ASTES

Advances in Science, Technology and Engineering Systems Journal
Vol. 5, No. 6, 1293-1305 (2020)
www.astesj.com
Special Issue on Multidisciplinary Innovation in Engineering Science & Technology

ASTES Journal
ISSN: 2415-6698

Empirical Probability Distributions with Unknown Number of Components

Marcin Kuropatwiński[*], Leonard Sikorski

Talking2Rabbit Ltd., R & D Department, 80-266, Poland

ARTICLE INFO

Article history:
Received: 28 August, 2020
Accepted: 09 December, 2020
Online: 16 December, 2020

Keywords:
probability mass function
probability density function
multinomial distribution
Laplace estimator
Gaussian mixture model

ABSTRACT

We consider the estimation of empirical probability distributions, both discrete and continuous. We focus on deriving formulas to estimate number of categories for the discrete distribution, when the number of categories is hidden, and the means and methods to estimate the number of components in the Gaussian mixture model representing a probability density function given implicitly in terms of its realizations. To reach the stated goals, we solve certain combinatorial problems for discrete distribution and develop methods to compute the expected Kullback-Leibler divergence for Gaussians. The last mentioned result is needed to develop the theory of continuous distributions. Sample applications and an extensive numerical study are given.

1 Introduction

This paper is an extension of work originally presented in 2019 Signal Processing Symposium [1]. The extension includes:

- measure theoretic account,
- derivation of the results first presented in the paper [1],
- new section devoted entirely to estimation of the number of components in continuous distributions,
- a numerical study of algorithms dealing with estimation of the continuous distributions,
- a specialized algorithm for counting number of monomials in a trace of covariance matrix raised to a power.

Empirical probability distributions are a crucial element of many applications such as machine learning [2], source coding [3], data compression [4], speech recognition [5], speaker recognition [6], image recognition [7], noise reduction [8], bandwidth extension [9], and many others. It is also a scientific discipline in itself, which is studied independently [10], [11]. In this paper, we deal with the means and methods to estimate probability distributions, both discrete and continuous. That being said, we also focus on a particular problem encountered during probability distributions estimation, which is the problem of an unknown number of components. In case of discrete distribution, the number of components is the number of categories, and in case of continuous probability distributions, it is the number of Gaussian components in the Gaussian mixture model (GMM) [2]. We provide estimators and methods to determine the number of components. Toward this, in case of GMMs, we formulate an information-theoretic criterion that allows the selection of an optimal, in some sense, number of components.

To strengthen the ties between discrete and continuous probability distribution, we provide a short account of measure theory and probability spaces. In particular, we show that the difference lies in the type of the underlying measurable space, which is a *standard space* with finite countable alphabets in case of discrete distributions and a *Polish Borel space* (where the *Polish space*, i.e., the complete separable metric space, is the sample space) in the case of continuous distributions. The name *Polish space* originates from the pioneering work by Polish mathematicians on such spaces.

1.1 Measure Theoretic Account

This account is based on the presentation provided by Robert M. Gray in an excellent book [12].

[*] Corresponding Author: Marcin Kuropatwiński, marcin@talking2rabbit.com

https://dx.doi.org/10.25046/aj0506154

1.1.1 Measurable Space

A measurable space is a pair, $(\Omega, \mathcal{B})$ consisting of a sample space Ω with a σ-field $\mathcal{B}$ of subsets of Ω (also called the event space). A σ-field or σ-algebra $\mathcal{B}$ is a collection of subsets of Ω with the following properties:

$$\Omega \in \mathcal{B} \tag{1}$$

$$\text{If } F \in \mathcal{B}, \text{ then } F^c = \{\omega : \omega \notin F\} \in \mathcal{B} \tag{2}$$

$$\text{If } F_i \in \mathcal{B};\ i = 1, 2, \dots, \text{ then } \cup F_i \in \mathcal{B} \tag{3}$$

The type of the measurable space is what differentiates discrete from continuous distributions. More precisely, it is the topology of the sample space. The discrete topology gives rise to discrete probability distributions, and the euclidean topology gives rise to continuous probability distributions.

Let $\mathcal{F} = \{F_i, i = 0, 1, 2, \dots, n-1\}$ be a finite field of sample space Ω, that is, $\mathcal{F}$ is a finite collection of sets in Ω that are closed under finite set theoretic operations. To make it more concrete, let us give an example of such a field, which is the power-set of a finite, countable set of atoms (also called categories in the sequel). A set F in $\mathcal{F}$ will be called an *atom* only if its subset is also a field member of itself and the empty set, that is, it cannot be broken up into smaller pieces that are also present in the field. Let $\mathcal{A}$ denote a collection of atoms of $\mathcal{F}$. Then, one can show that $\mathcal{A}$ consists of exactly all nonempty sets of the form

$$\bigcap_{i=0}^{n-1} F_i^* \tag{4}$$

where F_i^* is either F_i or F_i^c. Let us call such sets *intersection sets*; we observe that any two intersection sets must be disjoint since for at least one i, one intersection set must lie inside F_i and the other within F_i^c. In summary, given any finite field $\mathcal{F}$ of the sample space Ω, we can find a unique collection of atoms $\mathcal{A}$ of the field such that the sets in $\mathcal{A}$ are disjoint, nonempty, and have the space Ω as their union. Thus, $\mathcal{A}$ is a *partition* of Ω. Such a sample space has a *discrete topology* with the basis given by the set of all atoms $\mathcal{A}$.

Now, we turn our attention to continuous distributions. As already mentioned, the underlying space is the *Polish Borel space*. An example of such a space, which will be used in this paper, is the *Euclidean space* endowed with the Euclidean topology. The basis for the Euclidean topology is the set of all open balls in that space. The *Polish space* is a complete, separable, metric space. We will explain what the listed properties of the *Polish spaces* mean.

A space is called *metric* if it is a set A, with elements called *points*, such that for every pair of points in A, there is an associated non-negative number $d(a, b)$ with the following properties:

$$d(a, b) = 0 \text{ if and only if } a = b \tag{5}$$

$$d(a, b) = d(b, a) \quad \textit{symmetry} \tag{6}$$

$$d(a, b) \leq d(a, c) + d(c, b) \text{ all } c \in A \quad \textit{triangle inequality} \tag{7}$$

A set F is said to be *dense* in A if every point in A is a point in F or a limit point of F.

A metric space A is called *separable* if it has a countable dense subset, that is, if there is a discrete set, say B, such that all points in A can be well approximated by points in B. This means that all the points in A are points in B or limits of the points in B. For example, n-tuples of rational numbers are dense in $\mathbb{R}^n$.

A sequence $\{a_n; n = 0, 1, 2, \dots\}$ in A is called a *Cauchy sequence* if for every $\epsilon > 0$, there is an integer N such that $d(a_n, a_m) < \epsilon$ if $n \geq N$ and $m \geq N$. A metric space is *complete* if every Cauchy sequence converges, that is, if a_n is a Cauchy sequence, then there is $a \in A$ for which $a = \lim_{n\to\infty} a_n$.

1.1.2 Probability Spaces

A *probability space* $(\Omega, \mathcal{B}, P)$ is a triple consisting of a sample space Ω, a σ-field $\mathcal{B}$ of subsets of Ω, and a probability measure P defined on the σ-field; $P(F)$ assigns a real number to every member F of $\mathcal{B}$ so that the following conditions are satisfied:

Nonnegativity:

$$P(F) \geq 0, \text{ all } F \in \mathcal{B}, \tag{8}$$

Normalization:

$$P(\Omega) = 1. \tag{9}$$

Countable Additivity:

$$\begin{gathered}\text{If } F_i \in \mathcal{B},\ i = 0, 1, 2, \dots \text{ are disjoint, then} \\ P\Big(\bigcup_{i=0}^{\infty} F_i\Big) = \sum_{i=0}^{\infty} P(F_i).\end{gathered} \tag{10}$$

1.1.3 Densities

The probability density function (PDF) is a measure theoretic term defined through the *Radon-Nikodym theorem*.

A measure m is said to be *absolutely continuous* with respect to another measure P on the same measurable space, formally $m \ll P$, if $P(F) = 0$ implies $m(F) = 0$.

Theorem (*Radon-Nikodym theorem*)

Given the two measures m and P on a measurable space $(\Omega, \mathcal{F})$ such that $m \ll P$, there exists a measurable function $h : \Omega \to \mathbb{R}$ with the property that $h \geq 0$ such that

$$m(F) = \int_F h dP, \text{ all } F \in \mathcal{B}. \tag{11}$$

The function h is called the Radon-Nikodym *derivative* or *density* of m w.r.t. P and is denoted by $\frac{dm}{dP}$. If $\int f dP = 1$, then f is called a *probability density function*.

