Thèse de Doctorat de l'Université Paris 6

Spécialité:
Mathématiques

présentée

par **M. Jean-Philippe Vert**

pour obtenir le grade de Docteur de l'Université Paris 6

Sujet de la thèse:

Méthodes Statistiques pour la Modélisation du Langage Naturel

Soutenue le 30 mars 2001 devant le Jury composé de:

M. Robert Azencott
M. Lucien Birgé
M. Olivier Catoni
M. Gábor Lugosi
M. Stéphane Mallat
M. Pascal Massart
M. Bernard Prum
M. Alain Trouvé

Remerciements

Je remercie Olivier CATONI pour la passion et la disponibilité dont il a toujours fait preuve. Son enthousiasme mêlé de rigueur scientifique m'a beaucoup appris, et je lui suis reconnaissant d'avoir su me guider sans me brider.

Robert AZENCOTT me fait l'honneur de présider le jury; je le remercie pour l'intérêt qu'il a constamment témoigné à l'égard de ce travail, les nombreux conseils qu'il m'a prodigués et la confiance qu'il m'a maintes fois accordée.

Je remercie Gábor LUGOSI et Pascal MASSART d'avoir accepté d'être rapporteurs de ma thèse, et d'avoir été des sources d'inspirations à travers leurs travaux et les échanges que nous avons eus.

Merci à Lucien BIRGÉ, Stéphane MALLAT, Bernard PRUM et Alain TROUVÉ de faire partie du jury de cette thèse. Je remercie également Bernard PRUM et Cécile COT pour m'avoir fait découvrir les problèmes d'analyse de séquences biologiques en compagnie d'Olivier CATONI.

Gilles BLANCHARD et Olivier BOUSQUET m'ont beaucoup appris dans le domaine de l'apprentissage: je les remercie pour les nombreuses discussions passionnées que nous avons partagées.

Je remercie Stuart GEMAN et Willard MILLER pour l'opportunité qu'ils m'ont offerte de travailler pendant un mois à l'Institute for Mathematics and its Applications de l'Université du Minnessota et d'y participer à un colloque passionnant.

Ce travail a été soutenu par le Corps des Mines, et je remercie Benoît LEGAIT et Marie-Solange TISSIER de m'avoir toujours encouragé dans la voie de la recherche et permis de travailler dans des conditions idéales.

Je remercie également Bénédicte AUFFRAY, Céline BERGER et Roza DÉZÉ du Département de Mathématiques et Applications de l'Ecole Normale Supérieure d'avoir grandement facilité mon travail par leur compétence et leur gentillesse.

Mes collègues et amis ont su créer une atmosphère "Bureau V6" que je regretterai certainement longtemps: merci à Benoît COLLINS, Thomas DUQUESNE, Benoît MSELATI et Jeff QUINT.

Merci enfin à tous mes proches auxquels je pense sans les nommer pour leur soutien sans faille.

Merci surtout à Yasuko et Marina pour tout le reste.

Contents

Chapter 1

Introduction

1.1 Human language technology

Natural language is the language through which humans communicate. It is therefore widely used in our societies for exchange and storage of information. The last decades have witnessed an explosion in the amount of multimedia data available in a digital form, including texts, speech recordings or videos, in which natural language plays a central role to convey information. These data are stored in ever-increasing databases like corporate databases, digital libraries or the World Wide Web, whose sizes and growth rates prevent any human from taking the time to consult the entire set of documents in order to find some information he may be actively or passively looking for.

There is therefore a big incentive to develop technology to automatically process the information stored as natural language objects on the one hand, and to enable the machine to communicate using natural language skills on the other hand. Important applications which have been developed in the last decades of the twentieth century and are still subject to active research in 2001 include the following non-exhaustive list (see an extensive review in [28]):

- speech recognition,
- optical character and handwriting recognition,
- machine translation,
- document classification and routing,
- data mining,
- information extraction and retrieval from multimedia databases.

The reason why these tasks remain challenging is probably that the complexity of natural language prevents its mechanisms from being fully explained by a set of deterministic rules. To the contrary many methods involving an explicit stochastic modelling of natural language have been developed since the early 1980's and have been shown to outperform most deterministic approaches in almost all tasks we previously cited.

1.2 Stochastic language modelling

The general idea behind statistical approaches to natural language processing is that a stochastic process might be a good approximation to the process of generating a document in natural language. It does not mean that the real generating process by a human being is random by nature, but rather that this approximation provides a convenient way to represent complex objects and process them. In particular statistics gives a framework to capture regularities in natural language and include soft constraints in a decision making process.

1.2.1 Applications of statistical language models

To illustrate the power of stochastic models let us review several applications which currently heavily rely on stochastic process approximations to natural language. Here we suppose that one has been able to design a stochastic process on a finite alphabet $\mathcal{A}$ which "mimic" the behavior of the text generating process by human, where the alphabet $\mathcal{A}$ can be thought as the set of letters in the Latin alphabet, as the set of words in an English dictionary or as a set of ideograms to model Chinese or Japanese.

- **Text compression and transmission.** Information theory reveals that the knowledge of a process distribution can be used to design codes in order to compress and transmit data generated by the process. One of the first applications to stochastic modelling of natural language was therefore the compression and transmission over noisy channels of texts written in natural language, which is already presented in Shannon's pioneering book [78].

- **Bayes decision framework.** Many state-of-the-art methods in applications as different as speech recognition ([46]), optical character and handwriting recognition ([28]), or machine translation ([18], [19]), rely on the same paradigm: a "source-channel" model combined with a Bayesian decision. All these applications have an input which is modeled as a random variable I (which can be an acoustic signal, a digital image, or a sentence in a foreign language to translate) and output a sequence of words W. In the "source-channel" model one assumes that there exists a joint distribution $P(I,W)$ and follows the maximum a posteriori Bayesian rule to chose an output sequence W^* as follows:

$$\begin{aligned} W^* &= \arg\max_W P(W \mid I) \\ &= \arg\max_W \frac{P(W)P(I \mid W)}{P(I)} \\ &= \arg\max_W P(W)P(I \mid W) \quad . \end{aligned}$$

This decomposition shows that two statistical models play a role:

- the channel $P(I \mid W)$ which depends on the application considered,
- the a priori probability of strings $P(W)$ which is called the language model and is derived from the process distribution.

- **Bayes decision framework (bis).** In other applications including automatic document classification or information retrieval ([65], [11]) the same framework can be used with the language model playing the role of the likelihood instead of the prior. Consider for instance a document classification task, where every submitted document must be classified in one of k classes. Suppose that one has been able to design k language models $(P_1, \ldots, P_k)$ which correctly mimic the typical generating process corresponding to documents belonging to each class. Then the Bayesian classifier defines a prior $P(i)$ on the set of classes and assigns class k^* to a document W by maximizing the posterior probability of the class:

$$\begin{aligned} k^* &= \arg\max_{i=1,\ldots,k} P(i \mid W) \\ &= \arg\max_{i=1,\ldots,k} P(i)P_i(W) \quad . \end{aligned}$$

- **Text modelling.** Going further to the direction of local language models one can consider that if a stochastic language model P can be learned from a given object written in natural language then it can be used as a representation of that object. The object can be a category of texts as in the previous example, but can more generally be any text one want to process. Of course the text should be "long enough" for a local language model to be learned from it, but the emphasize here is not on the ability of the stochastic process to mimic natural language: it is rather on its ability to represent the initial text in a convenient and efficient way for further processing. Hence the stochastic modelling is used in that case as a way to represent a complex object. A typical application of this approach is the comparison of two texts: if each text is represented by a process distribution then the similarity between the two texts can be measured in terms of the distance between the two process distribution, where the distance is defined mathematically in the space of the process distribution.

1.2.2 Current language models and issues

A language model is a discrete-time process distribution on a finite alphabet. The most popular models up to now have been the so-called n-grams models, i.e. Markov models of order $n-1$. When the alphabet is a set of words to model English, for instance, n is usually set to 3, i.e. the probability of a word only depends on the two preceding words. Other more sophisticated models have been developed (see a survey in [70]) but barely outperform the basic n-grams models in most applications: they include decision tree models ([4]), stochastic context free grammars ([47]) or exponential models ([12]).

Among the difficulties met when building a language model one can mention the following.

- The statistical models involved have huge dimensions. A typical word dictionary contains at least 20,000 words, so the dimension of a basic trigram model is $20,000^3 = 8.10^{12}$...
- The number of observations is never infinite. In case one wants to build a local language model from a possibly small corpus (e.g. to model a text category) the situation is even worse than when the goal is to build a "general" language model.
- The language models have been experimentally shown to be highly context sensitive, which means that local language models outperform general language models for most applications.
- There exist many long distance correlations in language which can not be captured by local models like trigrams.

These difficulties can be summed up in one sentence: we would like to be able to learn from as few observations as possible (to obtain local models) models as complex as possible. One can observe that this paradox covers in fact two different situations:

- when the goal is to build a global language model the model complexity is usually fixed a priori (e.g. a trigram model) and data are collected until the models is considered as correctly trained, because the amount of data is assumed to be infinite in that case;

- when the goal is to build a local language model (e.g. for a category of text) then the amount of data is fixed a priori, and the complexity of the models should be adapted to it.

1.2.3 Motivations for this thesis

This thesis is an attempt to develop methods for building local language models and suggest applications where the model learned is used to represent the original object written in natural language. The goal of this thesis is therefore twofold.

- **Develop algorithms for efficient statistical estimation.** In view of the discussion in Sect. 1.2.2 we focus on the issue of estimating a possibly very complex process distribution from a finite number of observations. The main contribution of our work is to propose several methods and study their performance. The constraints we deal with make us work in a non-parametric framework (i.e. no strong assumption is made on the unknown distribution to estimate) and prove non-asymptotic results (i.e. we have in mind the question: given a finite number of observations, what can we learn from it?)
- **Use these estimators as representations of the original text for various applications.** As an application of the methods we develop we suggest a way of measuring the similarity between texts by the similarity between the corresponding estimated process distributions, and provide experimental evidence that such a similarity measure is meaningful for many applications.

1.3 Mathematical formulation

Let us formalize more precisely the issue we are confronted with. An finite alphabet $\mathcal{A}$ is given and our goal is to estimate a stationary process distribution on $\mathcal{A}$ from a "finite number of observations". A way to characterize a stationary process $T_{-\infty}^{\infty} = (T_i)_{i\in\mathbb{Z}}$ with distribution P is by its conditional distribution $P(T_0 \,|\, T_{-\infty}^{-1})$. In the reality however the observations we might get are never infinite so the problem is more pertinent if the length of the past is limited by an integer $D \in \mathbb{N}$. The issue is then to estimate a conditional probability distribution $P(T_0 \,|\, T_{-D}^{-1})$ from a finite number of observations.

1.3.1 Statistical framework

Let $\mathcal{X} = \mathcal{A}^D$ be the space of strings of length D to represent "the past letters" and $\mathcal{Y} = \mathcal{A}$ be the space of letters to represent "the next letter". Let $Z = (X, Y)$ be a random variable on a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ with values in $\mathcal{X} \times \mathcal{Y}$ and probability density P. Let $\mathcal{Q}$ be the set of conditional probability densities for Y conditionally to X, i.e.:

$$\forall (Q, x) \in \mathcal{Q} \times \mathcal{X}, \qquad \sum_{y \in \mathcal{Y}} Q(y \,|\, x) = 1 \ .$$

As a measure of similarity between the true unknown probability density P and any conditional density $Q \in \mathcal{Q}$ we use the conditional relative entropy or conditional Kullback-Leibler divergence ([29, p. 22]):

$$D^{(\mathrm{cond})}(P \,||\, Q) = \sum_{(x,y) \in \mathcal{X} \times \mathcal{Y}} P(x, y) \log \frac{P(y \,|\, x)}{Q(y \,|\, x)} \ .$$

This loss function is known to be non-negative and null only for $Q(y|x) = P(y|x)$ as soon as $P(x) > 0$. It plays a central role in information theory and is usually used to measure the pertinence of language models ([25]).

Let $n \in \mathbb{N}$ and $(Z_i)_{i=1}^n = (X_i, Y_i)_{i=1}^n$ be a series of independent and identically distributed random variables with common density P. An estimator $\hat{Q}$ for the conditional density $P(y \,|\, x)$ is a sequence of estimates

$$\left\{ \hat{Q}_n \,:\, (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{Q} \right\}_{n=0}^{\infty}$$

indexed by the number of observations used by the estimates. For any $(x, y) \in \mathcal{X} \times \mathcal{Y}$ we write $\hat{Q}_n(y \,|\, x; z_1^n)$ for the values of the conditional density obtained from z_1^n by the estimator $\hat{Q}_n$.

The goal of the estimator $\hat{Q}$ is to estimate $P(Y \,|\, X)$ from the observations Z_1^n. The risk of the estimator is defined as its expected loss, i.e.:

$$\begin{aligned} \mathcal{R}\left(P, n, \hat{Q}\right) &= \mathbf{E}_{P^{\otimes n}(dZ_1^n)} D^{(\mathrm{cond})}(P \,||\, \hat{Q}) \\ &= \sum_{z_1^n \in (\mathcal{X} \times \mathcal{Y})^n} P(z_1^n) \sum_{(x,y) \in \mathcal{X} \times \mathcal{Y}} \log \frac{P(y \,|\, x)}{\hat{Q}_n(y \,|\, x; z_1^n)} \ . \end{aligned}$$

The problem is now to define an estimator $\hat{Q}$ with small risk for "any P" and "any n". This formulation can result in several different problems in statistics, which we now briefly review in order to motivate our choice of designing "oracle estimators" (see Sect. 1.3.3).

1.3.2 Classical frameworks for density estimation

In the classical setting of statistical inference P is supposed to be unknown but to belong to some set $\{P_\theta, \theta \in \Theta\}$. Under this hypothesis the quality of an estimator has been widely studied from two points of view: the Bayesian and the minimax ones.

The Bayesian approach

In the Bayesian approach the set Θ is a measurable space on which a so-called prior probability distribution $w(d\theta)$ is chosen. The risk of an estimator $\hat{P}$, i.e. $\mathcal{R}(P, n, \hat{P})$, is averaged over Θ with respect to $w(d\theta)$ which results in the Bayesian risk:

$$\mathcal{R}^{(\text{Bayes})}(\Theta, w(d\theta), n, \hat{P}) = \mathbf{E}_{w(d\theta)} \mathcal{R}(P_\theta, n, \hat{P}) \ .$$

The Bayesian risk usually makes sense when one has a prior knowledge about the unknown density P, which is not the case in the situations we will consider in the sequel. Therefore we will not use the Bayesian risk as a measure of the performance of estimators, even though Bayesian analysis will be a source of inspiration for building estimators.

