood estimates (see Theorem 1).* The relative-frequency estimates $\tilde{p}_1$ and $\tilde{p}_2$ have already been calculated in Example 3. So, $\hat{p}$ is calculated as follows

$\hat{p}(x_1,x_2)$	$x_2=1$	$x_2=2$	$x_2=3$	$x_2=4$	$x_2=5$	$x_2=6$
$x_1=1$	0.0379493	0.0378813	0.0153069	0.0151906	0.0224156	0.0223763
$x_1=2$	0.0375198	0.0374526	0.0151337	0.0150187	0.022162	0.0221231
$x_1=3$	0.0496113	0.0495224	0.0200109	0.0198587	0.0293041	0.0292527
$x_1=4$	0.0251798	0.0251347	0.0101563	0.0100791	0.014873	0.014847
$x_1=5$	0.0377609	0.0376932	0.015231	0.0151152	0.0223044	0.0222653
$x_1=6$	0.0630989	0.0629859	0.0254511	0.0252577	0.0372709	0.0372055

Example 5 *We shall consider again Experiment 2 in which two loaded dice are rolled. Now, however, we shall assume that we are incompletely informed: the corpus of outcomes (which is given to us) consists only of the sums of the numbers which appear on the first and second die. Nevertheless, we shall compute an estimate for a probability model on the complete-data* $(x_1, x_2) \in \mathcal{X}$.

If we assume that the corpus which is given to us was calculated on the basis of the corpus given in Example 2, then the occurrence frequency of a sum y can be calculated as $f(y) = \sum_{x_1+x_2=y} f(x_1, x_2)$. These numbers are displayed in the following table

$f(y)$	y
3790	2
7508	3
10217	4
10446	5
12003	6
17732	7
13923	8
8595	9
6237	10
5876	11
3673	12

For example,

$$f(4) \quad = \quad f(1,3) + f(2,2) + f(3,1) = 1520 + 3794 + 4903 = \underline{\underline{10217}}$$

The problem is now, whether this corpus of sums can be used to calculate a good estimate on the outcomes (x_1, x_2) itself. *Hint: Examples 2 and 4 have shown that a unique relative-frequency estimate* $\tilde{p}(x_1, x_2)$ *and a unique maximum-likelihood estimate* $\hat{p}(x_1, x_2)$ *can be calculated on the basis of the corpus* $f(x_1, x_2)$. *However, right now, this corpus is not available!* Putting the example in the framework of the EM algorithm (see Definition 12), the set of **incomplete-data types** is

$$\mathcal{Y} = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$$

whereas the set of **complete-data types** is $\mathcal{X}$. We also know the **set of analyzes for each incomplete-data type** $y \in \mathcal{Y}$

$$\mathcal{A}(y) = \{(x_1, x_2) \in \mathcal{X} \mid x_1 + x_2 = y\}$$

As in Example 4, we are especially interested in an estimate of the (slightly restricted) **complete-data model** $\mathcal{M}_{1/2}$ which assumes that the numbers appearing on the first and second die are statistically independent. So, for this case, a randomly generated **starting instance** $p_0(x_1, x_2)$ of the complete-data model is simply the product of a randomly generated probability distribution $p_{01}(x_1)$ for the numbers appearing on the first dice, and a randomly generated probability distribution $p_{02}(x_2)$ for the numbers appearing on the second dice

$$p_0(x_1, x_2) = p_{01}(x_1) \cdot p_{02}(x_2)$$

The following tables display some randomly generated numbers for p_{01} and p_{02}

$p_{01}(x_1)$	x_1
0.18	1
0.19	2
0.16	3
0.13	4
0.17	5
0.17	6

$p_{02}(x_2)$	x_2
0.22	1
0.23	2
0.13	3
0.16	4
0.14	5
0.12	6

Using the random numbers for $p_{01}(x_1)$ and $p_{02}(x_2)$, a starting instance p_0 of the complete-data model $\mathcal{M}_{1/2}$ is calculated as follows

$p_0(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 4$	$x_2 = 5$	$x_2 = 6$
$x_1 = 1$	0.0396	0.0414	0.0234	0.0288	0.0252	0.0216
$x_1 = 2$	0.0418	0.0437	0.0247	0.0304	0.0266	0.0228
$x_1 = 3$	0.0352	0.0368	0.0208	0.0256	0.0224	0.0192
$x_1 = 4$	0.0286	0.0299	0.0169	0.0208	0.0182	0.0156
$x_1 = 5$	0.0374	0.0391	0.0221	0.0272	0.0238	0.0204
$x_1 = 6$	0.0374	0.0391	0.0221	0.0272	0.0238	0.0204

For example,

$$\begin{aligned} p_0(1,3) &= p_{01}(1) \cdot p_{02}(3) = 0.18 \cdot 0.13 = \underline{\underline{0.0234}} \\ p_0(2,2) &= p_{01}(2) \cdot p_{02}(2) = 0.19 \cdot 0.23 = \underline{\underline{0.0437}} \\ p_0(3,1) &= p_{01}(3) \cdot p_{02}(1) = 0.16 \cdot 0.22 = \underline{\underline{0.0352}} \end{aligned}$$

So, we are ready to start the procedure of the EM algorithm.