2 Discrete Distributions

For the statement of all results, as well as notations conventions, we refer the reader to the paper [1]. The results in the article [1] include discussion of the sun rise problem by Pierre-Simon Laplace from the essay [13] and some applications of the theory. We reference the paper [1] in its entirety. Here, only extensions of the material from the paper [1] are included, which have not found place there due to the form of a conference paper. However, for the readers convenience we provide a short summary of the notations used in the derivations:

- K - number of categories, possibly a hidden variable
- M - number of random experiments

- S - the partition, set of all categories
- $\{p_i\}_{i\in S}$ - multinomial proportions
- $X = \{x_i\}_{(i\in[1,M],x_i\in S)}$ - observations from the random experiments
- $|A|$ - cardinality of the set A
- UNIQUE(X) - unique elements in the multiset X
- $Z = |\text{UNIQUE}(X)|$ - the diversity index
- $P_i = |\{x_j : x_j = i\}|$ - counts of the observations

2.0.1 Derivations of the results for the uniform distribution

Assumption of uniform distribution is restrictive. However, it gives initial insight into the problem and, thus, is briefly presented here. As already introduced by X, we denote the sequence of observations. We can show that conditional probability of this sequence given the hypothetical K is equal to:

$$p(X|K) = \binom{K}{Z}\binom{M}{P_1 P_2 \cdots P_3}\frac{1}{K^M}. \tag{12}$$

It can be seen that the maximum likelihood estimate for the hypothetical number of categories, does not depend on the middle term, which includes the multinomial coefficient. Thus, the estimate can be obtained by,

$$K_{\text{ML}} = \underset{K}{\operatorname{argmax}}\left[\binom{K}{Z}K^{-M}\right]. \tag{13}$$

It is convenient to introduce another quantity, which plays an important role in our theory. This quantity is the *generalization coefficient* N, which equals by definition:

$$N \equiv \frac{M}{Z}. \tag{14}$$

The ML estimate of K can then be obtained by solving the following equation:

$$\frac{1}{v}\ln\left(\frac{1}{1-v}\right) = N, \tag{15}$$

where:

$$v = \frac{Z}{K_{\text{ML}}}, \tag{16}$$

is the fraction of the number of observed categories to the number of all categories. The condition for the likelihood to be monotonically decreasing is as follows:

$$M > \log_{\left(\frac{Z+1}{Z}\right)}(Z+1). \tag{17}$$

If this condition holds, the ML estimate for K is equal to Z. The derivations of the above conditions are in place next. First, we note a property that establishes the link with the known in the statistical literature problem of coupon collector [14]:

$$\lim_{K\to\infty}\left(\frac{K\times H(K)}{\log_{\left(\frac{K+1}{K}\right)}(K)}\right) = 1, \tag{18}$$

where one can easily recognize that $K \times H(K)$ is the expected number of trials before the coupon collector collects the whole collection. In the above expression, $H(K)$ is the harmonic number, equal by definition:

$$H(K) \equiv \sum_{i=1}^{K}\frac{1}{i}. \tag{19}$$

The main vehicle of the derivation is the following expression that is valid for the harmonic numbers [15]:

$$\sum_{i=1}^{K}\frac{1}{i} = C + \ln K + \frac{1}{2K} - \sum_{i=2}^{\infty}\frac{A_i}{K(K+1)\cdots(K+i-1)}, \tag{20}$$

where C is the Euler-Mascheroni constant. It can be seen that asymptotically, as K approaches infinity, the terms after the logarithmic term vanish to zero. This leads to the following property:

$$\lim_{K\to\infty}\left(\sum_{i=1}^{K}\frac{1}{i} - \ln K\right) = C. \tag{21}$$

The derivation for the condition (15) begins with taking the logarithm of the considered expression (13):

$$\ln[p(X|K)] = \sum_{i=1}^{K}\ln i - \sum_{i=1}^{Z}\ln i - \sum_{i=1}^{K-Z}\ln i - M\ln K. \tag{22}$$

Suppose, i is a continuous variable, which setting follows from allowing that variable to take on non-integer values. It can be seen that the middle sum does not depend on K; therefore, derivative w.r.t that variable reads:

$$\frac{d}{dK}\ln[p(X|K)] = \sum_{i=1}^{K}\frac{1}{i} - \sum_{i=1}^{K-Z}\frac{1}{i} - \frac{M}{K} = \ln K - \ln(K-Z) - \frac{M}{K}. \tag{23}$$

The last expression allows us to immediately state the (15) and (16). Equations (15) and (16) allow us to conclude that the sample length needed to learn a given percent of the categories S is a multiple of K. Setting $Z = M$, we see that, indeed, this ($Z = M$) is the sufficient and necessary condition for optimal K approaching infinity. This is due to the following identity:

$$\lim_{K\to\infty}\frac{K}{M}\ln\left(\frac{1}{1-\frac{M}{K}}\right) = 1. \tag{24}$$

It remains to prove the (17). In this case, the maximum of the likelihood should be attained at $K = Z$. Thus, we have the following inequality:

$$\ln[p(X|Z)] > \ln[p(X|Z+1)], \tag{25}$$

which implies:

$$\ln(Z+1) - M\ln(Z+1) < -M\ln Z, \tag{26}$$

and, after some algebra, we attain at the (17). From the above reasoning all the results pertaining to the discrete, uniform distribution from the paper [1] can be deduced. In Figure 1, we illustrate the dependence of the data amount M needed to learn a given percent of all categories.

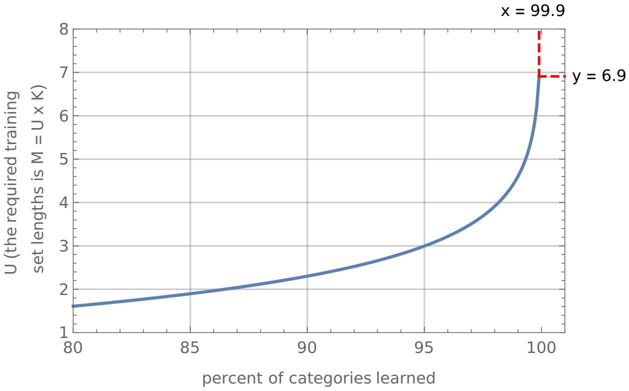

Figure 1: Data amounts requirement to learn a given percent of categories in case of uniform distribution

Next, we discuss the case of the non-uniform distribution, which is more relevant for real-world applications.

2.0.2 Derivations of the results for the non-uniform distribution

The next step will be the derivation of the conditions analogous to that introduced in the previous section, which are distribution-free (we relax the assumption of categories of equal probabilities). The desired effect will be achieved by using the earlier introduced multinomial proportions $\{p_i\}_{i \in S}$. Since we do not impose a constraint on the multinomial proportions, the considered probability function is now in the form,

$$p(X|K, p_1, \cdots, p_K) = \binom{M}{P_1 P_2 \cdots P_Z} \sum_{\{\mathbf{k}\,:\,\text{combinations of } Z \text{ objects out of } K \text{ objects}\}} \prod_{i=1}^{Z} p_{k_i}^{P_i}, \tag{27}$$

where $\mathbf{k} = [k_1, \cdots, k_Z]$. We integrate the above function over a unit simplex D:

$$D = \left\{ \mathbf{p} : \sum_{i=1}^{K} p_i = 1, p \in \mathbb{R}_+^K \right\}. \tag{28}$$

This corresponds to the assumption that all PMFs are equally likely, meaning we assume that we do not know the true PMF and attach to each possible $\mathbf{p} = [p_1, \cdots, p_K]$ an equal weight (we assume they are equally probable):

$$p(X|K) = \frac{1}{\text{vol}(D)} \int_D p(X|K, \mathbf{p}) d\mathbf{p} = \binom{K}{Z} \binom{M}{P_1 P_2 \cdots P_Z} \frac{1}{\text{vol}(D)} \int_D p_1^{P_1} \times \cdots \times p_Z^{P_Z} d\mathbf{p}, \tag{29}$$

where the expression (29) follows the fact, that the value of the integral does not depend on the choice and order of the probabilities in the monomial integrand. To proceed with the derivation, we compute the integral in (29) using Brion's formulae; please see [16]:

$$\frac{1}{\text{vol}(D)} \int_D p_1^{P_1} \times \cdots \times p_Z^{P_Z} d\mathbf{p} = (K-1)! \frac{\prod_{i=1}^{Z} P_i!}{(M+K-1)!}. \tag{30}$$

In light of the above expression, we get,

$$p(X|K, M) = \frac{K! M! \Gamma(K)}{Z!(K-Z)! \Gamma(K+M)}, \tag{31}$$

and next,

$$p(Z|K, M) = \frac{M \Gamma(K) \Gamma(K+1) \Gamma(M+Z)}{\Gamma(Z+1)^2 \Gamma(K+M) \Gamma(K-Z+1)}. \tag{32}$$

To get the probability of the hypothetical number of categories, given M and Z, we apply the following derivation:

$$p(K|Z, M) = \frac{p(Z|K, M) p(K)}{p(Z|M)} = \frac{p(Z|K, M) \times C}{\sum_{K=Z}^{\infty} p(Z|K, M) \times C} = \frac{p(Z|K, M)}{\sum_{K=Z}^{\infty} p(Z|K, M)}, \tag{33}$$

where C is some constant (not to be confused with the Euler-Mascheroni constant used earlier in this document). Evaluation of (33) yields the following:

$$p(K|Z, M) = \frac{\Gamma(K) \Gamma(K+1) \Gamma(M-1) \Gamma(M)}{\Gamma(Z) \Gamma(Z+1) \Gamma(K+M) \Gamma(K-Z+1) \Gamma(M-Z-1)}. \tag{34}$$

From the reasoning presented above we can state the following:

$$s = \frac{u}{u+1}, \quad s = \frac{Z}{K}, \quad u = \frac{M}{K}. \tag{35}$$

Also, all results pertaining to non-uniform distribution, stated in the paper [1], are a consequence of the above reasoning. Based on (35) we can plot the dependence of the amount of data needed to learn given percent of all categories. This dependence is shown in Figure 2.