The minimax approach

In the minimax approach the performance of an estimator is measured in terms of the worst-case risk:

$$\mathcal{R}(\Theta, n, \hat{P}) = \sup_{\theta \in \Theta} \mathcal{R}(P_\theta, n, \hat{P}) \ .$$

The *minimax risk* is defined as:

$$\mathcal{R}(\Theta, n) = \inf_{\hat{P}} \mathcal{R}(\Theta, n, \hat{P}) \ ,$$

where the minimization is over all estimators. The minimax risk approach is interesting only when the minimax risk tends to zero as the number of observations tends to infinity, in which case the rate of convergence to zero is characteristic of the class $\{P_\theta, \theta \in \Theta\}$. It is usually hopeless to find an estimator $\hat{P}$ whose worst-case risk is exactly the minimax risk for every number of observations n, and optimality is measured asymptotically by one of the following notions ([60]):

- An estimator P is said to be *asymptotically efficient* if

$$\mathcal{R}(\Theta, n, \hat{P}) = (1 + o(1))\, \mathcal{R}(\Theta, n) \,, \quad n \to \infty \ .$$

- An estimator P is said to be *optimal in order* if

$$\mathcal{R}(\Theta, n, \hat{P}) = O(1) \mathcal{R}(\Theta, n) \,, \quad n \to \infty \ .$$

This minimax formulation only makes sense if P belongs to $\{P_\theta, \theta \in \Theta\}$ because in that case the risk of an estimator $\hat{P}$ is upper bound by the following straightforward inequality:

$$\forall P \in \{P_\theta, \theta \in \Theta\}, \qquad \mathcal{R}(P, n, \hat{P}) \leq \mathcal{R}(\Theta, n, \hat{P}) \ .$$

Minimax rates of convergence have been extensively studied for various loss functions and functions classes. It is for instance known that the minimax rate of convergence for density and regression estimation is related to the metric entropy of the the parameter space in several general frameworks. The theory of minimax estimation being too rich to be summarized in this introduction we refer the interested reader to classical references on the topic including (but not limited to) [81], [13], [61], [85], [14], [8], [60].

An obvious issue with the minimax approach is that the fewer assumptions are made on P the larger the minimax risk. This worst-case philosophy is particularly objectionable when the class of possible P is a union of several classes with various complexities. i.e. various minimax risk, because in that case one could hope that densities belonging to low-complexity classes could be better approximated than densities belonging to large-complexity classes. This is the issue addressed by adaptive estimators as follows.

Adaptive minimax approach

Suppose that the family $\{P_\theta, \theta \in \Theta\}$ is a union of smaller families of different regularities, i.e.:

$$\Theta = \bigcup_{i \in I} \Theta_i \ ,$$

where I is a countable set. The complexity of a particular set Θ_i is represented by its associated minimax risk $\mathcal{R}(\Theta_i, n)$ for any $n \in \mathbb{N}$.

The goal of *adaptive* estimation is to construct a single estimator $\hat{P}$ which is simultaneously optimal in order on each class Θ_i, i.e., that the following holds:

$$\forall i \in I, \qquad \mathcal{R}(\Theta_i, n, \hat{P}) = O(1)\mathcal{R}(\Theta_i, n) \,, \quad n \to \infty \ . \tag{1.1}$$

A lot of attention has been devoted to the definition and the analysis of adaptive estimators in the recent years. One can cite adaptive function estimators for ellipsoid classes ([40] ,[39]), for some Lipschitz classes using adaptive kernel estimators ([45]) or for Besov classes using wavelet analysis ([36], [37]). Adaptation schemes by model selection using penalized maximum likelihood has been deeply studied in [15] and [6], and general adaptation risk bounds for density estimation based on minimum description length (MDL) criterion have been derived in [9].

Adaptive estimators enable the statistician to build a family of models $(\Theta_i)_{i\in I}$ which he thinks should contain the true unknown object P to be estimated and still obtain a risk of the same order as if he knew the precise class Θ_i that contains P. Thus adaptation refers to the ability of an estimator to be minimax optimal for every class Θ_i. As a result this approach is pertinent only if P belongs to $\{P_\theta, \theta \in \Theta_i\}$ for some $i \in I$ and gives an asymptotic result, i.e. the asymptotic risk of the estimator with respect to P is of order of the minimax risk on Θ_i. This approach is not satisfactory for our problem characterized by the facts that no assumption should be made on P and that we are interesting in non-asymptotic risk bounds.

1.3.3 Removing assumptions on P

In order to work with as few assumptions on P as possible we need to reverse our point of view and consider universality *with respect to a class of estimators* and not to a class of functions to estimate. In other words the question addressed becomes the following: given a set of models, is it possible to build an estimator which performs almost as well as the best density in the best model? As for the classical setting this question can be declined for a single model as well as for a family of models as follows.

Universal minimax approach

Let us call $\mathcal{P}$ the set of probability densities on $\mathcal{X}\times\mathcal{Y}$. In the general setting P is just assumed to belong to $\mathcal{P}$. Suppose now that a family of conditional densities $\{P_\theta, \theta \in \Theta\}$ is given, and consider the set of estimators $\hat{Q}$ with values in that family, i.e.:

$$\forall n \in \mathbb{N}, \forall z_1^n \in (\mathcal{X} \times \mathcal{Y})^n , \qquad \hat{Q}_n(.\,|\,.; z_1^n) \in \{P_\theta, \theta \in \Theta\} \ .$$

For general $P \in \mathcal{P}$ the risk of such an estimator $\hat{Q}$ might not converge to zero because it is lower bounded by the loss of the best approximation of P in the family:

$$\forall (P, n) \in \mathcal{P} \times \mathbb{N}, \qquad \mathcal{R}\left(P, n, \hat{Q}\right) \geq \inf_{\theta\in\Theta} \mathcal{R}\left(P, n, P_\theta\right) \ ,$$

where $\mathcal{R}\left(P, n, P_\theta\right) = D^{(\mathrm{cond})}(P\,||\,P_\theta)$. Even though $\hat{Q}$ can not estimate P precisely a pertinent question is the following: is it possible to design an estimator $\hat{Q}$ which approximates well the *projection* of any P on the family $\{P_\theta, \theta \in \Theta\}$? This question leads to the following definition of the *universal risk* of an estimator:

$$\mathcal{R}^{(\mathrm{universal})}\left(\mathcal{P}, n, \hat{Q}, \Theta\right) = \sup_{P\in\mathcal{P}} \left[\mathcal{R}\left(P, n, \hat{Q}\right) - \inf_{\theta\in\Theta} \mathcal{R}\left(P, n, P_\theta\right)\right] \ ,$$

and to the following definition of a *universal minimax risk*:

$$\mathcal{R}^{(\mathrm{universal})}\left(\mathcal{P}, n, \Theta\right) = \inf_{\hat{Q}} \sup_{P\in\mathcal{P}} \left[\mathcal{R}\left(P, n, \hat{Q}\right) - \inf_{\theta\in\Theta} \mathcal{R}\left(P, n, P_\theta\right)\right] \ ,$$

where the minimization is over all estimators $\hat{Q}$ with values in $\{P_\theta, \theta \in \Theta\}$. Like in the classical minimax setting we are interesting in estimators $\hat{Q}$ whose universal risk is asymptotically comparable to the universal minimax risk, i.e.:

$$\mathcal{R}^{(\text{universal})}\left(\mathcal{P}, n, \hat{Q}, \Theta\right) = \gamma(n)\mathcal{R}^{(\text{universal})}\left(\mathcal{P}, n, \Theta\right) \quad ,$$

where $\gamma(n)$ is of order $1 + o(1)$ or $O(1)$ in which case we respectively say that $\hat{Q}$ is asymptotically efficient or optimal in order for the universal minimax risk.

Oracle estimator

Let us now present the oracle point of view, which has been introduced by Donoho and Johnstone in the context of wavelet approximations ([35], [36], [37]) and which is the point of view we will adopt in this thesis.

Suppose that a family of models is available, i.e.

$$\Theta = \bigcup_{i \in I} \Theta_i \quad ,$$

where I is a countable set. When P is estimated by an estimator $\hat{Q}_i$ with value in $\{P_\theta, \theta \in \Theta_i\}$ for some $i \in I$ then the best possible uniform bound for the risk is:

$$\forall P \in \mathcal{P}, \qquad \mathcal{R}\left(P, n, \hat{Q}_i\right) \leq \inf_{\theta \in \Theta_i} \mathcal{R}\left(P, n, P_\theta\right) + \mathcal{R}^{(\text{universal})}\left(\mathcal{P}, n, \Theta_i\right) \quad .$$

For any $n \in \mathbb{N}$ the best risk upper bound which could be obtained using a family of estimators $\{\hat{Q}_i\}_{i \in I}$ is then:

$$\mathcal{R}^{(\text{oracle})}\left(P, n, \{\hat{Q}_i\}_{i \in I}\right) = \inf_{i \in I} \mathcal{R}\left(P, n, \hat{Q}_i\right) \quad .$$

This risk is called the *oracle* risk because it would require an oracle to decide which $i \in I$ achieves the smallest risk bound for every $P \in \mathcal{P}$ and $n \in \mathbb{N}$. Even though oracle usually don't exist in nature it is sometimes possible to design estimators $\hat{Q}$ which satisfy an oracle inequality of the following type:

$$\forall P \in \mathcal{P}, \qquad \mathcal{R}\left(P, n, \hat{Q}\right) \leq \inf_{i \in I}\left[\inf_{\theta \in \Theta_i} \mathcal{R}\left(P, n, P_\theta\right) + O(1)\mathcal{R}^{(\text{universal})}\left(\mathcal{P}, n, \Theta_i\right)\right] \quad . \tag{1.2}$$

Oracle inequalities have been obtained in relation with wavelet approximations ([36], [37]) and more recently general methods have been developed to build oracle estimators ([23], [22], [24], [95]).

Advantages of oracle estimators

An estimator $\hat{Q}$ which satisfies the oracle inequality (1.2) has the following interesting properties:

- No assumption is made on P, which is approximated by its projections on the various models designed by the statistician.

- For any $i \in I$, $\hat{Q}$ approximates P almost "as well" (i.e., up to the $O(1)$ term) as the best estimator $\hat{Q}_i$ of the projection of P on $\{P_\theta, \theta \in \Theta_i\}$. In particular the risk $\mathcal{R}\left(P, n, \hat{Q}\right)$ converges to the best possible risk:

$$\inf_{i \in I} \inf_{\theta \in \Theta_i} D^{(\text{cond})}(P \,||\, P_\theta) \ ,$$

  at the best possible rate up to a multiplicative constant.

- For any $n \in \mathbb{N}$ and $i \in I$ the risk $\mathcal{R}(P, n, \hat{Q}_i)$ of an asymptotically efficient estimator $\hat{Q}_i$ can be seen as the sum of a *bias* term $\inf_{\theta \in \Theta_i} \mathcal{R}(P, n, P_\theta)$ and an estimation risk $O(1)\mathcal{R}^{(\text{universal})}(\mathcal{P}, n, \Theta_i)$ which quantifies the difficulty of estimating a density in $\{P_\theta, \theta \in \Theta_i\}$. The oracle inequality (1.2) shows that the estimator $\hat{Q}$ achieves a trade-off for any $n \in \mathbb{N}$ between the bias term and the estimation risk for every model. Therefore its behavior can also be considered "almost optimal" for any finite number of observations.

These properties clearly fit and quantify the constraints of our problem at hand, namely that the estimator $\hat{Q}$ should approximate "well" any $P \in \mathcal{P}$ for any $n \in \mathbb{N}$. Therefore the main efforts of this thesis concentrate on the problem of designing and studying estimators for an unknown conditional probability density which satisfy an oracle inequality for some family Θ.

1.3.4 Link with universal coding

A central question which arises in information theory is the following: given an unknown process distribution P, how to approximate its marginals on blocks of length n under a Kullback-Leibler loss criterion? An estimator is a process distribution $\hat{P}$ whose risk is defined by:

$$\mathcal{R}^{(\text{compression})}(P, n, \hat{P}) = \sum_{t_1^n \in \mathcal{A}^n} P(t_1^n) \log \frac{P(t_1^n)}{\hat{P}(t_1^n)} \ .$$

This risk is also called *redundancy* and quantifies the loss in average code length when the probability $\hat{P}$ is used instead of P to encode messages of length n from the source P. The Bayesian and minimax risks with respect to a family $\{P_\theta, \theta \in \Theta\}$ are defined in a similar way as for the density estimation problem. The redundancy-capacity theorem of universal coding ([42], [32], [44]) identifies the minimax risk as a channel capacity which measures the richness of the class $\{P_\theta, \theta \in \Theta\}$.

As pointed out by many authors density estimation is closely related to universal coding ([5], [26], [9], [93], [43]). It would be beyond the scope of this introduction to develop this relationship, so let us just mentions three remarks about information theoretic approaches which were a source of inspiration for our work.

- Hierarchical universal codes (see a review in [58]) are equivalent to adaptive estimators in density estimation. Two general approaches have been investigated to design such codes: the first one consists in selecting one model by minimizing the total description length of the data ([71]) and the other one consists in building a Bayes mixtures of all models ([72], [41], [91]). Transposed in the framework of density estimation the latter approach has been investigated recently and resulted in several general techniques for model aggregation as opposed to model selection ([23], [24], [22], [95]). Our work further investigates these techniques of building mixture estimators which satisfy oracle inequalities.

- The context tree weighting algorithm ([91]) suggests an efficient implementation for estimating Markov models based on context trees, which can be adapted to conditional density estimation.

- Recoding provides a way to represent the design variable X more efficiently, by shortening the average number of bits used to represent it. As a result models with smaller dimension can mimic P after this recoding (see Part 5).

1.3.5 Choice of the alphabet

At this point it is time to discuss the choice of the alphabet $\mathcal{A}$ used to define the natural language generating process. Depending on the language considered several "natural" choices might exist: letters, words or "tokens" for languages like English of French, single ideograms or words for Chinese or Japanese, etc...

Let us focus on English, which is the language used in the experiments we carry out in this thesis. Language models are overwhelmingly based on words or word tokens dictionary, because of the underlying semantic content of words. However the size of such dictionaries is typically of the order of several tenths of thousands of entries, which make any density estimation from a "small" number of observations (typically a newspaper article) very tricky from a statistical point of view.

For this reason we decided to work with a much smaller alphabet, composed of the 26 letters of the Latin alphabet and a supplementary character for spaces, punctuation, numbers etc... We discuss in more detail this choice in Part. 4 and provide experimental evidences that a letter-based process modelling can capture semantical information as well as a word-based process modelling.

From a mathematical point of view one can ask the following question: what is lost when one decides to work at the letter level instead of the word level in terms of discrimination power between different processes? More formally, let $\mathcal{A}$ be the letter alphabet, ϵ be the supplementary character to separate words and $\mathcal{S} \subset \{\epsilon\} \times \mathcal{A}^*$ be a finite dictionary, where $\mathcal{A}^* = \bigcup_{n=0}^{\infty} \mathcal{A}^n$ is the set of finite strings made of letters (hence a word can be written as $w = \epsilon\alpha_1 \dots \alpha_{l-1}$, where $\alpha_i \in \mathcal{A}$ for $i = 1, \dots, l$, and $l = l(w)$ is the length of the word w). Let P and Q be two stationary processes distributions on $\mathcal{S}^{\mathbb{Z}}$, which we call *word processes*. As stated earlier the difference between the two process P and Q is measured in terms of their relative conditional entropy, i.e.:

$$D\left(P(dW_1 \,|\, W_{-\infty}^0) \,||\, Q(dW_1 \,|\, W_{-\infty}^0)\right) = \int_{W_{-\infty}^1 \in \mathcal{S}^{-\mathbb{N}} \times \mathcal{S}} P(dW_{-\infty}^1) \log \frac{dP}{dQ}(dW_1 \,|\, W_{-\infty}^0)$$

if $dP/dQ(dW_1 \,|\, W_{-\infty}^0)$ exists, $+\infty$ otherwise.