First EM iteration. In the **E-step**, we shall compute the **complete-data corpus** f_q expected by $q := p_0$. For this purpose, the probability of each incomplete-data type given the starting instance p_0 of the complete-data model has to be computed (see Definition 12.*)

$$p_0(y) = \sum_{x_1 + x_2 = y} p_0(x_1, x_2)$$

The above displayed numbers for $p_0(x_1, x_2)$ yield the following instance of the incomplete-data model

$p_0(y)$	y
0.0396	2
0.0832	3
0.1023	4
0.1189	5
0.1437	6
0.1672	7
0.1272	8
0.0867	9
0.0666	10
0.0442	11
0.0204	12

For example,

$$p_0(4) \quad = \quad p_0(1,3) + p_0(2,2) + p_0(3,1) = 0.0234 + 0.0437 + 0.0352 = \underline{\underline{0.1023}}$$

So, the complete-data corpus expected by $q := p_0$ is calculated as follows (see line (3) of the EM procedure given in Definition 13)

$f_q(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 4$	$x_2 = 5$	$x_2 = 6$
$x_1 = 1$	3790	3735.95	2337.03	2530.23	2104.91	2290.74
$x_1 = 2$	3772.05	4364.45	2170.03	2539.26	2821	2495.63
$x_1 = 3$	3515.53	3233.08	1737.39	2714.95	2451.85	1903.39
$x_1 = 4$	2512.66	2497.49	1792.29	2276.72	1804.26	1460.92
$x_1 = 5$	3123.95	4146.66	2419.01	2696.47	2228.84	2712
$x_1 = 6$	3966.37	4279.79	2190.88	2547.24	3164	3673

For example,

$$\begin{aligned}
f_q(1,3) &= f(4) \cdot \frac{p_0(1,3)}{p_0(4)} = 10217 \cdot \frac{0.0234}{0.1023} = \underline{\underline{2337.03}} \\
f_q(2,2) &= f(4) \cdot \frac{p_0(2,2)}{p_0(4)} = 10217 \cdot \frac{0.0437}{0.1023} = \underline{\underline{4364.45}} \\
f_q(3,1) &= f(4) \cdot \frac{p_0(3,1)}{p_0(4)} = 10217 \cdot \frac{0.0352}{0.1023} = \underline{\underline{3515.53}}
\end{aligned}$$

(The frequency $f(4)$ of the dice sum 4 is distributed to its analyzes (1,3), (2,2), and (3,1), simply by correlating the current probabilities $q = p_0$ of the analyses...)

In the **M-step**, we shall compute a maximum-likelihood estimate $p_1 := \hat{p}$ of the complete-data model $\mathcal{M}_{1/2}$ on the complete-data corpus f_q. This can be done along the lines of Examples 3 and 4. *Note: This is more or less the trick of the EM-algorithm! If it appears to be difficult to compute a maximum-likelihood estimate of an incomplete-data model then the EM algorithm might solve your problem. It performs a sequence of maximum-likelihood estimations on complete-data corpora. These corpora contain in general more complex data, but nevertheless, it might be well-known, how one has to deal with this data!* In detail: On the basis of the complete-data corpus f_q (where currently $q = p_0$), the corpus f_{q1} of outcomes of the first die is calculated as $f_{q1}(x_1) = \sum_{x_2} f_q(x_1, x_2)$, whereas the corpus of outcomes of the second die is calculated as $f_{q2}(x_2) = \sum_{x_1} f_q(x_1, x_2)$. The following tables display them:

$f_{q1}(x_1)$	x_1
16788.86	1
18162.42	2
15556.19	3
12344.34	4
17326.93	5
19821.28	6

$f_{q2}(x_2)$	x_2
20680.56	1
22257.42	2
12646.63	3
15304.87	4
14574.86	5
14535.68	6

For example,

$$f_{q1}(1) \quad = \quad f_q(1,1) + f_q(1,2) + f_q(1,3) + f_q(1,4) + f_q(1,5) + f_q(1,6)$$

$$
\begin{aligned}
&= 3790 + 3735.95 + 2337.03 + 2530.23 + 2104.91 + 2290.74 = \underline{\underline{16788.86}} \\
f_{q2}(1) &= f_q(1,1) + f_q(2,1) + f_q(3,1) + f_q(4,1) + f_q(5,1) + f_q(6,1) \\
&= 3790 + 3772.05 + 3515.53 + 2512.66 + 3123.95 + 3966.37 = \underline{\underline{20680.56}}
\end{aligned}
$$

The sizes of both corpora are still $|f_{q1}| = |f_{q2}| = |f| = 100\ 000$, resulting in the following relative-frequency estimates (p_{11} on f_{q1} respectively p_{12} on f_{q2})