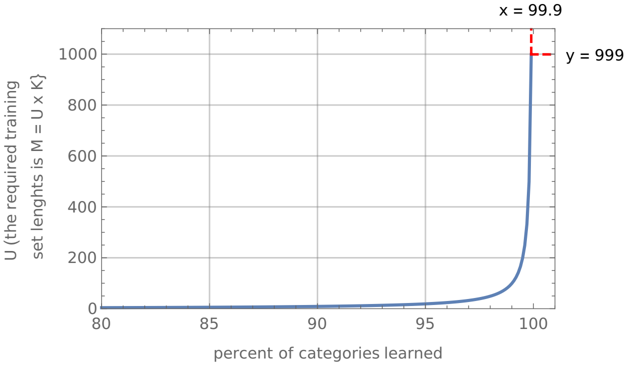

Figure 2: Data amounts requirement to learn a given percent of categories in case of non-uniform distribution

3 Continuous Distributions

The prominent method of density estimation for continuous distribution are Gaussian mixture models (GMM). The GMMs can approximate a PDF given by some training vectors drawn from that PDF. A wide-spread fitting procedure of the GMMs to the given training vectors is based on the Expectation-Maximization (EM) algorithm. Both, the mathematical definition of GMMs and the fitting procedure based on the EM is explained in detail in the book [2]. The neat property of the GMMs is that their expressive power allows modeling any PDF to any accuracy, given enough training vectors are available. However, the traditional fitting procedure is susceptible to overtraining if too many components are chosen. Thus, in this paper, we study a fitting method that does not overfit, and, secondly, allows for the estimation of the number of components given the newly introduced information-theoretic criterion. To reach the goal of formulating the criterion, we first develop a mathematical approach to compute the expected Kullback-Leibler (KL) divergence for the single multivariate Gaussian, whereas the expectation is taken over given number M of samples used to estimate the multivariate Gaussian. In this section, we first develop the theory of expected KL divergence; next, we provide the modified training procedure and end the section with a numerical study.

3.1 Computation of expected Kullback-Leibler divergence

To the best of the authors' knowledge, the computation of the expected Kullback-Leibler divergence for multivariate Gaussian variables has been never previously explored in the literature. Thus, this paper deals with this problem, which can be formally expressed as,

$$\mathrm{E}_{\mathrm{KL}}(M) = \mathbb{E}_{x_i\sim p(x)}\left[D_{\mathrm{KL}}(p(x)\|\hat{q}(x, x_1, \cdots, x_M))\right], \tag{36}$$

where $p(x)$ and $q(x)$ are multivariate Gaussians.

The solution to the problem indicated above can be used in the context of the Minimum Description Length (MDL) cf. [17] or Minimum Discrimination Information (MDI), cf. [18]. MDI and MDL principles have been used often for solving computational learning problems (see [19]–[23]).

The well-known result from the statistics is the following formula for the KL divergence between two multivariate Gaussians:

$$D_{\mathrm{KL}}(p\|q) = \int p(x)\log\frac{p(x)}{q(x)}\mathrm{d}x, \tag{37}$$

for $p(x)$ and $q(x)$ given as:

$$p(x) = \mathcal{N}(x|\mu_1, \Sigma_1) \quad \text{and} \quad q(x) = \mathcal{N}(x|\mu_2, \Sigma_2), \tag{38}$$

where:

$$\mathcal{N}(x|\mu, \Sigma) = \det(2\pi\Sigma)^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right), \tag{39}$$

and finally (see [24]):

$$D_{\mathrm{KL}}(p\|q) = \\ \frac{1}{2}\Big[\log\left(\frac{\det(\Sigma_2)}{\det(\Sigma_1)}\right) - d + \mathrm{tr}(\Sigma_2^{-1}\Sigma_1) + (\mu_2-\mu_1)^T\Sigma_2^{-1}(\mu_2-\mu_1)\Big], \tag{40}$$

where d is the dimension of PDFs p and q.

Next, let us assume that the positively defined and symmetric Σ_1 is given, μ_1 is a zero vector, and Σ_2 and μ_2 are estimated from samples x_i, where $x_i \sim \mathcal{N}(x|\mu_1, \Sigma_1)$:

$$\mu_2 = \tfrac{1}{M}\sum_{i=1}^{M} x_i \quad \Sigma_2 = \tfrac{1}{M}\sum_{i=1}^{M}(x_i-\mu_1)(x_i-\mu_1)^T = \tfrac{1}{M}\sum_{i=1}^{M} x_i x_i^T, \tag{41}$$

where M is the number of samples (length of the observation). We also assume that Σ_2 is positively definite and thus well-conditioned (otherwise KL divergence would go to infinity). This assumption leads to the requirement that M has to be greater than the dimension of Σ_1 when Σ_2 is non-diagonal.

As already indicated in (36), we are looking for the expectation of the Kullback-Leibler divergence over finite sample:

$$\mathbb{E}_{x_i\sim p(x)}\left[D_{\mathrm{KL}}(p(x)\|\hat{q}(x, x_1, \cdots, x_M))\right],$$

where $p(x)$ and $q(x)$ are given in (38).

We consider two cases: the first case is when the covariance matrix is diagonal, and the second case, which is a more complicated one, is when the covariance matrix is full.

3.1.1 Diagonal Case

At first, it is worth noticing that the expectation of the Kullback-Leibler divergence does not depend on a specific form of the covariance matrix Σ_1. Let us introduce the following substitution:

$$x_i = \Sigma_1^{\frac{1}{2}} y_i \quad \text{where} \quad y_i \sim \mathcal{N}(y|0, I), \tag{42}$$

which, in turn, under the assumption given by the (41), gives:

$$\mathbb{E}\left[\log\left(\frac{\det(\Sigma_2)}{\det(\Sigma_1)}\right)\right] = \mathbb{E}[\mathrm{tr}(\log\Sigma)], \tag{43}$$

where:

$$\Sigma = \frac{1}{M}\sum_{i=1}^{M}\mathrm{diag}(y_i \odot y_i), \tag{44}$$

where $\odot$ denotes the element-wise product, the Hadamard product. It is worth mentioning that we have used the following identity (see [25]):

$$\log(\det(\Sigma)) = \mathrm{tr}(\log\Sigma). \tag{45}$$

Since Σ is diagonal and y_i has a zero mean vector, we conclude,

$$\begin{aligned}\mathbb{E}[\mathrm{tr}(\log\Sigma)] = &\\ &\mathbb{E}\left[\sum_{k=1}^{d}\log\left(\frac{1}{M}\sum_{i=1}^{M}y_i(k)^2\right)\right] = \\ &\mathbb{E}\left[\sum_{k=1}^{d}\log(\sigma_M(k)^2)\right] = \\ &= d\cdot\mathbb{E}[\log(\sigma_M(1)^2)] = \\ &d\left(\psi\left(\frac{M-1}{2}\right) + \log 2 - \log(M-1)\right),\end{aligned} \tag{46}$$

where $\psi(n)$ is the digamma function.

The next expression, which appears in (40), is:

$$\mathrm{tr}(\Sigma_2^{-1}\Sigma_1) = \mathrm{tr}(\Sigma^{-1}), \tag{47}$$

where Σ is defined as in (44). After a number of transformations, we get

$$\mathbb{E}[\mathrm{tr}\Sigma^{-1}] = \mathbb{E}\left[\sum_{k=1}^{d}\tfrac{1}{\sigma_M^2(k)}\right] = d\cdot\mathbb{E}\left[\tfrac{1}{\sigma_M^2(1)}\right] = d\tfrac{1}{M-2}. \tag{48}$$

The last expression, which needs to be calculated, is

$$\mathbb{E}[\mu_2^T \Sigma_2^{-1} \mu_2] = \mathbb{E}\left[\sum_{k=1}^{d} \frac{\bar{x}(k)^2}{\sigma_M(k)^2}\right] = d \cdot \mathbb{E}\left[\frac{\bar{x}(1)^2}{\sigma_M(1)^2}\right] = d\frac{M-1}{M\cdot(M-2)}. \quad (49)$$

And finally, our derived formula reads,

$$\mathrm{E}_{\mathrm{KL}}(M) = \frac{d}{2}\left[\psi\left(\frac{M-1}{2}\right) + \log 2 - \log(M-1) - 1 + \frac{M}{M-2} + \frac{M-1}{M\cdot(M-2)}\right]. \quad (50)$$

3.1.2 Non-diagonal Case

In the non-diagonal covariance matrix case, we first introduce the same substitution as in the diagonal case, which is shown in (42):

$$x_i = \Sigma_1^{\frac{1}{2}} y_i \quad \text{where} \quad y_i \sim \mathcal{N}(y|0, I). \quad (51)$$

We start by calculating the logarithmic expression, which appears in (40). By using identity in (45), and transforming the expression using (41), as previously, we get,

$$\log\left(\frac{\det(\Sigma_2)}{\det(\Sigma_1)}\right) = \mathrm{tr}(\log \Sigma). \quad (52)$$

To calculate the expectation of this expression, we use the expansion of the matrix logarithm into the power series, see [26]:

$$\log A = \sum_{k=1}^{\infty} (-1)^{k+1} \frac{(A-I)^k}{k}, \quad (53)$$

and, after a number of transformations, we get,

$$\begin{aligned} &\mathbb{E}[\mathrm{tr}(\log \Sigma)] = \\ &\qquad \mathbb{E}[\mathrm{tr}(\log m\Sigma - \log mI)] = \\ &\mathbb{E}\left[\mathrm{tr}\left(\sum_{k=1}^{\infty}(-1)^{k+1}\frac{1}{k}\sum_{j=0}^{k}\binom{k}{j}(-1)^j m^{k-j}\Sigma^{k-j} - \log mI\right)\right], \quad (54)\end{aligned}$$

where m is chosen so as to $\|m\Sigma - I\| < 1$ is satisfied.