These stationary process distributions P and Q on $\mathcal{S}^{\mathbb{Z}}$ can be mapped to stationary process distributions Π_P and Π_Q on $(\mathcal{A} \cup \{\epsilon\})^{\mathbb{Z}}$ which we call *letter processes* and whose sample paths are infinite concatenations of words (see [79, p. 103] for the construction of Π_P by the cutting and stacking method). The corresponding measure of similarity between the processes Π_P and Π_Q is:

$$\begin{aligned} D\left(\Pi_P(dX_1 \,|\, X_{-\infty}^0) \,||\, \Pi_Q(dX_1 \,|\, X_{-\infty}^0)\right) \\ = \int_{X_{-\infty}^1 \in (\mathcal{A}\cup\{\epsilon\})^{-\mathbb{N}} \times (\mathcal{A}\cup\{\epsilon\})} \Pi_P(dX_{-\infty}^1) \log \frac{d\Pi_P}{d\Pi_Q}(dX_1 \,|\, X_{-\infty}^0) \ , \end{aligned} \tag{1.3}$$

if $d\Pi_P/d\Pi_Q(dW_1 \,|\, W_{-\infty}^0)$ exists, $+\infty$ otherwise.

The following lemma (whose proof is postponed to Sect. 1.5) shows that the effect of working with the letter processes Π_P and Π_Q instead of working with the word processes P and Q is to divide the conditional relative entropy by the average length of the words.

Lemma 1

$$D\left(\Pi_P(dX_1 \,|\, X_{-\infty}^0) \,||\, \Pi_Q(dX_1 \,|\, X_{-\infty}^0)\right)$$

$$= \frac{1}{\mathbf{E}_{P(dW_1)} l(W_1)} D\left(P(dW_1 \mid W_{-\infty}^0) \,||\, Q(dW_1 \mid W_{-\infty}^0)\right).$$

As a result the "topology" generated by the conditional relative entropy on the space of stationary word processes is conserved when these processes are considered as stationary letter processes. Suppose for instance that $\left(P_i(dW_1 \mid W_{-\infty}^0)\right)_{i=1,\dots,k}$ are k given conditional distributions and one wants to label any new word process distribution Q by choosing the label $i \in \{1, \dots, k\}$ for which

$$D\left(Q\left(dW_1 \mid W_{-\infty}^0\right) \,||\, P_i\left(dW_1 \mid W_{-\infty}^0\right)\right) \tag{1.4}$$

is the smallest (see Part. 4). Then Lemma 1 shows that this choice can be based on the comparison of the conditional relative entropies of the corresponding letter processes as well.

1.4 Contribution of this thesis

The remaining four parts of this thesis are made of published or submitted research articles. In spite of their obvious lack of unity let us try to highlight their coherence with respect to the general problem we presented in this introduction.

1.4.1 Design and study of oracle estimators

In view of the discussion in Sect. 1.3 we decided to investigate the possibility of designing, studying and implementing estimators for a conditional density which satisfy oracle inequalities like (1.2). In this framework we suppose that one is given a set of i.i.d. observations Z_1^N which should be used to build the estimator.

Context tree models

A family of models to approximate $P(Y \mid X)$ is defined in Part 2. For each model the variable X only depends on Y through one particular suffix as encoded in a context tree. These model generalize the models used in [91] to build the context tree weighting algorithms. This family is used in the other Parts as well.

Estimation by splitting the observations

In Part 2 we define several estimators which partition the observations Z_1^N into two groups. The observations of the first group are used to build a set of estimators $\{\hat{Q}_i\}_{i\in I}$ for the projections of P on every set $\{P_\theta, \theta \in \Theta_i\}$, and the observations of the second group are used to "aggregate" the estimates $\{\hat{Q}_i\}_{i\in I}$, i.e. to build estimators for which we prove oracle inequalities.

Estimation without splitting

In Part 3 we show how to estimate the projections on the different models and aggregate the models without partitioning the observations into two groups, i.e. by using the same data to estimate continuous parameters and discrete models in the same time. The estimator we define is proven to satisfy an oracle inequality, and we discuss the difference between this estimators and "twice-universal" estimators used in coding theory due to the difference in the criterion to optimize.

1.4.2 Experimental results

An experimental protocol is defined in Part 2 and Part 4 to draw i.i.d. samples from natural language texts and use the oracle estimators to define a measure of similarity between texts. This measure is used in Part 2 to carry out an unsupervised text clustering experiment which demonstrate the capacity of the estimator to recognize from which book a text is extracted, and in Part 4 to define a method to automatically classify texts into predefined categories.

1.4.3 Change of representation

In Part 5 we consider the problem of changing the representation of the past string to predict the next letter. In particular we present an iterative algorithm where the estimation of the process distribution in each iteration is used to recode the data in the next iteration, in order to concentrate more information in the D letters the past is made of.

1.4.4 Sampling from a Markov chain

In Part 5 we also consider the problem of sampling data from a realization of a Markov chain. Indeed i.i.d. observations are usually obtained by sampling random positions in a given text, which can be seen as one realization of the unknown Markov process we try to estimate. In the case the data are sampled from the empirical distribution of the process (i.e. form a bootstrap sample) we prove an oracle-like inequality for the density estimator using a measure concentration result for the empirical measure of a Markov chain.

1.5 Annex: proof of Lemma 1

For any two strings $(u,v) \in ((\mathcal{A} \cup \{\epsilon\})^*)^2$ we say that u is a *prefix* of v and write $u \prec v$ if there exists a string $w \in (\mathcal{A} \cup \{\epsilon\})^*$ such that $v = uw$. Let $\mathcal{S}_-$ be the set of prefixes of the elements of the dictionary $\mathcal{S}$, i.e.:

$$\mathcal{S}_- = \{u \in (\mathcal{A} \cup \{\epsilon\})^* \,:\, \exists v \in \mathcal{S}\,,\, u \prec v\} \ .$$

By definition of the conditional relative entropy we can write:

$$\begin{aligned}
& D\left(\Pi_P(dX_1 \,|\, X_{-\infty}^0) \,||\, \Pi_Q(dX_1 \,|\, X_{-\infty}^0)\right) \\
&= \int_{x_{-\infty}^0} \Pi_P(x_{-\infty}^0) \sum_{x_1 \in \mathcal{A}} \Pi_P(x_1 \,|\, x_{-\infty}^0) \ln \frac{\Pi_P(x_1 \,|\, x_{-\infty}^0)}{\Pi_Q(x_1 \,|\, x_{-\infty}^0)} \\
&= \int_{w_{-\infty}^0} \sum_{s \in S_-} \Pi_P(w_{-\infty}^0 s) \sum_{x_1 \in \mathcal{A}} \Pi_P(x_1 \,|\, w_{-\infty}^0 s) \ln \frac{\Pi_P(x_1 \,|\, w_{-\infty}^0 s)}{\Pi_Q(x_1 \,|\, w_{-\infty}^0 s)} \ ,
\end{aligned}$$

where the last inequality is due to the fact that the mapping $x_{-\infty}^0 \mapsto w_{-\infty}^0 s$ is a bijection from the support of Π_P to $\mathcal{S}^{-\mathbb{N}} \times \mathcal{S}_-$. Let us introduce the notation:

$$\forall (s, w_{-\infty}^0) \in \mathcal{S}_- \times \mathcal{S}^{\mathbb{N}}, \qquad p_s(w_{-\infty}^0) = \sum_{s \prec w, w \in \mathcal{S}} P(w | w_{-\infty}^0) \ .$$

and define $q_s(w_{-\infty}^0)$ the same way by replacing P by Q. By definition of the concatenated-block process the followings hold:

$$\begin{aligned}
\Pi_P(w_{-\infty}^0 s) &= \sum_{s \prec w_1} \left(\frac{l(w_1) P(w_1)}{\mathbf{E}_{P(dW_1)} l(W_1)} \times \frac{1}{l(w_1)} \frac{P(w_{-\infty}^1)}{P(w_1)} \right) \\
&= \frac{1}{\mathbf{E}_{P(dW_1)} l(W_1)} \times P(w_{-\infty}^0) p_s(w_{-\infty}^0) \ ,
\end{aligned}$$

and

$$\Pi_P(x_1 \,|\, w_{-\infty}^0 s) = \begin{cases} \dfrac{p_{sx_1}\left(w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)} & \text{if } x_1 \neq \alpha \ , \\ \dfrac{P\left(s \,|\, w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)} & \text{if } x_1 = \alpha \ . \end{cases}$$

We can now resume the computation:

$$
\begin{aligned}
&D\left(\Pi_P(dX_1 \,|\, X_\infty^0) \,||\, \Pi_Q(dX_1 \,|\, X_\infty^0)\right) \\
&= \int_{w_{-\infty}^0} \frac{P\left(w_{-\infty}^0\right)}{\mathbf{E}_{P(dW_1)} l(W_1)} \sum_{s \in \mathcal{S}_-} p_s(w_{-\infty}^0) \left(\frac{P\left(s \,|\, w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)} \ln \frac{\dfrac{P\left(s \,|\, w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)}}{\dfrac{Q\left(s \,|\, w_{-\infty}^0\right)}{q_s\left(w_{-\infty}^0\right)}} \right. \\
&\qquad\qquad \left. + \sum_{x \in \mathcal{A}, x \neq \alpha} \frac{p_{sx}\left(w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)} \ln \frac{\dfrac{p_{sx}\left(w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)}}{\dfrac{p_{sx}\left(w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)}} \right) \\
&= \int_{w_{-\infty}^0} \frac{P\left(w_{-\infty}^0\right)}{\mathbf{E}_{P(dW_1)} l(W_1)} \sum_{s \in \mathcal{S}_-} \left(P\left(s \,|\, w_{-\infty}^0\right) \ln \frac{\dfrac{P\left(s \,|\, w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)}}{\dfrac{Q\left(s \,|\, w_{-\infty}^0\right)}{q_s\left(w_{-\infty}^0\right)}} \right. \\
&\qquad\qquad \left. + \sum_{x \in \mathcal{A}, x \neq \alpha} p_{sx}\left(w_{-\infty}^0\right) \ln \frac{\dfrac{p_{sx}\left(w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)}}{\dfrac{p_{sx}\left(w_{-\infty}^0\right)}{p_s\left(w_{-\infty}^0\right)}} \right) \\
&= \int_{w_{-\infty}^0} \frac{P\left(w_{-\infty}^0\right)}{\mathbf{E}_{P(dW_1)} l(W_1)} \sum_{s \in \mathcal{S}_-} \left(P\left(s \,|\, w_{-\infty}^0\right) \ln \frac{P\left(s \,|\, w_{-\infty}^0\right)}{Q\left(s \,|\, w_{-\infty}^0\right)} \right. \\
&\qquad\qquad \left. - p_s\left(w_{-\infty}^0\right) \ln \frac{p_s\left(w_{-\infty}^0\right)}{q_s\left(w_{-\infty}^0\right)} + \sum_{x \in \mathcal{A}, x \neq \alpha} p_{sx}\left(w_{-\infty}^0\right) \ln \frac{p_{sx}\left(w_{-\infty}^0\right)}{p_{sx}\left(w_{-\infty}^0\right)} \right) \\
&= \int_{w_{-\infty}^0} \frac{P\left(w_{-\infty}^0\right)}{\mathbf{E}_{P(dW_1)} l(W_1)} \sum_{w_1 \in \mathcal{S}} P\left(w_1 \,|\, w_{-\infty}^0\right) \ln \frac{P\left(w_1 \,|\, w_{-\infty}^0\right)}{Q\left(w_1 \,|\, w_{-\infty}^0\right)} \ ,
\end{aligned}
$$

where the third equality results from the fact that

$$
\forall (s, w_{-\infty}^0) \in \mathcal{S}_- \times \mathcal{S}^{\mathbb{N}} \qquad p_s\left(w_{-\infty}^0\right) = P\left(s \,|\, w_{-\infty}^0\right) + \sum_{x \in \mathcal{A}, x \neq \alpha} p_{sx}\left(w_{-\infty}^0\right).
$$

This finishes the proof of Lemma 1. □

Chapter 2

Adaptive context trees

Abstract

In the finite alphabet context we propose four alternatives to fixed-order Markov models to estimate a conditional distribution. They consist in working with a large class of variable-length Markov models represented by context trees, and building an estimator of the conditional distribution with a risk of the same order as the risk of the best estimator for every model simultaneously, in a conditional Kullback-Leibler sense.

Such estimators can be used to model complex objects like texts written in natural language and define a notion of similarity between them. This idea is illustrated by experimental results of unsupervised text clustering.

2.1 Introduction

Consider the problem of measuring the similarity between two long strings in the finite alphabet context, e.g. two English texts or two DNA sequences. A possible approach to cope with the impossibility of comparing them directly consists in replacing the initial strings by *representations* easier to handle and compare. For this purpose finite order Markov models are widely used to catch statistical information from the initial strings and represent them. A trivial example is the so-called *vector-space model* introduced by Salton et al. [74] for indexing texts by the statistical distribution of words they contain, which can be seen as a 0-order Markov model. Larger order models appear for language models, e.g. in speech or optical character recognition systems (see a survey in [28]),

The order of any Markov model is usually limited because the number of parameters to estimate increases exponentially with it, while the initial strings have finite length. On the other hand these strings are supposed to have long-range correlations, which might be better caught by models of high order.

Our contribution in this paper is to present and study several alternatives to fixed order Markov models, and show through an experiment of unsupervised text clustering how to use our results to measure similarities between English texts. More precisely we consider a larger class $\mathcal{M}$ of Markov models in which the conditional distribution of the next symbol depends on a variable number of preceding symbols. Hence a particular model $m \in \mathcal{M}$ is a parametric family of conditional distributions $\left\{P_{\theta_m}, \theta_m \in \Theta_m \subset \mathbb{R}^{d(m)}\right\}$. Such models are interesting because they can catch long-range dependencies on some particular strings without having necessarily an exponentially growing number of parameters. However it is unknown *a priori* which model to use when confronted with a given text or DNA sequence: we show in the sequel how to use "aggregation rules" among models, i.e. methods of combining several models as opposed to selecting a particular one, to build an estimator $\hat{P}$ whose risk approaches the risk of the best conditional density in the family of models considered (Theorems 4 and 6), in the sense that:

$$R_P(\hat{P}) \leq \inf_{m \in \mathcal{M}, \theta \in \Theta_m} \left\{ R_P(P_{\theta_m}) + \frac{c_N(m)}{N} \right\} \quad , \tag{2.1}$$

where R_P denotes the distance of a conditional density with the true unknown density P in a Kullback-Leibler sense (see equation (2.2)), and $c_N(m)/N$ should be as close as possible as the minimax risk for the model m. The bound (2.1) is *universal* because it is obtained without restrictive hypotheses on P, in particular P is not required to belong to any model m. Yet if it does it can be approximated at the minimax rate in the model considered (with a loss in the constant), as if this information was known a priori : in such a case we say the estimator is *adaptive*.

There are many connections between our results and universal coding as defined by Davisson [30], which consists in building a probability on the set of strings of length N that approximates simultaneously every probability of a predefined set as N increases, in the Kullback-Leibler distance sense. The literature about universal codes is very rich, and many authors have proposed solutions to problem (2.1) in that case with $1/N$ being replaced by $\log N$ (including Rissanen and Langdon [69], Davisson [31], Ryabko [71], Willems et al. [91], Feder and Merhav [41] and Barron et al. [7]). The link with our concern in this paper is that the redundancy criterion of universal coding is the sum of the expected distances we consider for string sizes growing from 1 to N. In spite of this, results are difficult to adapt because a control of the Cesaro mean of a sequence does not always lead to a control of the sequence itself : We overcome this issue of *universal prediction* by using statistical aggregation methods.