$p_{11}(x_1)$	x_1
0.167889	1
0.181624	2
0.155562	3
0.123443	4
0.173269	5
0.198213	6

$p_{12}(x_2)$	x_2
0.206806	1
0.222574	2
0.126466	3
0.153049	4
0.145749	5
0.145357	6

So, the following instance is the maximum-likelihood estimate of the model $\mathcal{M}_{1/2}$ on f_q

$p_1(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 4$	$x_2 = 5$	$x_2 = 6$
$x_1 = 1$	0.0347204	0.0373677	0.0212322	0.0256952	0.0244696	0.0244038
$x_1 = 2$	0.0375609	0.0404247	0.0229692	0.0277973	0.0264715	0.0264003
$x_1 = 3$	0.0321711	0.034624	0.0196733	0.0238086	0.022673	0.022612
$x_1 = 4$	0.0255287	0.0274752	0.0156113	0.0188928	0.0179917	0.0179433
$x_1 = 5$	0.035833	0.0385651	0.0219126	0.0265186	0.0252538	0.0251858
$x_1 = 6$	0.0409916	0.044117	0.0250672	0.0303363	0.0288893	0.0288116

For example,

$$
\begin{aligned}
p_1(1,1) &= p_{11}(1) \cdot p_{12}(1) = 0.167889 \cdot 0.206806 = \underline{\underline{0.0347204}} \\
p_1(1,2) &= p_{11}(1) \cdot p_{12}(2) = 0.167889 \cdot 0.222574 = \underline{\underline{0.0373677}} \\
p_1(2,1) &= p_{11}(2) \cdot p_{12}(1) = 0.181624 \cdot 0.206806 = \underline{\underline{0.0375609}} \\
p_1(2,2) &= p_{11}(2) \cdot p_{12}(2) = 0.181624 \cdot 0.222574 = \underline{\underline{0.0404247}}
\end{aligned}
$$

So, we are ready for the second EM iteration, where an estimate p_2 is calculated. If we continue in this manner, we will arrive finally at the

1584th EM iteration. The estimate which is calculated here is

$p_{1584,1}(x_1)$	x_1
0.158396	1
0.141282	2
0.204291	3
0.0785532	4
0.172207	5
0.24527	6

$p_{1584,2}(x_2)$	x_2
0.239281	1
0.260559	2
0.104026	3
0.111957	4
0.134419	5
0.149758	6

yielding

$p_{1584}(x_1, x_2)$	$x_2 = 1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 4$	$x_2 = 5$	$x_2 = 6$
$x_1 = 1$	0.0379012	0.0412715	0.0164773	0.0177336	0.0212914	0.0237211
$x_1 = 2$	0.0338061	0.0368123	0.014697	0.0158175	0.018991	0.0211581
$x_1 = 3$	0.048883	0.0532299	0.0212516	0.0228718	0.0274606	0.0305942
$x_1 = 4$	0.0187963	0.0204678	0.00817158	0.00879459	0.0105591	0.011764
$x_1 = 5$	0.0412059	0.0448701	0.017914	0.0192798	0.0231479	0.0257894
$x_1 = 6$	0.0586885	0.0639074	0.0255145	0.0274597	0.032969	0.0367312

In this example, more EM iterations will result in exactly the same re-estimates. So, this is a strong reason to quit the EM procedure. Comparing $p_{1584,1}$ and $p_{1584,2}$ with the results of Example 3 *(Hint: where we have assumed that a complete-data corpus is given to us!)*, we see that the EM algorithm yields pretty similar estimates.

Acknowledgments

Parts of the paper cover parts of the teaching material of two courses at ESSLLI 2003 in Vienna. One of them has been sponsored by the *European Chapter of the Association for Computational Linguistics (EACL)*, and both have been co-lectured by Khalil Sima'an and me. Various improvements of the paper have been suggested by Wietse Balkema, Gabriel Infante-Lopez, Karin Müller, Mark-Jan Nederhof, Breanndán Ó Nualláin, Khalil Sima'an, and Andreas Zollmann.

References

Chi, Z. (1999). Statistical properties of probabilistic context-free grammars. *Computational Linguistics 25*(1).

Cover, T. M. and J. A. Thomas (1991). *Elements of Information Theory*. New York: Wiley.

DeGroot, M. H. (1989). *Probability and statistics* (2 ed.). Addison-Wesley.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the *EM* algorithm. *J. Royal Statist. Soc. 39*(B), 1–38.

Duda, R. O., P. E. Hart, and D. G. Stork (2001). *Pattern Classification — 2nd ed.* New York: Wiley.

Jaynes, E. T. (1957). Information theory and statistical mechanics. *Physical Review 106*, 620–630.

McLachlan, G. J. and T. Krishnan (1997). *The EM Algorithm and Extensions*. New York: Wiley.

Ratnaparkhi, A. (1997). A simple introduction to maximum-entropy models for natural language processing. Technical report, University of Pennsylvania.

Wu, C. F. J. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics 11*(1), 95–103.