We further develop our approximation by introducing the following auxiliary function:

$$z(k, k_{max}) = \begin{cases} \sum\limits_{i=1}^{k_{max}} (-1)^{k+1} \cdot \binom{i}{k} \cdot \frac{1}{i} & \text{when } k = 0 \\ \sum\limits_{i=k}^{k_{max}} (-1)^{k+1} \cdot \binom{i}{k} \cdot \frac{1}{i} & \text{when } k > 0 \end{cases}, \quad (55)$$

and, consequently:

$$\mathbb{E}[\mathrm{tr}(\log \Sigma)] = \sum_{k=0}^{k_{max}} z(k, k_{max}) \cdot m^k \cdot \mathbb{E}\left[\mathrm{tr}\left(\Sigma^k\right)\right] - d \log m, \quad (56)$$

where d is a dimension of the covariance matrix. To proceed further, we calculate the expectation of $\mathrm{tr}(\Sigma^k)$. It is worth noticing that the integral

$$\int_{-\infty}^{+\infty} t^n \mathcal{N}(t|0,1)\mathrm{d}t = \frac{1}{\sqrt{2\pi}} 2^{\frac{n-1}{2}}((-1)^n + 1)\Gamma\left(\frac{n+1}{2}\right), \quad (57)$$

vanishes for odd n. At this point, we need to calculate the number of monomials with only even exponents that appear in $\mathrm{tr}(\Sigma^k)$.

Now we can describe the algorithm for calculating the number of such monomials. First, we introduce the following auxiliary matrix:

$$A = \frac{1}{\sqrt{M}}\Big[y_1\Big|\ldots\Big|y_M\Big]. \quad (58)$$

One can easily notice that:

$$\begin{aligned} \Sigma &= \tfrac{1}{M}\sum_{i=1}^{M} y_i y_i^T = A \cdot A^T, \\ \Sigma^n &= \underbrace{(AA^T)\ldots(AA^T)}_{n}, \end{aligned} \quad (59)$$

which leads to formula for the specific element, ${\Sigma^n}_{ij}$, of the covariance matrix raised to the n-th power:

$$\begin{aligned} {\Sigma^n}_{ij} &= A_{i*} \cdot A^T \ldots A \cdot A_{*j}^T = \\ &\sum_{s_1=1}^{d}\sum_{k_1=1}^{M} a_{ik_1} \cdot a_{k_1 s_1}^{(T)} \ldots \sum_{s_{n-1}=1}^{d}\sum_{k_n=1}^{M} a_{s_{n-1}k_n} \cdot a_{k_n j}^{(T)} = \\ &= \sum_{s_1,\ldots s_{n-1}} \sum_{k_1,\ldots,k_n} a_{ik_1} a_{s_1 k_1} a_{s_1 k_2} \ldots a_{s_{n-1}k_n} a_{jk_n}, \end{aligned} \quad (60)$$

The number of the above configurations with only even exponents is equal to the number of even partitions of $2n$. For such a partition, we have to calculate the number of monomials separately. The detailed description of the algorithm for counting monomials is presented in section A. Implementation of the algorithm for counting monomials and computing expected KL divergence is available (see [27]). Below, we have shown examples of the formulas derived by the above-mentioned algorithm, for the power of $n = 3$, where d is a dimension of the covariance matrix and M is the number of samples used to estimate the matrix:

$$\begin{gathered} \text{Partition: } 2\ 2\ 2 \\ dM(M-1)(M-2) + dM(d-1)(d-2) + 3dM(M-1)(d-1) \\[1ex] \text{Partition: } 4\ 2 \\ 3dM(M-1) + 3dM(d-1) \\[1ex] \text{Partition: } 6 \\ dM \end{gathered} \quad (61)$$

Once we have calculated the number of monomials for each power n, it is relatively easy to calculate the expectation in (56).

The second expression, which has to be calculated, is

$$\mathrm{tr}\left(\Sigma_2^{-1}\Sigma_1\right). \quad (62)$$

We start in the same way as in the diagonal case:

$$\mathrm{tr}\left(\Sigma_2^{-1}\Sigma_1\right) = \mathrm{tr}\Sigma^{-1}. \quad (63)$$

The next step is to expand inversion of the covariance matrix into the Neumann series (see [28]):

$$\Sigma^{-1} = m \sum_{k=0}^{\infty} (I - m\Sigma)^k, \tag{64}$$

where m is selected so that $\|I - m\Sigma\| < 1$ is satisfied. After several transformations, we get

$$\Sigma^{-1} \approx \sum_{k=0}^{k_{max}} (-1)^k m^{k+1} \binom{k_{max}+1}{k+1} \Sigma^k, \tag{65}$$

which finally gives,

$$\mathbb{E}[\mathrm{tr}\Sigma^{-1}] \approx \sum_{k=0}^{k_{max}} (-1)^k m^{k+1} \binom{k_{max}+1}{k+1} \mathbb{E}[\mathrm{tr}\Sigma^k]. \tag{66}$$

We note that the resulting expression has a similar form to (56), with the difference of the k-dependent expression. The last expression for which the expectation needs be calculated is

$$(\mu_2 - \mu_1)^T \Sigma_2^{-1} (\mu_2 - \mu_1). \tag{67}$$

By using substitution (41), and subsequently (42), we get

$$\begin{aligned} &\mu_2^T \Sigma_2^{-1} \mu_2 = \\ &\left(\frac{1}{M} \sum_{i=1}^{M} \Sigma_1^{\frac{1}{2}} y_i \right)^T \Sigma_2^{-1} \left(\frac{1}{M} \sum_{i=1}^{M} \Sigma_1^{\frac{1}{2}} y_i \right) = \\ &\left(\frac{1}{M} \sum_{i=1}^{M} y_i \right)^T \Sigma^{-1} \left(\frac{1}{M} \sum_{i=1}^{M} y_i \right), \end{aligned} \tag{68}$$

then, given (58), and by expanding the inverse of the covariance matrix as in (64) into the Neumann series, we get:

$$\begin{aligned} &\mu_2^T \Sigma_2^{-1} \mu_2 = \\ &\sum_{i,j} \left[\frac{1}{M^2} A^T \Sigma^{-1} A \right]_{ij} \approx \\ &\frac{1}{M^2} \sum_{i,j} \sum_{k=0}^{k_{max}} (-1)^k m^{k+1} \binom{k_{max}+1}{k+1} A_{i*}^T \Sigma^k A_{*j}. \end{aligned} \tag{69}$$

Now, we need to calculate the following expression:

$$A_{i*}^T \Sigma^n A_{*j}. \tag{70}$$

By multiplying out we get:

$$\begin{aligned} &A_{i*}^T \Sigma^n A_{*j} = \\ &\sum_{s_0=1}^{d} \sum_{s_n=1}^{d} a_{is_0}^{(T)} \left[\Sigma^n\right]_{s_0 s_n} a_{s_n j} = \\ &\sum_{s_0} \sum_{s_n} a_{s_0 i} \sum_{s_1,\ldots s_{n-1}} \sum_{k_1,\ldots,k_n} a_{s_0 k_1} a_{s_1 k_1} a_{s_1 k_2} \ldots a_{s_{n-1} k_n} a_{s_n k_n} a_{s_n j} = \\ &= \sum_{s_1,\ldots s_{n+1}} \sum_{k_1,\ldots,k_n} a_{s_1 i} a_{s_1 k_1} \ldots a_{s_{n+1} k_n} a_{s_{n+1} j}. \end{aligned} \tag{71}$$

An important observation is in place, namely, the monomials cannot have only even exponents if $i \neq j$. This allows us to bring down the calculation of the expectation of the above expression to the expectation of the trace of Σ^{n+1}. Thus, finally, we get

$$\mathbb{E}[\mu_2^T \Sigma_2^{-1} \mu_2] \approx \tfrac{1}{M} \sum_{k=0}^{k_{max}} (-1)^k m^{k+1} \tbinom{k_{max}+1}{k+1} \mathbb{E}[\mathrm{tr}\Sigma^{k+1}]. \tag{72}$$

Obviously, the algorithm for calculating the above expression is the same as in the case of the two previously examined terms.