This paper is organized as follows. After setting up the statistical framework and presenting the family of Markov models in Section 2.2, we study two estimators for the parameters of a single model in Section 2.3 and prove universal bounds on their risk. In Section 2.4 we build a probability on the family of Markov models defined earlier, and propose two aggregation methods in Sections 2.5 and 2.6 with universal bounds. Each of these two methods can be used to aggregate each of both estimators studied for a given model, therefore resulting in four possible global estimators. In Section 2.7, we show how using a data-dependent prior on the models improves the estimators, and in Section 2.8 we propose an efficient implementation in the spirit of the Context Tree Weighting algorithm ([91]). Finally Section 2.9 is devoted to presenting some experimental results : The estimators studied in the paper are used to represent texts written in natural language, and an unsupervised text clustering experiment based on this representation is carried out.

The implementation of one of them is discussed in Section 2.8.

2.2 Definitions and framework

Let $a \in \mathbb{N}^*$ be an integer fixed throughout this paper. Consider an *alphabet*, $\mathcal{A} = \{1, .., a\}$ with size $|\mathcal{A}| = a$ and whose elements are called *letters*. A *string* s is a finite concatenation of letters which can be written as $s = q_{1-l}q_{2-l}...q_0$ with $q_{-i} \in \mathcal{A}$ for $i = 0, 1, ..., l-1$. l is called the *length* of the string s and written $l(s)$. The empty string λ has length $l(\lambda) = 0$. The set of all strings is $\mathcal{A}^* = \bigcup_{i=0}^{\infty} \mathcal{A}^i$. The concatenation of two strings s and s' is written ss'. We say that a string $s = q_{1-l}q_{2-l}...q_0$ is a *suffix* of the string $s' = q'_{1-l}q'_{2-l}...q'_0$ if $l \leq l'$ and $q_{-i} = q'_{-i}$ for $i = 0, ..., l-1$. The empty string λ is a suffix of all strings.

For any random variable X on a finite space $\mathcal{X}$ with probability distribution P we use the notation $P(x) = Pr\{X = x\}$. The expectation of a measurable function $f : \mathcal{X} \to \mathbb{R}$ with respect to P is denoted by $\mathbf{E}_{P(dX)} f(X)$ or $\mathbf{E}_P f(X)$ if there is no ambiguity.

2.2.1 Statistical framework

Let D be an integer, fixed throughout this paper. We consider the measurable product space $(\mathcal{X} \times \mathcal{Y}, \mathcal{B}_1 \otimes \mathcal{B}_2)$, where $\mathcal{Y} = \mathcal{A}$, $\mathcal{X} = \mathcal{A}^D$, and $\mathcal{B}_1$ and $\mathcal{B}_2$ are the discrete sigma algebras on $\mathcal{X}$ and $\mathcal{Y}$. We address in this paper the issue of estimating the conditional distribution of a letter $Y \in \mathcal{Y}$ given a string $X \in \mathcal{X}$ based on a series of observations. In order to model the random nature of X and Y we suppose that a family of unknown probability distributions is given :

$$\forall N \in \mathbb{N} \qquad P_N \in \mathcal{M}_+^1\left((\mathcal{X} \times \mathcal{Y})^N, (\mathcal{B}_1 \otimes \mathcal{B}_2)^{\otimes N}\right) ,$$

and we let $\{(X_i, Y_i) = Z_i; i = 1, \ldots, N\}$ be the canonical process.

One can for instance think of P_N as $P^{\otimes N}$, with P being a probability on $(\mathcal{X} \times \mathcal{Y}, \mathcal{B}_1 \otimes \mathcal{B}_2)$, if the observations are supposed to be independent and identically distributed. However we will only use the weaker assumption that P_N is *exchangeable*, i.e. that for any permutation σ of $\{1, \ldots, N\}$ and any $A \in (\mathcal{B}_1 \otimes \mathcal{B}_2)^N$,

$$P_N\left(Z_1^N \in A\right) = P_N\left((\sigma Z)_1^N \in A\right) ,$$

where σZ is the exchanged process

$$(\sigma Z)_i = Z_{\sigma(i)}, \qquad i = 1, \ldots, N .$$

An estimator $\hat{P}_N$ for the conditional probability of Y_N knowing X_N maps any observation (z_1^{N-1}, x_N) to a probability distribution $\hat{P}_N(.\,|\, z_1^{N-1}, x_N)$ on $\mathcal{Y}$. The performance of an estimator is measured in terms of the Kullback-Leibler divergence $D(.||.)$ as follows :

$$\begin{aligned} r_{P_N,\hat{P}_N}(z_1^{N-1}, x_N) &= D\left(P_N(.\,|\, z_1^{N-1}, x_N) || \hat{P}_N(.\,|\, z_1^{N-1}, x_N)\right) \\ &= \sum_{y_N \in \mathcal{Y}} P_N(y_N \,|\, z_1^{N-1}, x_N) \log \frac{P_N(y_N \,|\, z_1^{N-1}, x_N)}{\hat{P}_N(y_N \,|\, z_1^{N-1}, x_N)} . \end{aligned}$$

The observation itself having a random nature the performance of the estimator is judged according to its expected divergence, which we call the *risk* of the estimator $\hat{P}_N$:

$$\begin{aligned} R_{P_N}\left(\hat{P}_N\right) &= \mathbf{E}_{P_N}\left(r_{P_N,\hat{P}_N}(Z_1^{N-1}, X_N)\right) \\ &= \sum_{z_1^N \in (\mathcal{X} \times \mathcal{Y})^N} P_N(z_1^N) \log \frac{P_N(y_N \,|\, z_1^{N-1}, x_N)}{\hat{P}_N(y_N \,|\, z_1^{N-1}, x_N)} . \end{aligned} \tag{2.2}$$

This risk is the *conditional Kullback-Leibler divergence* (also called *conditional relative entropy*, see e.g. [29, p. 22]) and plays a central role in universal coding and prediction (see a survey in [58]).

2.2.2 Tree models

In order to estimate the conditional distribution of Y_N let us consider a family of conditional probability models. As in the statistical literature, a *model* m is a family of conditional distributions which are indexed by a parameter $\theta_m \in \Theta_m \subset \mathbb{R}^{d(m)}$, where $d(m)$ is called the *dimension* of the model m.

The models we consider are represented by *trees*. A tree $\mathcal{S}$ is by definition a non-empty set of strings $\mathcal{S} \subset \mathcal{A}^*$ such that *every suffix of every string of $\mathcal{S}$ be also in $\mathcal{S}$*. In particular, this implies that the empty string λ belongs to $\mathcal{S}$. Any tree can be represented graphically as a graph whose vertices are the strings it is made of and whose edges link together every string $s \in \mathcal{S}$ with its suffix of size $l(s) - 1$. As an example, Figure 1 shows a tree $\mathcal{S} = \{\lambda, a, ba, b, c, ac, bc\}$ when $\mathcal{A} = \{a, b, c\}$. The parent of a string $s \in \mathcal{S}$ is its suffix of size $l(s) - 1$, and its children are the set of strings $s' \in \mathcal{S}$ of length $l(s') = l(s) + 1$ such that s is a suffix of s'. Not that a tree might be *incomplete*, i.e. the number of children of any string $s \in \mathcal{S}$ might be different from 0 or a.

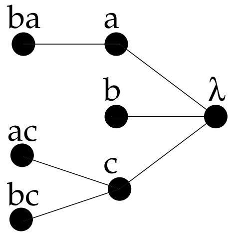


Figure 1: Representation of the tree model $\{\lambda, a, ba, b, c, ac, bc\}$

We denote by $\mathcal{C}_D$ the *tree class of memory D*, i.e. the set of trees $\mathcal{S}$ such that for any $s \in \mathcal{S}$, $l(s) \leq D$.

For any tree $\mathcal{S} \in \mathcal{C}_D$ the *suffix functional* $s_{\mathcal{S}} : \mathcal{X} \to \mathcal{S}$ is the mapping which transforms any string $x \in \mathcal{X}$ into its longest suffix that is an element of $\mathcal{S}$. If there is no ambiguity on the tree considered, we will also write s instead of $s_{\mathcal{S}}$.

Example 1 *The suffix functional s associated with the tree represented in Figure 1 is such that $s(\ldots bac) = ac$ and $s(\ldots bcc) = c$*

Any tree $\mathcal{S} \in \mathcal{C}_D$ can be considered as a conditional distribution model thanks to the following construction :

Definition 1 *Let $\mathcal{S} \in \mathcal{C}_D$ be a tree and Σ be the $(a-1)$-dimensional simplex $\Sigma = \{\theta \in [0,1]^{\mathcal{Y}} : \sum_{y \in \mathcal{Y}} \theta(y) = 1\}$. For any $\theta = (\theta_s)_{s \in \mathcal{S}} \in \Sigma^{\mathcal{S}}$ let $P_{\mathcal{S},\theta}$ denote the conditional probability density on $\mathcal{X} \times \mathcal{Y}$ defined by:*

$$\forall (x, y) \in \mathcal{X} \times \mathcal{Y} \qquad P_{\mathcal{S},\theta}\,(y \,|\, x) \stackrel{def}{=} \theta_{s_{\mathcal{S}}(x)}(y) \ .$$

The tree model $\mathcal{S}$ *is by definition the set of conditional densities $\{P_{\mathcal{S},\theta} : \theta \in \Sigma^{\mathcal{S}}\}$.*

As a result, a tree model $\mathcal{S}$ is a model with dimension $d(\mathcal{S}) = |\mathcal{S}| \times (a-1)$.

2.3 Estimator for a given tree model

Let us first consider the case when a tree model $\mathcal{S} \in \mathcal{C}_D$ is given and one wants to use the observations Z_1^{N-1} is order to estimate a parameter $\hat{\theta}(Z_1^{N-1}) \in \Sigma^{\mathcal{S}}$ such that $R_{P_N}\left(P_{\mathcal{S},\hat{\theta}(Z_1^{N-1})}\right)$ be "small". We propose two estimators for this problem: the first one is the well-known *Laplace estimator* for which we generalize known universal bounds (Theorem 1), while the second one is a new estimator for which we prove a better bound when the support of the conditional distribution is smaller than the whole alphabet (Theorem 2). $\mathcal{S}$ being fixed we will use the notation $s(.)$ instead of $s_{\mathcal{S}}(.)$ for the suffix functional associated with $\mathcal{S}$.

Remark 1 *The problem of parameter estimation for an i.i.d. source on a finite space is well known in Information Theory. It seems the first method was considered in [73]; then the problem of optimal estimation was considered in [53] and an asymptotically optimal method was suggested. Recently new results about exact prediction were found in [83]. The results that follow are non-asymptotic (as opposed to [53]) and remain true if the samples are not i.i.d. but only drawn from an exchangeable distribution. Even though the estimators we study are not asymptotically minimax (as opposed to [53]) the non-asymptotic upper bounds we obtain are of the order of the minimax risk.*

2.3.1 Laplace estimator

For any $n \in \mathbb{N}$ let us introduce the random variables :

$$\begin{cases} \forall (y,s) \in \mathcal{Y} \times \mathcal{S} & \mu_n(s,y) = \sum_{i=1}^{n} \mathbf{1}\left(s\left(X_i\right) = s \text{ and } Y_i = y\right) \quad , \\ \forall s \in \mathcal{S} & \nu_n(s) = \sum_{i=1}^{n} \mathbf{1}\left(s\left(X_i\right) = s\right) \quad . \end{cases} \tag{2.3}$$

Hence $\nu_n(s)$ counts the number of samples Z_i in $Z_1, \ldots, Z_n$ such that X_i is mapped to s by the suffix functional $s(.)$, and $\mu_n(s,y)$ counts the number of samples in that subset such that $Y_i = y$.

A node $s \in \mathcal{S}$ is said to be *visited* by Z_1^N if $\nu_N(s) > 0$, and we denote by $v_N(\mathcal{S})$ the random set of visited nodes, i.e. :

$$v_N(\mathcal{S}) \stackrel{def}{=} \{s \in \mathcal{S} \,:\, \nu_N(s) > 0\} \quad .$$

The Laplace estimator $\hat{\theta}$ is defined by:

$$\forall (s,y) \in \mathcal{S} \times \mathcal{Y}, \qquad \hat{\theta}_s(y) = \frac{\mu_{N-1}(s,y)+1}{\nu_{N-1}(s)+a} \ ,$$

and results in an estimator which we call the *Laplace estimator for the tree* $\mathcal{S}$ defined by the formula :

$$\forall z_1^N \in (\mathcal{X} \times \mathcal{Y})^N, \qquad Q_{\mathcal{S}}^N\left(y_N \,|\, x_N; z_1^{N-1}\right) = \frac{\mu_{N-1}(s\left(x_N\right), y_N)+1}{\nu_{N-1}(s\left(x_N\right))+a} \ . \tag{2.4}$$

The following theorem gives an upper bound for the risk of this estimator :

Theorem 1 *For any exchangeable distribution P_N on $(\mathcal{X} \times \mathcal{Y})^N$ and for any tree $\mathcal{S} \in \mathcal{C}_D$ the risk of the Laplace estimator for the tree $\mathcal{S}$ satisfies :*

$$\begin{aligned} R_{P_N}\left(Q_{\mathcal{S}}^N\right) &\leq \inf_{\theta \in \Sigma^{\mathcal{S}}} R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{a-1}{N} \mathbf{E}_{P_N} \left|v_N(\mathcal{S})\right| \\ &\leq \inf_{\theta \in \Sigma^{\mathcal{S}}} R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{a-1}{N} \left|\mathcal{S}\right| \ . \end{aligned}$$

Remark 2 *The first inequality of Theorem 1 shows that the risk bound depends on the design distribution, i.e. on the distribution of X_1^N, and therefore that the Laplace estimator can* adapt *to it.*

When $\mathcal{S}$ is reduced to a single node, this result is proven in [73] when P_N is a product distribution and in [24] when P_N is exchangeable. Here we generalize the method of proof of the latter for a general tree model $\mathcal{S}$ (see also [16] for a similar result in the case of decision trees).