3.2 Procedure

Here, we show how to estimate the GMM using the modified training procedure. The introduced procedure solves the problem of overtraining. First, we focus on training the GMM with fixed K - the number of components in the mixture. Based on this development, we formulate a method to select the optimal number of components for a given training set without resorting to the development sets. The classical Expectation-Maximization algorithm, cf. [2], produces in each iteration more and more accurate estimates of the mean vectors, covariance matrices, and weights. We denote the trajectory of these parameters in subsequent iterations as,

$$\theta = \left[\theta^{(1)}, \cdots, \theta^{(J)}\right], \tag{73}$$

where J is the maximum acceptable number of iterations (or the number of iterations until convergence), and

$$\theta^{(j)} = \left[\Sigma_1^{(j)}, \cdots, \Sigma_K^{(j)}, \mu_1^{(j)}, \cdots, \mu_K^{(j)}\right], \tag{74}$$

with j the iteration number, Σ_i denoting covariance matrix for the $i-$ th component, μ_i denoting the mean vectors for the $i-$ th component, and K the number of components in the Gaussian mixture. Moreover, we denote $\theta_k^{(j)} = [\Sigma_k^{(j)}, \mu_k^{(j)}]$. To proceed further, we introduce and define some mathematical entities:

- the *entropy* in nats of the categorical distribution $\{r_i\}_i$ is defined as $H(\{r_i\}) = -\sum_i r_i \log r_i$
- the *kernel width* is the effective number of samples used to estimate a given component. Given the samples $\{x_i\}_{i=1}^M$, we compute the *kernel width*, $w(\theta, \{x_i\})$, also abbreviated by $w(\theta)$, using the following set of formulas:

$$r_i = \frac{\mathcal{N}(x_i|\theta)}{\sum_l \mathcal{N}(x_l|\theta)}, \tag{75}$$

$$w(\theta) = \exp\left(H(\{r_i\})\right), \tag{76}$$

- Gaussian *component differential entropy* in nats, $H(\theta)$, will be defined as:

$$H(\theta) = \frac{1}{2} \log\left(\det(2\pi e \Sigma)\right), \tag{77}$$

- the *score* or *cross entropy* for the component is expressed by the following formula:

$$s(\theta) = H(\theta) + \mathrm{E_{KL}}\left(w(\theta)\right). \tag{78}$$

We see that the expression for *cross entropy* depends only on the observed quantities.

Equipped with the above definitions, we can formulate the algorithm,

```
Algorithm 1: train GMM

Require: starting parameters values $\theta^{(0)}$, maximal number of iterations $J$, initial scores $s\left(\theta_k^{(0)}\right) = \infty$, set all weights to $\rho_k = \frac{1}{K}$.
Ensure: $\theta^{(J)}$ and weights $\{\rho_k\}$.
 1: for $j \in [1, \dots, J]$ do
 2:     compute $\theta^{(j)}$ update according to EM (keeping weights unchanged)
 3:     for $k \in [1, \dots, K]$ do
 4:         if $s\left(\theta_k^{(j)}\right) > s\left(\theta_k^{(j-1)}\right)$ then
 5:             $\theta_k^{(j)} = \theta_k^{(j-1)}$ (backtracking)
 6:         end if
 7:     end for
 8: end for
 9: run few EM updates of weights alone
10: return $\theta^{(J)}$ and weights $\{\rho_k\}$
```

```
Algorithm 2: train GMM

Require: starting parameters values $\theta^{(0)}$, maximal number of iterations $J$, initial scores $s\left(\theta_k^{(0)}\right) = \infty$, set all weights to $\rho_k = \frac{1}{K}$.
Ensure: $\theta^{(J)}$ and weights $\{\rho_k\}$.
 1: for $j \in [1, \dots, J]$ do
 2:     compute $\theta^{(j)}$ update according to EM (keeping weights unchanged)
 3:     for $k \in [1, \dots, K]$ do
 4:         if $s\left(\theta_k^{(j)}\right) > s\left(\theta_k^{(j-1)}\right)$ then
 5:             $\theta_k^{(j)} = \theta_k^{(j-1)}$ (backtracking)
 6:         end if
 7:     end for
 8: end for
 9: ~~run few EM updates of weights alone~~
10: return $\theta^{(J)}$ and weights $\{\rho_k\}$
```

The mechanism behind the algorithm 1 is as follows:

- if the number of components is large in comparison to the number of training vectors, the precision of each component goes up with each iteration until it eventually reaches infinity; this means at the same time, $H(\theta)$ goes toward $-\infty$ (in practice, it will stop at a low value due to the constraint we put on the minimal eigenvalue of the covariance matrix)
- the second term in the expression for *score*, see 78, to the contrary, grows toward $+\infty$ as *kernel width* approaches dimension of the training vectors d. Thus, even if *componenent differential entropy* goes toward $-\infty$, the *cross entropy* has a minimal extreme point and at the very last grows toward $+\infty$.
- thus, the statements from 3-7 in the algorithm 1 prevent the collapse of the components to something resembling a dirac delta
- in effect, the algorithm does not overtrain

We also examine the algorithm 2, which does not run a few EM updates of weights (statement 9 in algorithm 1).

3.2.1 Number of components selection criterion

The *cross entropy* in 78 is the expression on number of nats needed to code from the component. Based on the *cross entropy*, we can get the number of bits needed to code with given fidelity using the Shannon-Lower-Bound [29].

Given the component *weights* $\{\rho_k\}_{k=1}^K$, the average number of nats needed to code from the GMM modeled source PDF is equal to

$$C_1\left(K, \theta^{(J)}, \{\rho_k\}\right) = C_1(K) = \underbrace{H\left(\{\rho_k\}_{k=1}^K\right)}_{\text{components indices entropy}} + \underbrace{\sum_{k=1}^{K} \rho_k s\left(\theta_k^{(J)}\right)}_{\text{mean per component cross entropy}} . \quad (79)$$

We also consider a second criterion, which seems to give meaningful results:

$$C_2\left(K, \theta^{(J)}, \{\rho_k\}\right) = C_2(K) = \underbrace{\sum_{k=1}^{K} \rho_k s\left(\theta_k^{(J)}\right)}_{\text{mean per component cross entropy}} . \quad (80)$$

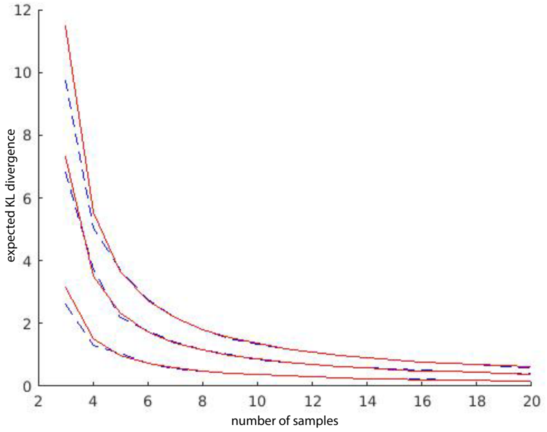

Figure 3: Comparison of the empirically and analytically obtained expectation of the KL divergence for Gaussians with the diagonal covariance matrices. Dashed lines represent the Monte Carlo result. Dimensions of covariance matrices from the beginning of the coordinate axes: 3, 7, and 11, respectively.

To find the optimal number of components K (which minimizes the average number of nats needed to code from the source), we proceed as follows:

- we swap the K from 1 to M
- for each K we run the algorithm 1 or 2.

- we smooth with smoothing splines, see [30], the resulting curve $C_*(K)$, obtaining a smoothed curve $\tilde{C}_*(K)$
- we select $\hat{K}$ for which $\tilde{C}_*(\hat{K})$ attains minimum and returns the optimal number of components.

Above, the asterisk means either 1 or 2.

3.3 Numerical experiments

3.3.1 Expected Kullback-Leibler Divergence

In this section, the comparison our analytical results with the Monte-Carlo obtained curves of expected Kullback-Leibler divergence will be presented. First, the results for the diagonal covariance matrix will be shown in Figure 3.

As can be seen, the analytical expectation shows high accuracy for the diagonal covariance matrices.

The results are a bit worse for the non-diagonal matrix. The following issues affected the accuracy of the result:

- the calculation of the higher degree series expansions was too hard as the computation of the number of even exponent monomials grew exponentially with n. We were able to compute the number of monomials for n as high as eight (the formulas for $n = 8$ when written on A4 page taking 44000 rows).
- The Monte-Carlo curves on the left are sensitive to the threshold for the detection of the semi-definiteness of the matrices. It sometimes happens that one of the eigenvalues of the covariance matrix is very small. Then value of term $\mathrm{tr}\Sigma^{-1}$ grows enormously. It is clear that the result depends on the choice of the threshold.

Figure 4 depicts the results for non-diagonal covariance matrices. To obtain the plot, we rejected matrices with the smallest eigenvalue less than 0.01 (which is around $\frac{1}{100}$ of the largest eigenvalue).

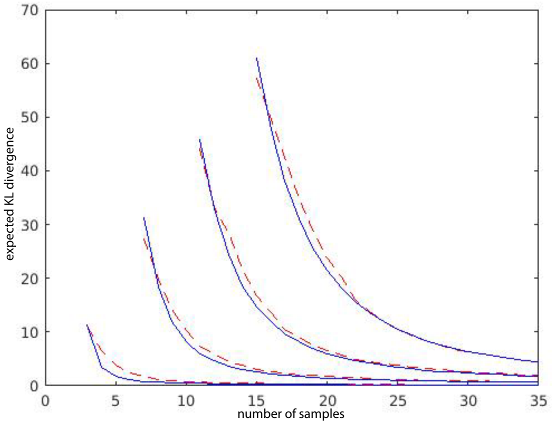

Figure 4: Comparison of empirically and analytically obtained expectation of the KL divergence for Gaussians with the non-diagonal covariance matrices. Dashed lines represent the Monte Carlo estimation. Dimensions of covariance matrices from the beginning of the coordinate axes: 3, 7, 11, and 15, respectively.

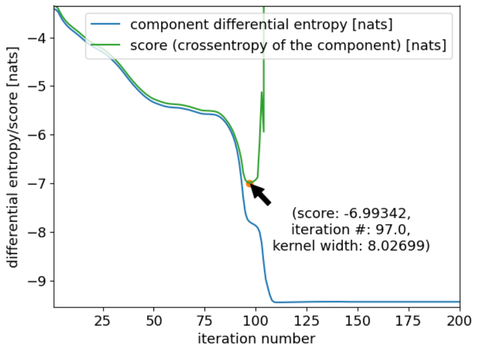

Figure 5: Illustration of the overtraining prevention mechanism. Plot shows that score prevents the overtraining, differential entropy of the Gausssian component goes down and the score starts increasing after kernel width exceeds 8 training points

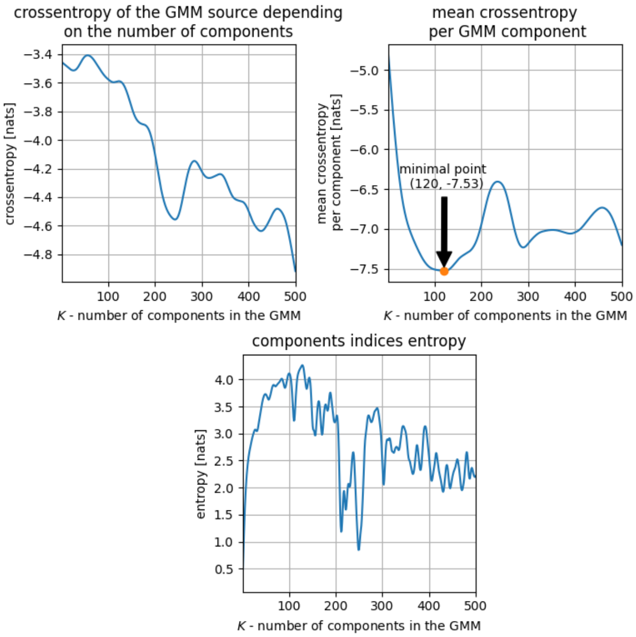

Figure 6: Results of the algorithm 1 for $d = 2$

3.3.2 Overtraining Prevention Mechanism

Here, we present a drawing (Figure 5) showing the trajectory of the *score* and *component differential entropy*. The plot has been obtained by by recording the parameters for subsequent iterations for one chosen component of the GMM. We see that the *score* attains minimum but *component differential entropy* goes to $-\infty$. This allows to use the backtracking procedure present in algorithms 1 and 2.

3.3.3 Numerical Analysis of the Number of Components Selection Criteria

All numerical experiments of this section has been carried out using the Line Spectral Frequencies (LSF) as the data to which the GMMs are fitted. The LSFs were computed from the LibriSpeech database. In all subsequent experiments the training set length M equals 1000.

First, the workings of the algorithm 1 will be presented. The Figure 6 shows the criteria C_1 and C_2 and the *components indices entropy* on pictures from left to right and from top to bottom, respectively. We see erratic behaviour of the *components indices entropy* that falls down, meaning it silences some components by driving some weights to zero. The criterion C_1 suggests that the optimal number of components is > 500, what is unlikely and probably caused by silencing of some components by zero weights. The C_2 criterion, instead, gives a sensible result. We see that for algorithm 2, selected by the C_2 criterion, number of components is similar.

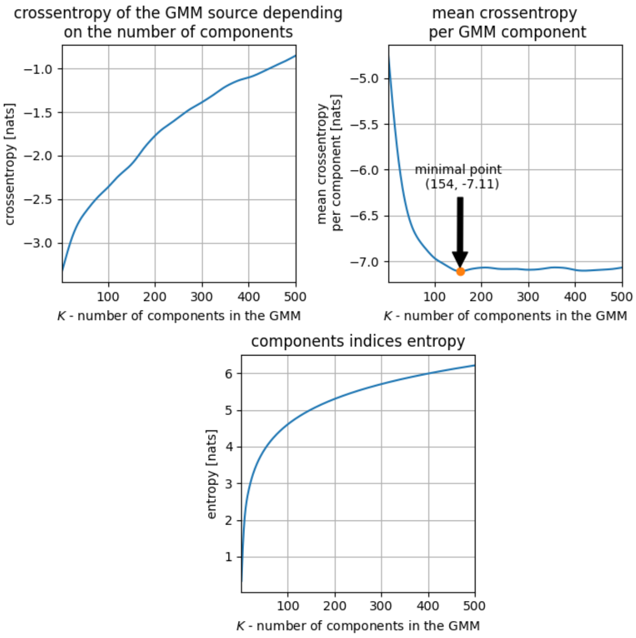

Figure 7: Results of the algorithm 2 for $d = 2$

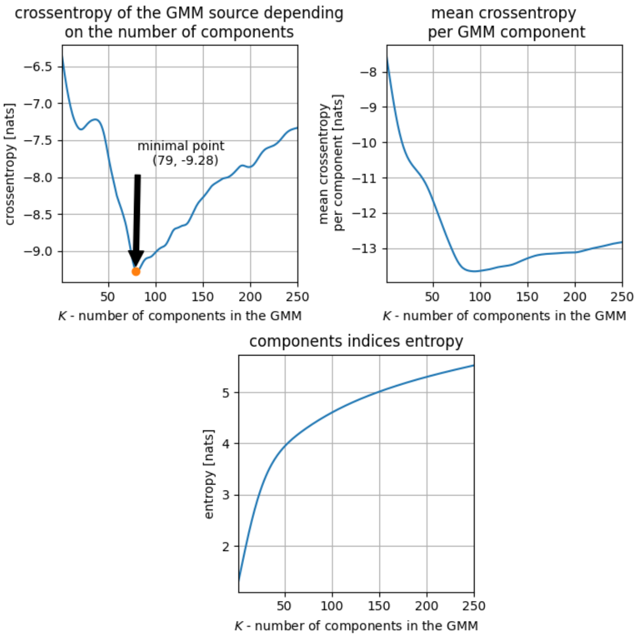

Figure 8: Results of the algorithm 2 for $d = 4$

As the second example, we show the workings of the algorithm 2. The plots in Figure 7 show the same curves as in the case of the Figure 6 (see previous paragraph). The C_1 criterion points as the optimal number of components exactly one component, that is, a single Gaussian. Above $K = 1$, the *components indices entropy* penalizes the *score* to the extent that the score curve is monotonically increasing. Another optimal number of components is suggested by C_2. We computed the GMMs for the number of components selected using the C_2 criterion, for which the contour plots are shown on Figure 10. We see that the GMM is tightly fitted to the data for the selection criterion C_2. However, the composite criterion C_1 suggests a single component, which seems to by trivial, but still optimal, as indicated by the methodology proposed.

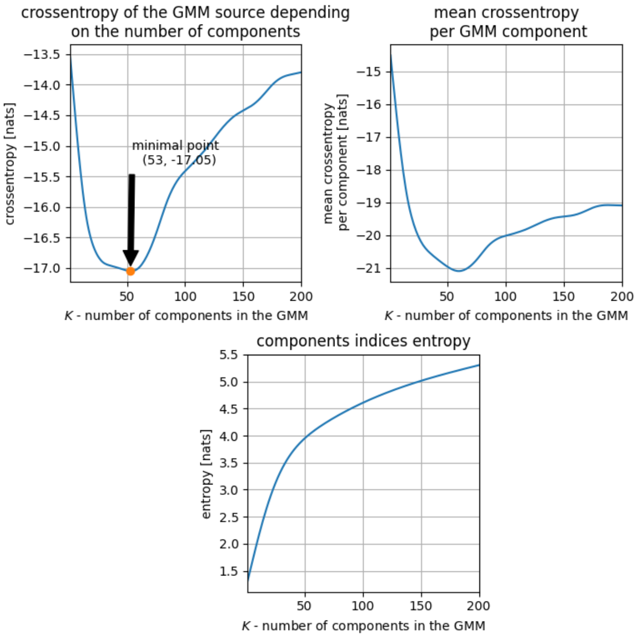

Figure 9: Results of the algorithm 2 for $d = 8$

As the third example, we show the working of the algorithm 2 for the problem dimension $d = 4$ (first four line spectral frequencies). In this case, the criterion C_1 returns trustworthy results, the minimum on the C_1 curve is clearly noticeable. The returned result is sensible and indicates that for $d = 4$, the training vectors distribution needs more components to be modeled accurately; the distribution is probably far from Gaussian. The curves are presented in Figure 8. This experiment is an indication that the proposed methodology is sound. To make the evidence even stronger we present more experiments for dimensions $d = 8$, Figure 9, and $d = 16$, Figure 12. As expected the optimal number of components decreases with the dimension of the problem. This is to maintain proper generalization.