Proof of Theorem 1:

First observe that for any $s \in \mathcal{S}$,

$$\begin{aligned} \nu_N\left(s(X_N)\right) &= \sum_{i=1}^{N} \mathbf{1}\left(s\left(X_i\right) = s(X_N)\right) \\ &= \sum_{i=1}^{N-1} \mathbf{1}\left(s\left(X_i\right) = s(X_N)\right) + 1 \\ &= \nu_{N-1}(s(X_N)) + 1 \ . \end{aligned}$$

and a similar computation shows that for any $(s,y) \in \mathcal{S} \times \mathcal{Y}$,

$$\mu_N\left(s(X_N), Y_N\right) = \mu_{N-1}\left(s(X_N), Y_N\right) + 1 \ .$$

As a result the Laplace estimator (2.4) can be rewritten in terms of μ_N and ν_N as follows:

$$\forall z_1^N \in (\mathcal{X} \times \mathcal{Y})^N, \qquad Q_{\mathcal{S}}^N \left(y_N \,|\, x_N ; z_1^{N-1} \right) = \frac{\mu_N(s(x_N), y_N)}{\nu_N(s(x_N)) + a - 1} \ .$$

Observe also that the maximum likelihood estimator for $\prod_{i=1}^N P_{\mathcal{S},\theta}(Y_i \,|\, X_i)$ is:

$$\hat{\theta}_s(y) = \mu_N(s,y)/\nu_N(s) \ ,$$

with corresponding log-likelihood :

$$\sup_{\theta \in \Sigma^{\mathcal{S}}} \log \prod_{i=1}^N P_{\mathcal{S},\theta}(Y_i \,|\, X_i) = \sum_{\substack{s \in \mathcal{S} \\ \nu_N(s) > 0}} \sum_{y \in \mathcal{Y}} \mu_N(s,y) \log \frac{\mu_N(s,y)}{\nu_N(s)} \ .$$

Using the fact that P_N is exchangeable to get the first equality and the fact that μ_N and ν_N are invariant under permutations of $\{1, \dots, N\}$ to get the second, we can now write :

$$\begin{aligned}
&\mathbf{E}_{P_N} \log \frac{1}{Q_{\mathcal{S}}^N \left(Y_N \,|\, X_N ; Z_1^{N-1} \right)} \\
&\quad = -\frac{1}{N} \mathbf{E}_{P_N} \sum_{i=1}^N \log Q_{\mathcal{S}}^N \left(Y_i \,|\, X_i ; Z_k, k \neq i, 1 \leq k \leq N \right) \\
&\quad = -\frac{1}{N} \mathbf{E}_{P_N} \sum_{s \in \mathcal{S}} \sum_{y \in \mathcal{Y}} \mu_N(s,y) \log \frac{\mu_N(s,y)}{\nu_N(s) + a - 1} \\
&\quad = \mathbf{E}_{P_N} \inf_{\theta \in \Sigma^{\mathcal{S}}} - \frac{1}{N} \log \prod_{i=1}^N P_{\mathcal{S},\theta} \left(Y_i \,|\, X_i \right) + \frac{1}{N} \sum_{\substack{s \in \mathcal{S} \\ \nu_N(s) > 0}} \mathbf{E}_{P_N} \nu_n(s) \log \left(1 + \frac{a-1}{\nu_n(s)} \right) \\
&\quad \leq \inf_{\theta \in \Sigma^{\mathcal{S}}} \mathbf{E}_{P_N} \log \frac{1}{P_{\mathcal{S},\theta} \left(Y_N \,|\, X_N \right)} + \frac{a-1}{N} \mathbf{E}_{P_N} |v_N(\mathcal{S})| \ .
\end{aligned}$$

Theorem 1 follows by adding $\mathbf{E}_{P_N} \log P_N \left(Y_N \,|\, X_N ; Z_1^{N-1} \right)$ to both sides of the inequality and observing that $v_N(\mathcal{S}) \subset \mathcal{S}$ implies $|v_N(\mathcal{S})| \leq |\mathcal{S}|$.$\square$

2.3.2 Adaptive Laplace estimator

In this Section we suppose that P_N is a product measure $P^{\otimes N}$ with $P \in \mathcal{M}_+^1(\mathcal{X} \times \mathcal{Y})$, i.e. $Z_1, \dots, Z_N$ are supposed to be i.i.d. with common distribution P.

Suppose that for every $s \in \mathcal{S}$ the support of the conditional distribution $P\left(Y \,|\, s(X) = s\right)$ is known to be a subset $\mathcal{A}_s \subset \mathcal{A}$ of size $a(s) = |\mathcal{A}_s|$, i.e. :

$$\forall (x,y) \in \mathcal{X} \times \mathcal{Y}, \qquad \begin{cases} P(y \,|\, x) > 0 \text{ if } y \in \mathcal{A}_{s(x)}, \\ P(y \,|\, x) = 0 \text{ otherwise.} \end{cases}$$

In that case one could replace the Laplace estimator for the tree $\mathcal{S}$ by the following estimator which takes into account the information about the supports :

$$\bar{Q}_{\mathcal{S}}^{N}\left(y_N \,|\, x_N ; z_1^{N-1}\right) = \begin{cases} \dfrac{\mu_{N-1}(s(x_N), y_N) + 1}{\nu_{N-1}(s(x_N)) + a(s)} & \text{if } y_N \in \mathcal{A}_{s(x_n)}, \\ 0 & \text{otherwise.} \end{cases}$$

Using a computation similar to the one in the proof of Theorem 1 it is straightforward to show that this estimator satisfies :

$$R_{P_N}\left(\bar{Q}_{\mathcal{S}}^{N}\right) \leq \inf_{\theta \in \Sigma^{|\mathcal{S}|}} R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{\sum\limits_{s \in \mathcal{S}} (a(s) - 1)}{N} \quad , \tag{2.5}$$

which is a smaller upper bound than the one given in Theorem 1 if $a(s) < a$ for some $s \in \mathcal{S}$. However this estimator requires the prior knowledge of the supports $\{\mathcal{A}_s\}_{s \in \mathcal{S}}$. In case these supports are not known it is still possible to observe the size of the empirical supports given by :

$$\forall (n,s) \in \mathbb{N} \times \mathcal{S}, \qquad a_n(s) = \sum_{y \in \mathcal{A}} \mathbf{1}\left(\mu_n(s,y) > 0\right) \quad .$$

Using these observations we define the *adaptive Laplace estimator for the tree* $\mathcal{S}$ by the formula, $\forall z_1^N \in (\mathcal{X} \times \mathcal{Y})^N$:

$$\tilde{Q}_{\mathcal{S}}^{N}\left(y_N \,|\, x_N ; z_1^{N-1}\right) = \begin{cases} \dfrac{\mu_{N-1}\left(s(x_N), y_N\right) + \dfrac{a_{N-1}\left(s(x_N)\right)}{a}}{\nu_{N-1}\left(s(x_N), y_N\right) + a_{N-1}\left(s(x_N)\right)} & \text{if } \nu_{N-1}(s(x_N)) > 0, \\ \dfrac{1}{a} & \text{otherwise .} \end{cases}$$

The effect of this modification to the Laplace estimator is to “boost” the estimated probabilities of letters which have already been observed. It is easy to check that :

$$\forall (z_1^{N-1}, x_N) \in (\mathcal{X} \times \mathcal{Y})^{N-1} \times \mathcal{X}, \qquad \sum_{y \in \mathcal{Y}} \tilde{Q}_{\mathcal{S}}^{N}(y \,|\, x_N ; z_1^{N-1}) = 1 \quad ,$$

which ensures that $\tilde{Q}_{\mathcal{S}}^{N}$ is an admissible conditional probability density. The risk of this estimator can be upper bounded as follows :

Theorem 2 *For any probability distribution P on $\mathcal{X}\times\mathcal{Y}$ and $P_N = P^{\otimes N}$, for any incomplete tree model $\mathcal{S}\in\mathcal{C}_D$,*

$$R_{P_N}\left(\tilde{Q}_{\mathcal{S}}^N\right) \leq \inf_{\theta\in\Sigma^{\mathcal{S}}} R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{\sum_{s\in\mathcal{S}}\gamma_N(s)}{N} \ ,$$

with

$$\forall s\in\mathcal{S},\qquad \gamma_N(s) = a(s)\left(1-\frac{a(s)}{a}\right) + a(s) - 1 + o(1) \ .$$

Remark 3 *Up to the vanishing term $o(1)$ the upper bound provided in Theorem 2 is smaller than the upper bound provided by Theorem 1 for the Laplace estimator by a factor :*

$$\frac{1}{N}\sum_{s\in\mathcal{S}}\left(a-1-a(s)+1-a(s)\left(1-\frac{a(s)}{a}\right)\right) = \frac{1}{N}\sum_{s\in\mathcal{S}}\frac{\left(a-a(s)\right)^2}{a} \ ,$$

which is always positive. Therefore the asymptotic rate of convergence to zero is smaller for the adaptive Laplace estimator than for the Laplace estimator if $a(s) < a$ for some s.

However by (2.5) the corresponding rate of convergence for the risk of the Laplace estimator $\bar{Q}_{\mathcal{S}}^N$ in the case $\{\mathcal{A}_s\}_{s\in\mathcal{S}}$ is known is $\sum_{s\in\mathcal{S}}(a(s)-1)/N$, which is smaller than the upper bound of Theorem 2 by a factor :

$$\frac{1}{N}\sum_{s\in\mathcal{S}}a(s)\left(1-\frac{a(s)}{a}\right) \leq \frac{1}{N}\sum_{s\in\mathcal{S}}a(s) \ .$$

This factor can be considered as the "cost" of not knowing $\{\mathcal{A}_s\}_{s\in\mathcal{S}}$.

Proof of theorem 2:

First observe that if $\mu_N(s(X_N),Y_N)=1$ then for all $i<N$, $s(X_i)\neq s(X_N)$ or $Y_i\neq Y_N$. As a result, $a_N\left(s(X_N)\right) = a_{N-1}\left(s(X_N)\right)+1$. On the other hand if $\mu_N(s(X_N),Y_N)>1$ then $a_N\left(s(X_N)\right) = a_{N-1}\left(s(X_N)\right)$. Therefore we can compute :

$$\mathbf{E}_{P_N}\log\frac{1}{\tilde{Q}_{\mathcal{S}}^N\left(Y_N \,|\, X_N; Z_1^{N-1}\right)}$$

$$
\begin{aligned}
&= -\frac{1}{N}\mathbf{E}_{P_N} \sum_{i=1}^{N} \log \tilde{Q}_{\mathcal{S}}^{N}\left(Y_i \,|\, X_i; Z_k, k \neq i, 1 \leq k \leq N\right) \\
&= -\frac{1}{N}\mathbf{E}_{P_N} \sum_{\substack{s\in\mathcal{S}\\ \nu_N(s)>1}} \sum_{y\in\mathcal{Y}} \mu_N(s,y) \log \frac{\mu_N(s,y) - 1 + \frac{a_{N-1}(s)}{a}}{\nu_N(s) - 1 + a_{N-1}(s)} - \frac{1}{N}\mathbf{E}_{P_N} \sum_{\substack{s\in\mathcal{S}\\ \nu_N(s)=1}} \log \frac{1}{a} \\
&= -\frac{1}{N}\mathbf{E}_{P_N} \sum_{\substack{s\in\mathcal{S}\\ \nu_N(s)>1}} \left(\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\geq 2}} \mu_N(s,y) \log \frac{\mu_N(s,y) - 1 + \frac{a_N(s)}{a}}{\nu_N(s) + a_N(s) - 1} \right. \\
&\quad \left. + \sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)=1}} \log \frac{\frac{a_N(s)-1}{a}}{\nu_N(s) + a_N(s) - 2} \right) - \frac{1}{N}\mathbf{E}_{P_N} \sum_{\substack{s\in\mathcal{S}\\ \nu_N(s)=1}} \log \frac{1}{a} \\
&\leq \inf_{\theta\in\Sigma^{\mathcal{S}}} \mathbf{E}_{P_N} \log \frac{1}{P_{\mathcal{S},\theta}\left(Y_N \,|\, X_N\right)} + \frac{1}{N} \sum_{s\in\mathcal{S}} \left(A_s + B_s + C_s + D_s + E_s\right) \quad ,
\end{aligned}
$$

with :

$$
\begin{cases}
A_s = \mathbf{E}_{P_N} \displaystyle\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\geq 2}} \mu_N(s,y) \log \frac{\mu_N(s,y)}{\mu_N(s,y) - 1 + \frac{a_N(s)}{a}} \quad , \\
B_s = \mathbf{E}_{P_N} \displaystyle\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\geq 2}} \mu_N(s,y) \log \frac{\nu_N(s) + a_N(s) - 1}{\nu_N(s)} \quad , \\
C_s = \mathbf{E}_{P_N} \left(\mathbf{1}(\nu_N(s) > 1) \displaystyle\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)=1}} \log \frac{a}{a_N(s) - 1} \right) \quad , \\
D_s = \mathbf{E}_{P_N} \left(\mathbf{1}(\nu_N(s) > 1) \displaystyle\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)=1}} \log \frac{\nu_N(s) + a_N(s) - 2}{\nu_N(s)} \right) \quad , \\
E_s = \mathbf{E}_{P_N} \left(\mathbf{1}(\nu_N(s) = 1) \log a \right) \quad .
\end{cases}
$$

For any $s \in \mathcal{S}$ and $y \in \mathcal{Y}$ let $P(s) = Pr\{s(X) = s\}$ and $\theta_s(y) = Pr\{Y = y \,|\, s(X) = x\}$. Then $\nu_N(s)$ and $\mu_N(s,y)$ are binomial variables $B(N, P(s))$ and $B\left(N, P(s)\theta_s(y)\right)$. Let $\epsilon > 0$ be defined by :

$$
\epsilon = \min_{\substack{s\in\mathcal{S}\\ P(s)>0}} \min_{y\in\mathcal{A}_s} P(s)\theta_s(y) \quad .
$$

Then for any $s \in \mathcal{S}$ such that $P(s) > 0$, the expectation of the empirical support size satisfies :

$$
\begin{aligned}
a(s) - \mathbf{E}_{P_N} a_N(s) &= a(s) - \sum_{k=0}^{a(s)} k.Pr\{a_N(s) = k\} \\
&\leq a(s) Pr\{a_N(s) < a(s)\} \\
&\leq a(s) \left(\sum_{y \in \mathcal{A}_s} Pr\{\mu_N(s,y) = 0\} \right) \\
&\leq a(s) \sum_{y \in \mathcal{A}_s} (1-\epsilon)^N \\
&\leq a(s)^2 e^{-N\epsilon} \quad .
\end{aligned}
\tag{2.6}
$$

On the other hand if $\mu_N(s,y) \geq 2$ then $(\mu_N(s,y) - 1)^{-1} \leq 3\,(\mu_N(s,y)+1)^{-1}$ and therefore, for any $s \in \mathcal{S}$ such that $P(s) > 0$ we have :

$$
\begin{aligned}
\mathbf{E}_{P_N} \sum_{\substack{y \in \mathcal{Y} \\ \mu_N(s,y) \geq 2}} \frac{1}{\mu_N(s,y) - 1} &\leq 3\mathbf{E}_{P_N} \sum_{\substack{y \in \mathcal{Y} \\ \mu_N(s,y) \geq 2}} \frac{1}{\mu_N(s,y) + 1} \\
&\leq 3 \sum_{y \in \mathcal{A}_s} \mathbf{E}_{P_N} \frac{1}{1 + B(N, P(s)\theta_s(y))} \\
&\leq 3 \sum_{y \in \mathcal{A}_s} \frac{1}{(N+1)P(s)\theta_s(s)} \\
&\leq \frac{3a(s)}{N\epsilon} \quad ,
\end{aligned}
\tag{2.7}
$$

where we used the fact (see e.g. [34, p. 587]) that for a binomial $B(n,p)$, $\mathbf{E}1/(1+B(n,p)) \leq 1/((n+1)p)$.

We can now upper bound the five terms for any $s \in \mathcal{S}$ such that $P(s) > 0$. For A_s we write :

$$
\begin{aligned}
A_s &\leq \mathbf{E}_{P_N} \sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\geq 2}} \mu_N(s,y) \frac{1-\frac{a_N(s)}{a}}{\mu_N(s,y)-1+\frac{a_N(s)}{a}} \\
&\leq \mathbf{E}_{P_N} \sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\geq 2}} \left[1-\frac{a_N(s)}{a} + \frac{\left(1-\frac{a_N(s)}{a}\right)^2}{\mu_N(s,y)-1+\frac{a_N(s)}{a}} \right] \\
&\leq a(s)\left(1-\frac{a(s)}{a}\right) + \frac{a(s)}{a}\left(a(s)-\mathbf{E}_{P_N} a_N(s)\right) + \mathbf{E}_{P_N} \sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\geq 2}} \frac{1}{\mu_N(s,y)-1} \\
&\leq a(s)\left(1-\frac{a(s)}{a}\right) + \frac{a(s)^3}{a} e^{-N\epsilon} + \frac{3a(s)}{N\epsilon} \quad ,
\end{aligned}
$$

where (2.6) and (2.7) are used to get the last inequality. The terms B_s and D_s can be taken together :

$$
\begin{aligned}
B_s + D_s &\leq \mathbf{E}_{P_N} \left(\mathbf{1}(\nu_N(s)>1) \sum_{y\in\mathcal{A}_s} \mu_N(s,y) \log \frac{\nu_N(s)+a(s)-1}{\nu_N(s)} \right) \\
&\leq \mathbf{E}_{P_N} \left(\mathbf{1}(\nu_N(s)>1) \nu_N(s) \log\left(1+\frac{a(s)-1}{\nu_N(s)}\right)\right) \\
&\leq a(s)-1 \quad .
\end{aligned}
$$

Finally, one can observe that if $\mu_N(s,y)=1$ and $\nu_N(s)\geq 2$ then $a_N(s)\geq 2$. This provides an upper bound for the integrand in C_s and therefore :

$$
\begin{aligned}
C_s + E_s &\leq \mathbf{E}_{P_N} \left(\mathbf{1}(\nu_N(s)>1) \sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)=1}} \log a + \mathbf{1}(\nu_N(s)=1)\log a \right) \\
&\leq \log(a) \times \left(\sum_{y\in\mathcal{A}_s} Pr\{\mu_N(s,y)=1\} + Pr\{\nu_N(s)=1\} \right) \\
&\leq N\log(a) \left[(1-P(s))^{N-1} + \sum_{y\in\mathcal{A}_s} (1-P(s)\theta_s(y))^{N-1} \right] \\
&\leq N\log(a)\left(a(s)+1\right) e^{-(N-1)\epsilon} \quad .
\end{aligned}
$$

We can now sum up the upper bounds obtained for A_s, B_s, C_s, D_s and E_s to get :

$$R_{P_N}\left(\tilde{Q}_{\mathcal{S}}^N\right) \leq \inf_{\theta\in\Sigma^{\mathcal{S}}} R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{1}{N}\sum_{\substack{s\in\mathcal{S}\\ P(s)>0}}\left[a(s)-1+a(s)\left(1-\frac{a(s)}{a}\right)+\zeta_N(s)\right] \ ,$$

with

$$\zeta_N(s) = \frac{a(s)^3}{a}e^{-N\epsilon} + \frac{3a(s)}{N\epsilon} + N\log(a)\left(a(s)+1\right)e^{-(N-1)\epsilon} \ .$$

This finishes the proof of Theorem 2. □

2.4 Probability on the model space

The goal in the rest of this paper is to build estimators which satisfy risk bounds like (2.1). For this purpose we propose to use aggregation methods introduced by Catoni (the progressive mixture estimator in [24] and the Gibbs estimator in [22]), both of which require a prior probability distribution to be given on the model set. The idea of setting a probability on a model space is well known in source coding and prediction : besides underlying any Bayesian approach it was suggested in [71] and [72] to obtain non-asymptotic risk bounds and later this idea was used in many papers (see for example [91] and [92]).

If π is a probability distribution on a model space $\mathcal{M}$ then $\log 1/\pi(m)$ is called *model risk*. The choice of π is arbitrary, but has an influence on the performance of the aggregated estimator. Optimizing this choice is impossible without further assumptions on the true probability distribution P and the approximation properties of the family of models considered.

In addition to performance the possibility of a fast implementation should be regarded as a guideline for the choice of a prior distribution π. For instance the prior model probability distribution considered in the context tree weighting algorithm ([91]) leads to a remarkably efficient implementation, which should be regarded as a fundamental advantage of the algorithm.

Generalizing the idea of the context tree weighting method, let us define a probability distribution π_D on $\mathcal{C}_D$, the tree class of memory D, as follows :

$$\forall\mathcal{S}\in\mathcal{C}_D, \qquad \pi_D\left(\mathcal{S}\right) = c_D^{|\mathcal{S}|} \ ,$$

where $c_D\in\mathbb{R}$ satisfies :

$$\sum_{\mathcal{S}\in\mathcal{C}_D} c_D^{|\mathcal{S}|} = 1 \ . \tag{2.8}$$

The model risk is then linear with respect to the size of the model, because :

$$\forall \mathcal{S} \in \mathcal{C}_D, \qquad \log \frac{1}{\pi_D(\mathcal{S})} = |\mathcal{S}| \log \frac{1}{c_D} \quad . \tag{2.9}$$

The prior π_D will be used in the following sections to build convex combinations of different models. We will obtain particular upper bounds for the risks with this arbitrary choice (Theorems 4 and 6), but the reader should be aware that any different choice of prior is possible and would lead to different upper bounds. We propose to chose a prior which results in a model risk proportional to $|\mathcal{S}|$ because the "parameter risk", i.e. the risk of an estimator for the model $\mathcal{S}$ like the Laplace estimator, is also linear in $|\mathcal{S}|$ (Theorem 1).

The following lemma provides a useful upper bound on the model risk independent of D :

Lemma 2 *The family of probabilities* $\{\pi_D\}_{D\in\mathbb{N}}$ *satisfies :*

$$\forall D \in \mathbb{N}, \forall \mathcal{S} \in \mathcal{C}_D, \qquad \log \frac{1}{\pi_D(\mathcal{S})} \leq |\mathcal{S}| \left(\log(a) + 1\right) \quad .$$

Proof of Lemma 2:

By (2.8) it is clear that $(c_D)_{D\in\mathbb{N}}$ is a decreasing function of D, because $\mathcal{C}_D \subset \mathcal{C}_{D+1}$ for any $D \in \mathbb{N}$. Therefore this non-negative series has a limit $c = \lim_{D\to\infty} c_D$, such that $\forall D \in \mathbb{N}, c \leq c_D$.

For any $(D, x) \in \mathbb{N} \times \mathbb{R}$ let :

$$u_D(x) = \sum_{\mathcal{S}\in\mathcal{C}_D} x^{|\mathcal{S}|} \quad .$$

The function $u_D(x)$ is increasing with x and D, and by definition $u_D(c_D) = 1$ for any $D \in \mathbb{N}$. Therefore $u_D(c) \leq 1$ for any $D \in \mathbb{N}$, and :

$$\lim_{D\to\infty} u_D(c) \leq 1 \quad . \tag{2.10}$$

By decomposing any tree $\mathcal{S} \in \mathcal{C}_D$ as the root node and a (eventually empty) subtrees $(\mathcal{S}_1, \ldots, \mathcal{S}_a) \in (\mathcal{C}_{D-1} \cup \{\emptyset\})^a$ one gets the following inductive relation:

$$\begin{aligned} u_D(x) &= \sum_{\mathcal{S}\in\mathcal{C}_D} x^{|\mathcal{S}|} \\ &= \sum_{(\mathcal{S}_1,\ldots,\mathcal{S}_a)\in(\mathcal{C}_{D-1}\cup\{\emptyset\})^a} x^{1+|\mathcal{S}_1|+\ldots+|\mathcal{S}_a|} \\ &= x\left(u_{D-1}(x) + 1\right)^a \quad . \end{aligned}$$

If we introduce the function $f_x(y) = x(1+y)^a$ then this can be rewritten :

$$u_D(x) = f_x\left(u_{D-1}(x)\right) \quad .$$

It is well known that for $u_D(x)$ to stay bounded when D tends to infinity it is necessary that the equation $f_x(y) = y$ have a solution y. By (2.10) this implies that $f_c(y) - y$ must be equal to zero for some y.

If we now study the function $g_x(y) = f_x(y) - y$ its derivative is :

$$g_x'(y) = ax(1+y)^{a-1} - 1 \ ,$$

therefore g_x is minimum for y^* such that $g_x'(y^*) = 0$, i.e. :

$$y^* = (ax)^{\frac{1}{1-a}} - 1 \ .$$

As a result the minimum value of g_x is :

$$g_x(y^*) = 1 - \frac{a-1}{a}(ax)^{\frac{1}{1-a}} \quad .$$

The necessary condition that $f_c(y) - y = 0$ for some y is equivalent to $g_c(y^*) \leq 0$, i.e. :

$$c \leq \frac{(a-1)^{a-1}}{a^a} \ ,$$

which implies :

$$\begin{aligned}
\log\frac{1}{c} &\leq a\log(a) - (a-1)\log(a-1) \\
&\leq \log(a) + (a-1)\log\frac{a}{a-1} \\
&\leq \log(a) + 1 \ .
\end{aligned}$$

Lemma 2 now follows from this inequality, the fact that $c \leq c_D$ and (2.9). □

2.5 Aggregation using a progressive mixture estimator

In Section 2.3 we presented two estimators for the parameters of every given model $\mathcal{S}$: the Laplace estimator $Q_{\mathcal{S}}$ and the adaptive Laplace estimator $\tilde{Q}_{\mathcal{S}}$. In this Section we show how to aggregate the Laplace (resp. adaptive Laplace) estimators for various $\mathcal{S}$, i.e. build a convex combination of the $\{Q_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$ (resp. $\{\tilde{Q}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$), by using the so-called *progressive mixture* *estimator*, introduced by Catoni in [24]. Instead of selecting one model $\hat{\mathcal{S}}$ and the corresponding estimator $Q_{\hat{\mathcal{S}}}$ (resp. $\tilde{Q}_{\hat{\mathcal{S}}}$) as in classical model selection procedures, this estimator is a mixture of all the Laplace (resp. adaptive Laplace) estimators.

Let us first describe the construction of the progressive mixture estimator which aggregates the Laplace estimators defined in section 2.3 and which we denote by $Q_\pi^N(Y_N \mid X_N; Z_1^{N-1})$.

An integer $K \in [1, N-1]$ is first chosen and the observations Z_1^{N-1} are split into an *estimation set* Z_1^K and a *validation set* Z_{K+1}^{N-1}.

For each model $\mathcal{S} \in \mathcal{C}_D$ the estimation set is mapped by the Laplace estimator to a conditional distribution $Q_\mathcal{S}^{K+1}(Y \mid X)$ defined by :

$$\forall \mathcal{S} \in \mathcal{C}_D, \qquad Q_\mathcal{S}^{K+1}(Y \mid X) = Q_\mathcal{S}^{K+1}\left(Y \mid X; Z_1^K\right) \quad , \tag{2.11}$$

where the latter is defined by (2.4).

For any $n \in [0, N-K-1]$ let now $Q_\pi^{(n)}(Y \mid X)$ be the conditional distribution obtained as a Bayesian mixture of the primary estimators $\{Q_\mathcal{S}^{K+1}(Y \mid X)\}_{\mathcal{S} \in \mathcal{C}_D}$ with the prior distribution π on $\mathcal{C}_D$ and the observations Z_{K+1}^{K+n}, i.e. :

$$Q_\pi^{(n)}(Y \mid X) = \frac{\sum_{\mathcal{S} \in \mathcal{C}_D} \pi(S) \left(\prod_{i=K+1}^{K+n} Q_\mathcal{S}^{K+1}(Y_i \mid X_i)\right) Q_\mathcal{S}^{K+1}(Y \mid X)}{\sum_{\mathcal{S} \in \mathcal{C}_D} \pi(S) \left(\prod_{i=K+1}^{K+n} Q_\mathcal{S}^{K+1}(Y_i \mid X_i)\right)} \ .$$

The progressive mixture estimator Q_π^N is then a Cesaro mean of these Bayesian estimators trained on subsamples of growing sizes, i.e :

$$Q_\pi^N(Y_N \mid X_N; Z_1^{N-1}) \stackrel{def}{=} \frac{1}{N-K} \sum_{n=0}^{N-K-1} Q_\pi^{(n)}(Y_N \mid X_N) \ .$$

The idea of building a progressive estimator has been proposed independently by Barron ([5], [8]) and Catoni ([24]) who proved the following property :

Theorem 3 (Catoni, [24])

$$R_{P_N}\left(Q_\pi^N\right) \leq \inf_{\mathcal{S} \in \mathcal{C}_D} \left\{ R_{P_N}\left(Q_\mathcal{S}^{K+1}\right) + \frac{1}{N-K} \log \frac{1}{\pi(\mathcal{S})} \right\} \ . \tag{2.12}$$

The construction of the progressive mixture estimator $\tilde{Q}_\pi^N$ which aggregates the adaptive Laplace estimators is exactly the same as the construction of Q_π^N except that each Q should be replaced by $\tilde{Q}$.

We can now evaluate the risks of Q_π^N and $\tilde{Q}_\pi^N$:

Theorem 4 *Let Q^N_π (resp $\tilde{Q}^N_\pi$) denote the progressive mixture estimator based on the family of Laplace estimators $\{Q^{K+1}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$ (resp. adaptive Laplace estimators $\{\tilde{Q}^{K+1}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$) and on the prior π defined in Section 2.4, with the size of the training being set to :*

$$K = \left[\frac{\sqrt{a-1}N - \sqrt{\log(a)+1}}{\sqrt{a-1} + \sqrt{\log(a)+1}}\right] \quad ,$$

where $[.]$ denotes greatest integer.

For any exchangeable distribution P_N on $(\mathcal{X}\times\mathcal{Y})^N$, the risk of Q^N_π satisfies :

$$R_{P_N}\left(Q^N_\pi\right) \le \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}} \left\{ R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{|\mathcal{S}|\, C_N}{N+1} \right\} \quad ,$$

with :

$$C_N = \left(\sqrt{1+\log(a)} + \sqrt{a-1}\right)^2 \left(1 + \frac{1}{N-2}\right) \quad .$$

Let γ_{K+1} be defined in as in Theorem 2. The risk of $\tilde{Q}^N_\pi$ satisfies :

$$R_{P_N}\left(\tilde{Q}^N_\pi\right) \le \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{|\mathcal{S}|}} \left\{ R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{\sum_{s\in\mathcal{S}}\delta_N(s)}{N+1} \right\} \quad ,$$

with :

$$\begin{aligned}\delta_N(s) = \left(\sqrt{\gamma_{K+1}(s)} + \sqrt{\log(a)+1}\right)^2 + \sqrt{\frac{\log(a)+1}{a-1}}\left(\sqrt{a-1} - \sqrt{\gamma_{K+1}(s)}\right)^2 \\ + \frac{\gamma(s)}{(a-1)(N-1)}\left(\sqrt{a-1} + \sqrt{\log(a)+1}\right)^2 \quad .\end{aligned}$$

Remark 4 *The definition of K shows that the larger the alphabet, the longer it takes to train the Laplace estimators compared with the time it takes to aggregate them with the progressive mixture estimator (i.e. K/N increases with a, with limit 1 as a tends to infinity). For a large a the risk bound associated with any model $\mathcal{S}$ is very close to $|\mathcal{S}|\,(a-1)/N$, which is the risk of the Laplace estimator for this model.*

Remark 5 *The term $\delta_N(s)$ is the sum of three terms. The first one is the term one would expect if $\gamma_{K+1}(s)$ was known a priori so that the size of the training set K could be better adjusted. The second one is the loss due to the fact that $\gamma_{K+1}(s)$ is not known a priori and we decided to take for K the value corresponding to the best split for Q^N_π instead of $\tilde{Q}^N_\pi$. The third term vanishes to zero and is the loss due to the fact that K has to be an integer.*

Proof of theorem 4:

Using Theorem 1, Theorem 3 and Lemma 2 we can write :

$$\begin{aligned} R_{P_N}\left(Q_\pi^N\right) &\leq \inf_{\mathcal{S}\in\mathcal{C}_D}\left\{R_{P_N}\left(Q_{\mathcal{S}}^{K+1}\right)+|\mathcal{S}|\frac{\log(a)+1}{N-K}\right\} \\ &\leq \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}}\left\{R_{P_N}\left(P_{\mathcal{S},\theta}\right)+|\mathcal{S}|\left(\frac{a-1}{K+1}+\frac{\log(a)+1}{N-K}\right)\right\} \ . \end{aligned}$$