The last result shows the behavior of the classical number of components selection criterion, that is the maximization of the likelihood on the development set. For this experiment, from 1000 samples of the training set, we excluded 100 samples as the development set. In Figure 13, we show the plot of the log-likelihood on the development set as a function of the number of components. It is evident that the plot is quite erratic of what decreases trust in this method of selection of the number of components. The results can

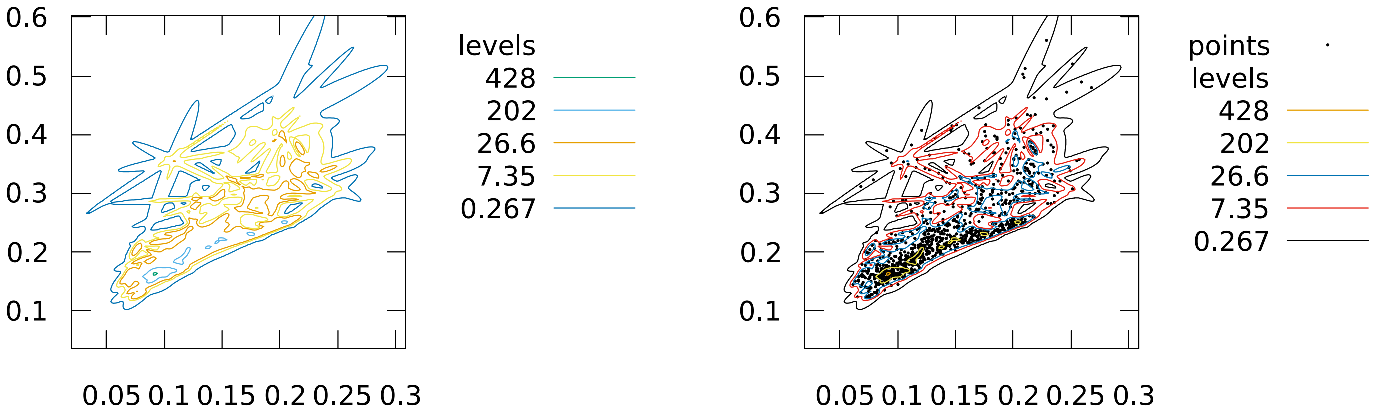

Figure 10: GMM fitted using algorithm 2 with number of components selected using the criterion C_2

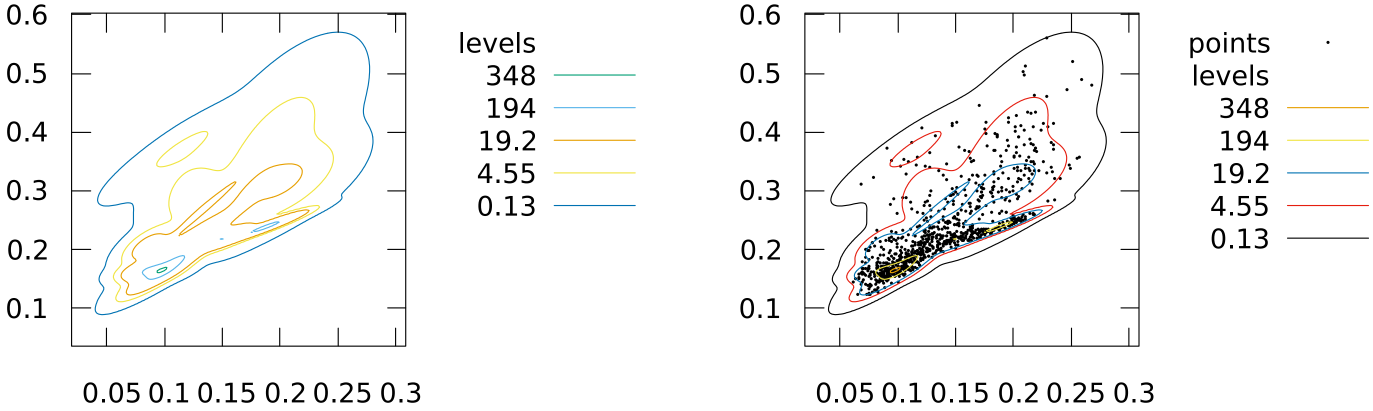

Figure 11: GMM fitted using traditional EM algorithm with number of components selected using tuning on the development set

be made more stable if we allow the development set to be larger. Such a measure will come at the cost of reducing the number of training points. Here, our proposed method shows the advantage of not using development sets. The contour plots for the GMM fitted using traditional method with the number of components tuned on the development set are shown in Figure 11.

4 Conclusions

In this paper, the theory of both discrete and continuous distribution with unknown number of components has been developed. The measure theoretic reason for the inherent similarity between PMFs and PDFs, with the difference in the structure of the underlying sample space, has been given. The difference causes the two distribution categories to be treated with much different mathematical tools. The main result of the paper are the means and methods of computing the number of components, that is, the number of categories in the PMF case and the number of Gaussian components in the GMM modeling in a PDF case, which is given implicitly in terms of its realizations.

Development of the theory required considering the expected Kullback-Leibler divergence - a difficult problem on its own rights. Especially, the algorithm for counting monomials with only even exponents in the expression for a trace of a matrix raised to a power has been developed. This algorithm allows to compute certain integrals analytically without resorting to the Monte-Carlo experiments. This algorithm can also be of interest on its own rights.

We believe that the theory presented in this paper will find many practical applications in diverse fields of science, technology, and engineering as a convenient tool of data analysis.

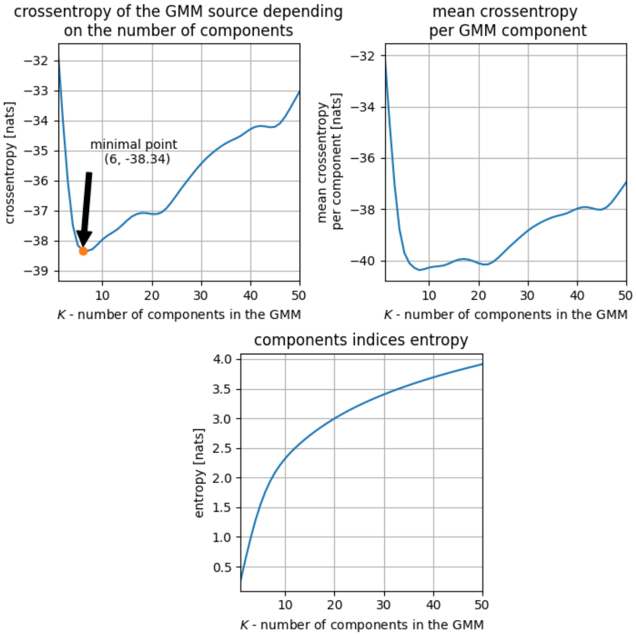

Figure 12: Results of the algorithm 2 for $d = 16$

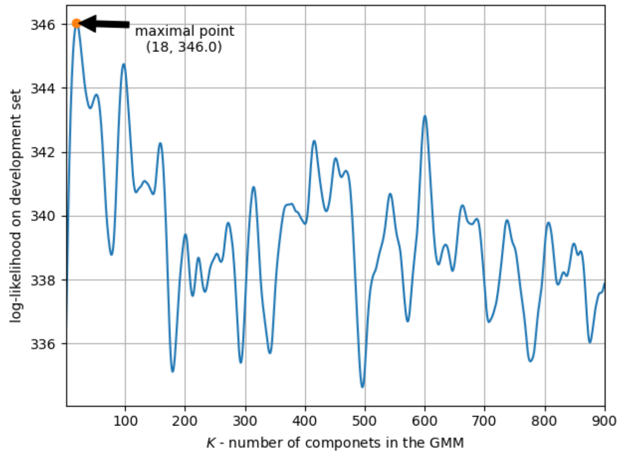

Figure 13: Results of the traditional EM algorithm $d = 2$

References

[1] M. Kuropatwiński, “Estimation of Quantities Related to the Multinomial Distribution with Unknown Number of Categories,” in 2019 Signal Processing Symposium (SPSympo), 277–281, 2019, doi:10.1109/sps.2019.8881992.

[2] C. M. Bishop, Pattern recognition and machine learning, Springer, 2006.

[3] R. M. Gray, Source coding theory, Springer, 2012.

[4] R. M. Gray, “Gauss Mixture Vector Quantization,” in IEEE International Conference on Acoustics, Speech, and Signal Processing, 1769–1772, 2001, doi:10.1109/icassp.2001.941283.

[5] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, et al., The HTK book, Cambridge University Engineering Department, 2002.