The function

$$x\mapsto f(x)=\frac{a-1}{x+1}+\frac{\log(a)+1}{N-x}$$

is minimum on $(0,N)$ at the point :

$$x^*=\frac{\sqrt{a-1}N-\sqrt{\log(a)+1}}{\sqrt{a-1}+\sqrt{\log(a)+1}} \ .$$

K must be an integer so a good candidate to ensure a risk as small as possible for Q_π^N is $K=[x^*]$ for which we can compute :

$$\begin{aligned} f(K) &\leq \frac{a-1}{x^*}+\frac{\log(a)+1}{N-x^*} \\ &\leq \frac{(a-1)\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)}{\sqrt{a-1}N-\sqrt{\log(a)+1}}+\frac{(\log(a)+1)\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)}{\sqrt{\log(a)+1}(N+1)} \\ &\leq \frac{\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)^2}{N+1}\left(1+\frac{1}{N-\sqrt{\frac{\log(a)+1}{a-1}}}\right) \ . \end{aligned}$$

The upper bound concerning the Laplace estimator in Theorem 4 follows by observing that $a\geq 2$ and therefore :

$$\sqrt{\frac{\log(a)+1}{a-1}}\leq\sqrt{1+\log(2)}\leq 2 \ .$$

For the second part of the theorem concerning the aggregation of adaptive Laplace estimators we follow the same computation except that by Theorem 2 we get :

$$R_{P_N}\left(\tilde{Q}_\pi^N\right)\leq\inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}}\left\{R_{P_N}\left(P_{\mathcal{S},\theta}\right)+\sum_{s\in\mathcal{S}}g_s(K)\right\} \ ,$$

where g_s is defined by :

$$g_s(x)=\frac{\gamma_{K+1}(s)}{x+1}+\frac{\log(a)+1}{N-x} \ .$$

We now just need an upper bound for $g_s(K)$ where K is chosen as in Theorem 4, which is given by :

$$\begin{aligned}
g_s(K) &\leq \frac{\gamma_{K+1}(s)}{x^*} + \frac{\log(a)+1}{N-x^*} \\
&\leq \frac{\gamma_{K+1}(s)\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)}{\sqrt{a-1}N-\sqrt{\log(a)+1}} + \frac{(\log(a)+1)\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)}{\sqrt{\log(a)+1}(N+1)} \\
&\leq \frac{\sqrt{a-1}+\sqrt{\log(a)+1}}{N+1}\left[\frac{\gamma_{K+1}(s)}{\sqrt{a-1}}\left(1+\frac{\sqrt{a-1}+\sqrt{\log(a)+1}}{N\sqrt{a-1}-\sqrt{\log(a)+1}}\right)+\sqrt{\log(a)+1}\right] \\
&\leq \frac{1}{N+1}\left\{\left(\sqrt{\gamma_{K+1}(s)}+\sqrt{\log(a)+1}\right)^2+\sqrt{\frac{\log(a)+1}{a-1}}\left(\sqrt{a-1}-\sqrt{\gamma_{K+1}(s)}\right)^2\right. \\
&\qquad \left.\frac{\gamma_{K+1}(s)}{(a-1)(N-2)}\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)^2\right\} .\square
\end{aligned}$$

2.6 Aggregation using a Gibbs estimator

In this Section we present a second aggregation method based on the Gibbs estimator, introduced by Catoni in [22]. Let us first describe this estimator $G^N_{\pi,\beta}(Y_N \mid Y_N, Z_1^{N-1})$ to aggregate Laplace estimators.

As for the progressive mixture estimator presented in Section 2.5 the observations Z_1^{N-1} are split into two set Z_1^K and Z_{K+1}^{N-1} where K is an integer in $[1, N-1]$, and the observation set Z_1^K is used to define the set of primary estimators $\{Q_{\mathcal{S}}^{K+1}(Y \mid X)\}_{\mathcal{S}\in\mathcal{C}_D}$ using the Laplace estimators as in (2.11).

The Gibbs estimator at inverse temperature $\beta \in \mathbb{R}_+$ using the prior π on $\mathcal{C}_D$ is now the following conditional distribution :

$$G^N_{\pi,\beta}\left(Y_N \mid X_N; Z_1^{N-1}\right) \stackrel{def}{=} \frac{\displaystyle\sum_{\mathcal{S}\in\mathcal{C}_D} \pi(\mathcal{S})\left(\prod_{n=K+1}^{N-1} Q_{\mathcal{S}}^{K+1}(Y_n \mid X_n)\right)^\beta Q_{\mathcal{S}}^{K+1}(Y_N \mid X_N)}{\displaystyle\sum_{\mathcal{S}\in\mathcal{C}_D} \pi(\mathcal{S})\left(\prod_{n=K+1}^{N-1} Q_{\mathcal{S}}^{K+1}(Y_n \mid X_n)\right)^\beta} . \tag{2.13}$$

This definition shows that the Gibbs estimator can be considered as a "thermalized" version of both the Bayesian ($\beta = 1$) and the maximum likelihood ($\beta = +\infty$) estimators. Catoni studied in [22] this estimator in the high temperature region $\beta < 1$ which is equivalent to a deliberate underestimation of the sample size : to compute the Gibbs estimator, the empirical distribution of $N-K-1$ observations is plugged into the Bayes estimator for a sample of size $\beta(N-K-1)$. The reason to consider high temperatures is that the estimator gains stability with respect to the empirical process when β decreases (at the limit, it is constant when $\beta=0$). This property is used by Catoni to prove a general upper bound for its risk in the spirit of (2.1), which takes the following form in the particular case when the primary estimators are log-bounded :

Theorem 5 (Catoni, [22]) *Let $\chi>0$ such that :*

$$\forall \mathcal{S}\in\mathcal{C}_D, \forall (z_1^K, z)\in(\mathcal{X}\times\mathcal{Y})^{K+1}, \qquad -\chi \le \log Q_{\mathcal{S}}^{K+1}(y\,|\,x, z_1^K) \le 0 \ .$$

If β satisfies :

$$\beta \le \frac{1}{\chi-1}\left(\sqrt{1-(\chi-1)\left(2-\frac{\log\chi}{\chi}\right)\frac{\log\chi}{\chi}}-1\right) \ ,$$

then the Gibbs estimator $G_{\pi,\beta}^N$ defined by (2.13) satisfies :

$$R_{P_N}\left(G_{\pi,\beta}^N\right) \le \inf_{\mathcal{S}\in\mathcal{C}_D}\left\{R_{P_N}\left(Q_{\mathcal{S}}^{K+1}\right)+\frac{1}{\beta(N-K)}\log\frac{1}{\pi\left(\mathcal{S}\right)}\right\} \ . \tag{2.14}$$

The definition of the Gibbs estimator $\tilde{G}_{\pi,\beta}^N\left(Y_N\,|\,X_N; Z_1^{N-1}\right)$ to aggregate adaptive Laplace estimators follows exactly the same construction by replacing every Q by $\tilde{Q}$.

We can now evaluate the risk of $G_{\pi,\beta}^N$ and $\tilde{G}_{\pi,\beta}^N$:

Theorem 6 *Let*

$$\begin{cases} \chi_N = \log\left(N+a\right), \\ \tilde{\chi}_N = \log\left(N+a\right)+\log(a) \ . \end{cases}$$

Let

$$\begin{aligned} \beta_N &= \frac{1}{\chi_N-1}\left(\sqrt{1-(\chi_N-1)\left(2-\frac{\log\chi_N}{\chi_N}\right)\frac{\log\chi_N}{\chi_N}}-1\right) \\ &\underset{N\to+\infty}{\sim} \frac{\sqrt{2\log\log N}}{\log N} \ , \end{aligned}$$

and let $\tilde{\beta}_N$ be deduced from $\tilde{\chi}_N$ as β_N is deduced from χ_N.

Let $G^N_{\pi,\beta}$ (resp $\tilde{G}^N_{\pi,\beta}$) denote the Gibbs estimator at inverse temperature β_N (resp. $\tilde{\beta}_N$) based on the family of Laplace estimators $\{Q^{K+1}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$ (resp. adaptive Laplace estimators $\{\tilde{Q}^{K+1}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$) and on the prior π defined in Section 2.4, with the size of the training being set to :

$$K = \left[\frac{\sqrt{a-1}N - \sqrt{\beta_N^{-1}\left(\log(a)+1\right)}}{\sqrt{a-1} + \sqrt{\beta_N^{-1}\left(\log(a)+1\right)}} \right] \quad ,$$

where $[.]$ denotes greatest integer (resp. to $\tilde{K}$ defined like K with β_N replaced by $\tilde{\beta}_N$).

For any exchangeable distribution P_N on $(\mathcal{X}\times\mathcal{Y})^N$ the risk of $G^N_{\pi,\beta}$ satisfies :,

$$R_{P_N}\left(G^N_{\pi,\beta}\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}} \left\{ R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{|\mathcal{S}|\,C_N}{N+1} \right\} \quad ,$$

with :

$$C_N = \left(\sqrt{\left(1+\log(a)\right)\beta_N^{-1}} + \sqrt{a-1} \right)^2 \left(1 + \frac{1}{N-2} \right) \quad .$$

Let γ_{K+1} be defined in as in Theorem 2. The risk of $\tilde{G}^N_{\pi,\beta}$ satisfies :

$$R_{P_N}\left(\tilde{G}^N_{\pi,\tilde{\beta}}\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}} \left\{ R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \frac{\sum_{s\in\mathcal{S}}\delta_N(s)}{N+1} \right\} \quad ,$$

with :

$$\delta_N(s) = \left(\sqrt{\gamma_{K+1}(s)} + \sqrt{\left(\log(a)+1\right)\tilde{\beta}_N^{-1}} \right)^2 + \sqrt{\frac{\left(\log(a)+1\right)\tilde{\beta}_N^{-1}}{a-1}} \left(\sqrt{a-1} - \sqrt{\gamma_{K+1}(s)} \right)^2 + \frac{\gamma(s)}{(a-1)(N-1)} \left(\sqrt{a-1} + \sqrt{\left(\log(a)+1\right)\tilde{\beta}_N^{-1}} \right)^2 \quad .$$

Remark 6 *Asymptotically, the upper bound on the risks of the Gibbs estimators provided by Theorem 6 appear to be worse than the risks of the corresponding progressive mixture estimators given by Theorem 4 because of the factor $(\beta_N)^{-1}$. This is due to the fact that the inverse temperature has to be taken smaller and smaller as N increases in order to prove that (2.14) holds. However the conditions imposed on β which involve a uniform bound on the likelihood of the primary estimators might very conservative in the particular problem we consider. Therefore larger values of β might also ensure the validity of (2.14), and the actual performance of this estimator is probably better than the one proven in Theorem 6 (it is reasonable to think from the computations in [22] that $\beta = 1/2$ will work in many cases).*

Remark 7 *Even though the risk of the Gibbs estimator is worse than the risk of the progressive mixture estimator one might prefer to implement the former because it only involves the computation of one mixture, while the latter one involves the computation of $N-K$ Bayesian mixtures which are then averaged.*

Proof of theorem 6:

The family of Laplace estimators $\{Q_{\mathcal{S}}^{K+1}\}_{\mathcal{S}\in\mathcal{C}_D}$ is uniformly bounded by :

$$\begin{aligned}
&\forall z_1^{K+1} \in (\mathcal{X}\times\mathcal{Y})^{K+1}, \forall \mathcal{S}\in\mathcal{C}_D,\\
&\qquad 0 \geq \log Q_{\mathcal{S}}^{K+1}\left(y_{K+1}\,|\,x_{K+1};z_1^K\right) = \log\frac{\mu_K(s_{\mathcal{S}}(x_{K+1}),y_K)+1}{\nu_K(s_{\mathcal{S}}(x_{K+1}))+a}\\
&\qquad\qquad \geq -\log\left(K+1+a\right)\\
&\qquad\qquad \geq -\log\left(N+a\right) \quad .
\end{aligned}$$

Similarly the family of adaptive Laplace estimator $\{\tilde{Q}_{\mathcal{S}}^{K+1}\}_{\mathcal{S}\in\mathcal{C}_D}$ is uniformly bounded by :

$$\begin{aligned}
&\forall z_1^{K+1} \in (\mathcal{X}\times\mathcal{Y})^{K+1}, \forall \mathcal{S}\in\mathcal{C}_D,\\
&\qquad 0 \geq \log \tilde{Q}_{\mathcal{S}}^{K+1}\left(y_{K+1}\,|\,x_{K+1};z_1^K\right) \geq -\log\left(N+a\right) - \log(a) \quad .
\end{aligned}$$

We can therefore apply Theorem 5 with χ_N (resp. $\tilde{\chi}_N$) and β_N (resp. $\tilde{\beta_N}$) as defined in Theorem 6 to get :

$$R_{P_N}\left(G_{\pi,\beta}^N\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D}\left\{R_{P_N}\left(Q_{\mathcal{S}}^{K+1}\right) + \frac{1}{\beta_N(N-K)}\log\frac{1}{\pi\left(\mathcal{S}\right)}\right\} \quad ,$$

and

$$R_{P_N}\left(\tilde{G}_{\pi,\tilde{\beta}}^N\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D}\left\{R_{P_N}\left(Q_{\mathcal{S}}^{K+1}\right) + \frac{1}{\tilde{\beta}_N(N-K)}\log\frac{1}{\pi\left(\mathcal{S}\right)}\right\} \quad ,$$

Using these two inequalities instead of (2.12) the proof of Theorem 6 now follows exactly the proof of Theorem 4. □

2.7 Data-dependent prior on the trees

Theorem 1 provides two bounds for the risk of the Laplace estimator on a given tree : the first one depends on the design distribution, i.e. the distribution of X_1^N, and reflects the property of adaptiveness of the estimator, while the second one does not depend on the design law, and is therefore weaker. The aggregation of these estimators described in Sect. 2.5 and 2.6 are also distribution-independent because the model risk is chosen a-priori.

In this section we present a modification which can be applied to any of the four estimators studied in Sect. 2.5 and 2.6. It consists in replacing the prior distribution π on the set of trees $\mathcal{C}_D$ by a *data-dependent* prior $\bar{\pi}$ to aggregate the primary estimators in order to get a better upper bound on the risk, which depends on the design distribution. This modification should be especially useful when the design distribution $P_N(X_1^N)$ is concentrated on a small subspace of $\mathcal{A}^D$, which is for instance the case in natural language modelling (see Sect. 2.9).

For clarity reasons we just show the construction of the estimator $Q_{\bar{\pi}}^N$ which is the modification of Q_π^N, the progressive mixture estimator which aggregates Laplace primary estimators and is defined in Sect. 2.5. Let us therefore formally define the density $Q_{\bar{\pi}}^N(y_N \,|\, x_1^N; y_1^{N-1})$ for any $z_1^N \in (\mathcal{X} \times \mathcal{Y})^N$.

Let $\mathcal{T}(x_1^N)$ denote the tree (in the sense of Sect. 2.2.2) whose vertices are the suffixes of the x_i's, i.e. :

$$\mathcal{T}(x_1^N) = \left\{ (x_i)_{-l}^0 \,:\, (i,l) \in [1,N] \times [0,D] \right\} \quad ,$$

and let $\bar{\mathcal{T}}(x_1^N)$ be the graph obtained by removing from $\mathcal{T}(x_1^N)$ the vertices with only one child and merging the two edges starting from a removed node (i.e. the edge toward its parent and the edge toward its single child). A *subtree* of the graph $\bar{\mathcal{T}}(x_1^N)$ is by definition any connex subgraph which contains the root λ as a vertex.