[6] D. A. Reynolds, T. F. Quatieri, R. B. Dunn, “Speaker verification using adapted Gaussian mixture models,” Digital Signal Processing, **10**(1), 19–41, 2000, doi:10.1006/dspr.1999.0361.

[7] H. Permuter, J. Francos, I. Jermyn, “A study of Gaussian mixture models of color and texture features for image classification and segmentation,” Pattern Recognition, **39**(4), 695–706, 2006, doi:10.1016/j.patcog.2005.10.028.

[8] A. Kundu, S. Chatterjee, A. S. Murthy, T. Sreenivas, “GMM Based Bayesian Approach To Speech Enhancement in Signal/Transform Domain,” in 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, 4893–4896, 2008, doi:10.1109/icassp.2008.4518754.

[9] M. Nilsson, S. Gustaftson, S. V. Andersen, W. B. Kleijn, “Gaussian Mixture Model Based Mutual Information Estimation Between Frequency Bands in Speech,” in 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing, 525–528, 2002, doi:10.1109/icassp.2002.1005792.

[10] B. W. Silverman, Density estimation for statistics and data analysis, CRC Press, 1986.

[11] L. Devroye, G. Lugosi, Combinatorial methods in density estimation, Springer, 2012.

[12] R. M. Gray, Probability, random processes, and ergodic properties, Springer, 1988.

[13] P. S. Laplace, Philosophical essay on probabilities, Springer, 1998.

[14] P. Flajolet, D. Gardy, L. Thimonier, “Birthday paradox, coupon collectors, caching algorithms and self-organizing search,” Discrete Applied Mathematics, **39**, 207–229, 1992, doi:10.1016/0166-218x(92)90177-c.

[15] S. Gradshteyn, I. Ryzhik, Table of integrals, series, and products, Academic Press, 2014.

[16] V. Baldoni, N. Berline, J. De Loera, M. Köppe, M. Vergne, “How to integrate a polynomial over a simplex,” Mathematics of Computation, **80**(273), 297–325, 2011, doi:10.1090/s0025-5718-2010-02378-6.

[17] J. Rissanen, “Modeling by shortest data description,” Automatica, **14**(5), 465–471, 1978, doi:10.1016/0005-1098(78)90005-5.

[18] S. Kullback, Information theory and statistics, Courier Corporation, 1997.

[19] R. H. Davies, C. J. Twining, T. F. Cootes, J. C. Waterton, C. J. Taylor, “A minimum description length approach to statistical shape modeling,” IEEE Transactions on Medical Imaging, **21**(5), 525–537, 2002, doi: 10.1109/tmi.2002.1009388.

[20] R. M. Gray, J. C. Young, A. K. Aiyer, “Minimum Discrimination Information Clustering: Modeling and Quantization with Gauss Mixtures,” in 2001 International Conference on Image Processing, 14–17, 2001, doi: 10.1109/icip.2001.958039.

[21] M. H. Hansen, B. Yu, “Model selection and the principle of minimum description length,” Journal of the American Statistical Association, **96**(454), 746–774, 2001, doi:10.1198/016214501753168398.

[22] Y. Ephraim, A. Dembo, L. R. Rabiner, “A minimum discrimination information approach for hidden Markov modeling,” IEEE Transactions on Information Theory, **35**(5), 1001–1013, 1989, doi:10.1109/18.42209.

[23] R. M. Gray, A. Gray, G. Rebolledo, J. Shore, “Rate-distortion speech coding with a minimum discrimination information distortion measure,” IEEE Transactions on Information Theory, **27**(6), 708–721, 1981, doi:10.1109/tit.1981.1056410.

[24] J. Duchi, “Derivations for Linear Algebra and Optimization,” Technical report, University of California at Berkeley, 2007.

[25] C. S. Withers, S. Nadarajah, “log det A = tr log A,” International Journal of Mathematical Education in Science and Technology, **41**(8), 1121–1124, 2010, doi:10.1080/0020739x.2010.500700.

[26] B. Hall, Lie groups, Lie algebras, and representations: An elementary introduction, Springer, 2015.

[27] L. Sikorski, M. Kuropatwiński, “Counting Monomials Algorithm,” `https://www.codeocean.com/`, 2019, doi:10.24433/CO.3748956.v1.

[28] D. Zhu, B. Li, P. Liang, “On the Matrix Inversion Approximation Based on Neumann Series in Massive MIMO Systems,” in IEEE International Conference on Communications (ICC), 1763–1769, 2015, doi:10.1109/icc.2015.7248580.

[29] T. Berger, Rate distortion theory: A mathematical basis for data compression, Prentice-Hall, 1971.

[30] C. De Boor, A practical guide to splines, Springer, 1978.

A The algorithm for counting monomials

The solution of the problem from section 3.1.2 is divided into six short algorithms

Algorithm 3: Counting monomials with only even exponents for given even partition of number n

```
Require: Even partition $\mathcal{P} = \{p_1, ..., p_k\}$ of the number $n$, problem dimension $d$,number of random samples $m$
Ensure: Number of monomials with only even exponents
$S := 0$
for every multipermutation of $\mathcal{P}$ do
    Create $\mathcal{T} = \{\underbrace{1, ..., 1}_{p_1}, ..., \underbrace{k, ..., k}_{p_k}\}$
    for every multipermutation of $\mathcal{T}$ do
        Check if $\mathcal{T}$ is correct with Algorithm (7).
        If Algorithm (7) returned false, go to the next multipermutation of $\mathcal{P}$.
        Create matrices $D$ and $M$ for the table $\mathcal{T}$ with Algorithm (4).
        If there are unfilled places in matrices $D$ and $M$, create pairs of matrices $D'$ and $M'$ by filling unfilled places in $D$ and $M$ in every possible way.
        From the set of every generated in previous step pairs of matrices, discard these which are incorrect. Correctness of the pair of matrices is checked with Algorithm (5).
        For every pair of matrices calculate it's numerical value with Algorithm (6) and add it to $S$.
    end for
end for
return $S$
```

Algorithm 4: Creating matrices D and M

```
Require: $\mathcal{T} = \{t_1, ..., t_n\}$, where $t_i \in \{1, ..., k\}$
Ensure: Filled matrices $D$ and $M$
Create matrices $D$ and $M$ of a dimension $k$
for $i = 1$ to $n$ do
    if $i$ is odd then
        To $M_{t_i t_{i+1}}$ and $M_{t_{i+1} t_i}$ write "=".
        To $D_{t_i t_{i+1}}$ and $D_{t_{i+1} t_i}$ write "≠".
    end if
    if $i$ is even then
        To $D_{t_i t_{i+1}}$ and $D_{t_{i+1} t_i}$ write "=".
        To $M_{t_i t_{i+1}}$ and $M_{t_{i+1} t_i}$ write "≠".
    end if
end for
return matrices $D$ and $M$
Note: If there is an occurrence of writing "≠" or "=" in previously filled place, stop the algorithm and return nothing.
```

Algorithm 5: Checking if given pair of matrices D and M is correct

```
Require: Matrices $D$ and $M$
Ensure: True, if given pair is correct; False in other cases
for every pair of indexes $i, j$ of the matrix do
    if $D_{ij} = M_{ij} = "="$ then
        return False
    end if
    if the negation of any from implications listed below is true:
    $[(X_{ij} = "=") \wedge (X_{ik} = "=")] \Rightarrow X_{kj} = "="$ or $[(X_{ij} = "=") \wedge (X_{ik} = "\neq")] \Rightarrow X_{kj} = "\neq"$
    where X is a matrix $D$ or $M$ then
        return False
    end if
end for
return True
```

Algorithm 6: Calculating a component of sum from formula for number of monomials

```
Require: Pair of matrices $D$ and $M$, problem dimension $d$, number of random samples $m$
Ensure: A component of sum from formula for number of monomials
$q = 1$
for $i = 1$ to $\dim D$ do
    if $i = 1$ then
        $q = q \cdot d \cdot m$
    else
        Calculate $c_d$ oraz $c_m$ with Algorithm (8) with input data: $(D, i)$ and $(M, i)$
        if there exists $i < j$, such that $D_{ij} = "\neq"$ then
            $q = q \cdot (d - c_d)$
        end if
        if there exists $i < j$, such that $M_{ij} = "\neq"$ then
            $q = q \cdot (m - c_m)$
        end if
    end if
end for
return q
```

Algorithm 7: Checking if table $\mathcal{T}$ is correct

```
Require: $\mathcal{T} = \{t_1, ..., t_n\}$, where $t_i \in \{1, ..., k\}$
Ensure: True, if $\mathcal{T}$ is correct; False in other cases
$A = \emptyset$
for $i = 1$ to $n$ do
    if there exists $s < i$, such that $t_s > t_i$ and $t_i \notin A$ then
        return False
    end if
    insert $t_i$ into $A$
end for
return True
```

Algorithm 8: Calculating auxiliary minuend

```
Require: Matrix $X$, number $i$
Ensure: Auxiliary minuend
if $i = 1$ then
    return 0
end if
for each $j < i$ do
    if If $X_{ij} = "\neq"$ then
        Calculate $c_j$ with Algorithm (6) with input data $(X, j)$
    end if
end for
$c = \min_j c_j$
return $c + 1$
```