Example 2 *Figure 2(a) shows the graph $\bar{\mathcal{T}}(x_1^N)$ when $D = 4$ and the observation is $x_1^3 = (caba, aacc, cbcc)$. In that case the set of vertices of $\bar{\mathcal{T}}(x_1^3)$ is $\{\lambda, caba, cc, aacc, cbcc\}$. Two possible subtrees of $\bar{\mathcal{T}}(x_1^3)$ are shown on the right-hand parts of Figures 2(b) and 2(c), with respective sets of vertices $\{\lambda, caba, cc\}$ and $\{\lambda, cc, cbcc\}$.*

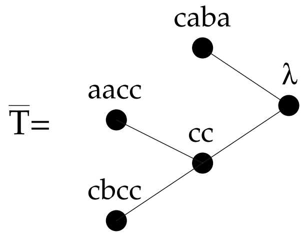
Let $\bar{\mathcal{C}}(x_1^N)$ be the set of subtrees of $\bar{\mathcal{T}}(x_1^N)$. For any $\bar{\mathcal{S}} \in \bar{\mathcal{C}}(x_1^N)$ the suffix functional $s_{\bar{\mathcal{S}}}$ is defined in the same way as when $\mathcal{S}$ is a classical tree (see Sect. 2.2.2). For any $\bar{\theta} \in \Sigma^{\bar{\mathcal{S}}}$ let $P_{\bar{\mathcal{S}},\bar{\theta}}$ denote the conditional probability distribution :

$$\forall (x,y) \in \mathcal{X} \times \mathcal{Y}, \qquad P_{\bar{\mathcal{S}},\bar{\theta}}(y \,|\, x) = \bar{\theta}_{s_{\bar{\mathcal{S}}}(x)}(y) \quad .$$

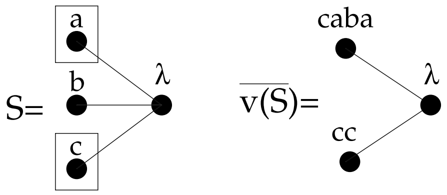
The counters $(\nu_n(s))_{s \in \bar{\mathcal{S}}}$ and $(\mu_n(s,y))_{(s,y) \in \bar{\mathcal{S}} \times \mathcal{Y}}$ are defined as before by (2.3). Therefore the distribution $Q_{\bar{\mathcal{S}}}^n(y_n \,|\, x_n, z_1^{n-1})$ can also be defined as before by (2.4).

Let $\bar{\pi}_{(x_1^N)}$ be the distribution on $\bar{\mathcal{C}}(x_1^N)$ defined by :

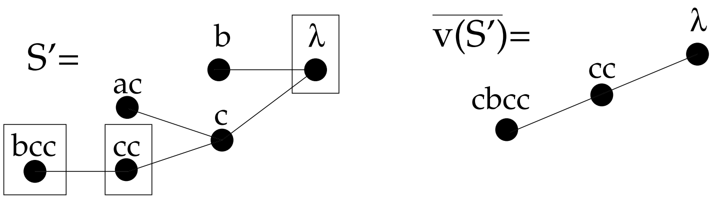
$$\forall \bar{\mathcal{S}} \in \bar{\mathcal{C}}(x_1^N), \qquad \bar{\pi}_{(x_1^N)}(\bar{\mathcal{S}}) = c^{|\bar{\mathcal{S}}|} \quad ,$$



(a) $\bar{\mathcal{T}}(x_1^3)$ for Example 2



(b) A tree $\mathcal{S}$, squares on its visited nodes, and corresponding subtree of $\bar{\mathcal{T}}(x_1^3)$ (see Example 3)



(c) Same as Figure 2(b) with a different tree $\mathcal{S}$ (see Example 3) where c is the real number which satisfies :

$$\sum_{\bar{\mathcal{S}} \in \bar{\mathcal{C}}(x_1^N)} c^{|\bar{\mathcal{S}}|} = 1 \quad .$$

Using this data-dependent prior $\bar{\pi}_{(x_1^N)}$ instead of the data-independent prior π in the definition of Q_π^N (see Sect. 2.6) we finally obtain a modified estimator $Q_{\bar{\pi}}^N$.

For any tree $\mathcal{S}$ in $\mathcal{C}_D$ recall that $v_N(\mathcal{S})$ denotes the set of visited nodes of $\mathcal{S}$, i.e. :

$$v_N(\mathcal{S}) = \{s \in \mathcal{S} \,:\, \mu_N(s) > 0\} \quad ,$$

and let $\overline{v(\mathcal{S})}$ be the smallest subtree $\bar{\mathcal{S}}$ of $\bar{\mathcal{T}}(x_1^N)$ such that for any $s \in v_N(\mathcal{S})$, there is a $s' \in \bar{\mathcal{S}}$ such that s is a suffix of s'.

Example 3 *As in Example 2 suppose that $D = 4$, $N = 3$ and $x_1^3 = (caba, aacc, cbcc)$. The left-hand parts of Figures 2(b) and 2(c) show two trees $\mathcal{S}$ and $\mathcal{S}'$ in $\mathcal{C}_D$. The squares around nodes on $\mathcal{S}$ and $\mathcal{S}'$ indicate the nodes which belong to $v_3(\mathcal{S})$ and $v_3(\mathcal{S}')$. The right-hand parts of the same Figures show the corresponding $\overline{v(\mathcal{S})}$ and $\overline{v(\mathcal{S}')}$*

We can now give an upper bound on the risk of the estimator $Q_{\bar{\pi}}^N$:

Theorem 7 *Let the size of the training set be the same as in Theorem 4. For any exchangeable distribution P_N on $(\mathcal{X} \times \mathcal{Y})^N$, the estimator $Q_{\bar{\pi}}^N$ using the data-dependent prior $\bar{\pi}$ satisfies :*

$$R_{P_N}\left(Q_{\bar{\pi}}^N\right) \leq \inf_{\mathcal{S} \in \mathcal{C}_D, \theta \in \Sigma^{|\mathcal{S}|}} \left\{ R_{P_N}\left(P_{\mathcal{S},\theta}\right) + \mathbf{E}_{P_N}\left(\left|\overline{v(\mathcal{S})}\right|\right) \frac{C_N}{N+1} \right\} \quad ,$$

with

$$C_N = \left(\sqrt{1 + \log(a)} + \sqrt{a-1}\right)^2 \left(1 + \frac{1}{N-2}\right) \quad .$$

Remark 8 *For any $\mathcal{S} \in \mathcal{C}_D$, $\left|\overline{v(\mathcal{S})}\right|$ is always smaller than $|\mathcal{S}|$. The upper bound in Theorem 7 is therefore smaller than the corresponding upper bound in Theorem 4. The difference can be large in cases when $P_N(X_1^N)$ is concentrated on a small subset of $\mathcal{A}^D$, because in that case $\bar{\mathcal{T}}(X_1^N)$ is a small subtree of $\bigcup_{i=0}^D \mathcal{A}^i$ with high probability.*

Remark 9 *The Laplace estimator for a given tree requires no modification because its risk is already bounded in terms of the number of visited nodes (see Theorem 1). Therefore only the prior π needs to be modified to become data-dependent.*

Remark 10 *Every tree $\mathcal{S} \in \mathcal{C}_D$ splits the data x_1^N into $|v(\mathcal{S})| = |v(\overline{v(\mathcal{S})})|$ clusters. The number of different separation of the data x_1^N by trees in $\mathcal{C}_D$ is therefore :*

$$\mathcal{N}(x_1^N) = \left|\{v\left(\overline{v(\mathcal{S})}\right) \;:\; \mathcal{S} \in \mathcal{C}_D\}\right| .$$

which is equal to $|\bar{\mathcal{C}}(x_1^N)|$ up to the number of trees with unvisited nodes. If we had chosen for $\bar{\pi}$ a uniform prior on $\bar{\mathcal{C}}(x_1^N)$ the model risk would have been of the order of $\mathbf{E}\log\mathcal{N}(X_1^N)$. The idea of computing an upper bound involving such a model risk instead of a model risk of order $\log|\mathcal{C}_D|$ (resulting from a uniform prior) is classical in statistical learning theory (see [86]), where the numbers $\mathcal{N}(x_1^N)$ and $\mathbf{E}\log(\mathcal{N}(X_1^N))$ are respectively known as the shatter coefficient *and the* annealed entropy.

Proof of Theorem 7:

The random tree $\bar{\mathcal{T}}(X_1^N)$ is invariant under permutation of the indices $[1, N]$. As a result, for any such tree $\bar{\mathcal{T}}$, the distribution $P_N\left(Z_1^N \,|\, \bar{\mathcal{T}}(X_1^N) = \bar{\mathcal{T}}\right)$ is exchangeable. On the event $\{\bar{\mathcal{T}}(X_1^N) = \bar{\mathcal{T}}\}$ the prior $\bar{\pi}$ is independent of the data and therefore Theorem 4 can be applied. As a result the following holds for any $\bar{\mathcal{S}} \in \bar{\mathcal{C}}(X_1^N)$ and $\bar{\theta} \in \Sigma^{\bar{\mathcal{S}}}$:

$$\begin{aligned}\mathbf{E}_{P_N\left(dZ_1^N \,|\, \bar{\mathcal{T}}(X_1^N)=\bar{\mathcal{T}}\right)} &\log \frac{1}{Q_{\bar{\pi}}^N(Y_N \,|\, X_N; Z_1^{N-1})} \\ &\leq \mathbf{E}_{P_N\left(dZ_1^N \,|\, \bar{\mathcal{T}}(X_1^N)=\bar{\mathcal{T}}\right)} \log \frac{1}{P_{\bar{\mathcal{S}},\bar{\theta}}(Y_N \,|\, X_N)} + |\bar{\mathcal{S}}| \frac{C_N}{N+1} \quad , \qquad (2.15)\end{aligned}$$

where C_N is defined in Theorem 7.

For any $\mathcal{S} \in \mathcal{C}^D$ and $\theta \in \Sigma^{\mathcal{S}}$ let $\bar{\theta} \in \Sigma^{\overline{v(\mathcal{S})}}$ be the parameter defined by :

$$\forall s \in \overline{v(\mathcal{S})}, \qquad \bar{\theta}_s = \theta_{s'} \ ,$$

where s' is the longest visited suffix of s in $\mathcal{S}$. For any $i \in [1, N]$ this definition leads to :

$$\begin{aligned}P_{\overline{v(\mathcal{S})},\bar{\theta}}(y_i \,|\, x_i) &= \bar{\theta}_{s_{\overline{v(\mathcal{S})}}(x_i)}(y_i) \\ &= \theta_{s'_i}(y_i) \ ,\end{aligned}$$

where s'_i is the longest visited suffix of $s_{\overline{v(\mathcal{S})}}(x_i)$ in $\mathcal{S}$. But $s_{\overline{v(\mathcal{S})}}(x_i)$ is by definition a suffix of x_i thus s'_i must also be a suffix of x_i. The largest suffix of x_i in $\mathcal{S}$ is $s_{\mathcal{S}}(x_i)$, which is by definition a suffix of $s_{\overline{v(\mathcal{S})}}(x_i)$. This shows that $s'_i = s_{\mathcal{S}}(x_i)$, and therefore :

$$\forall i \in [1, N], \qquad P_{\mathcal{S},\theta}(y_i \,|\, x_i) = P_{\overline{v(\mathcal{S})},\bar{\theta}}(y_i \,|\, x_i) \ .$$

The parameter $\bar{\theta}$ only depends on Z_1^N through $\bar{\mathcal{T}}$ and therefore we can integrate this equality to get :

$$\mathbf{E}_{P_N\left(dZ_1^N \mid \bar{\mathcal{T}}(X_1^N)=\bar{\mathcal{T}}\right)} \log \frac{1}{P_{\mathcal{S},\theta}(Y_N \mid X_N)} = \mathbf{E}_{P_N\left(dZ_1^N \mid \bar{\mathcal{T}}(X_1^N)=\bar{\mathcal{T}}\right)} \log \frac{1}{P_{\overline{v(\mathcal{S})},\bar{\theta}}(Y_N \mid X_N)} .$$

From (2.15) we deduce that for any $\bar{\mathcal{T}}$, $\mathcal{S}$ and θ the following holds :

$$\mathbf{E}_{P_N\left(dZ_1^N \mid \bar{\mathcal{T}}(X_1^N)=\bar{\mathcal{T}}\right)} \log \frac{1}{Q_{\bar{\pi}}^N(Y_N \mid X_N; Z_1^{N-1})} \leq \mathbf{E}_{P_N\left(dZ_1^N \mid \bar{\mathcal{T}}(X_1^N)=\bar{\mathcal{T}}\right)} \log \frac{1}{P_{\mathcal{S},\theta}(Y_N \mid X_N)} + |\overline{v(\mathcal{S})}| \frac{C_N}{N+1} .$$

Taking the expectation of this inequality with respect to P_N yields the upper bound in Theorem 7. □

2.8 Implementation for the aggregation using a Gibbs estimator

In this section we show how the estimator $G_{\pi,\beta}^N(Y_N \mid X_N; Z_1^{N-1})$ using the Gibbs estimator to aggregate Laplace estimators (see Sect. 2.6) can be computed using a recursive algorithm in the spirit of the Context Tree Weighting algorithm ([91]). The construction we present can be adapted to the other estimators studied in this paper.

2.8.1 Exact computation

Let $\mathcal{T}_D = \bigcup_{i=0}^{D} \mathcal{A}^i$ be the *context tree* of depth D, and for every $z_1^N \in (\mathcal{X} \times \mathcal{Y})^N$ let the following counters be attached to the nodes of the context tree, i.e. $\forall (s,y) \in \mathcal{T}_D \times \mathcal{Y}$:

$$\begin{cases} \mu_T(s,y) &= \sum_{i=1}^{K} \mathbf{1}\left(s \text{ is a suffix of } x_i \text{ and } y_i = y\right) \quad , \\ \mu_V(s,y) &= \sum_{i=K+1}^{N-1} \mathbf{1}\left(s \text{ is a suffix of } x_i \text{ and } y_i = y\right) \quad , \\ \mu_*(s,y) &= \mathbf{1}\left(s \text{ is a suffix of } x_N \text{ and } y_N = y\right) \quad , \\ \nu_T(s) &= \sum_{y \in \mathcal{Y}} \mu_T(s,y) \quad . \end{cases}$$

The subscripts T and V refer to the training set and the validation set respectively. Using these counters we can define the following functions attached to each node $s \in \mathcal{T}_D$, and defined for any subset $\mathcal{N} \subset \mathcal{Y}$ and $\xi \in \{0,1\}$:

$$w_{\mathcal{N}}^{(\xi)}(s) \overset{def}{=} c_D \prod_{y \in \mathcal{Y}} \left(\frac{\mu_T(s,y) - \sum_{i \in \mathcal{N}} \mu_T(is,y) + 1}{\nu_T(s) - \sum_{i \in \mathcal{N}} \nu_T(is) + a} \right)^{\beta \left(\mu_V(s,y) - \sum_{i \in \mathcal{N}} \mu_V(is,y) \right) + \xi \mu_*(s,y)} .$$

For any $\mathcal{S} \in \mathcal{C}_D$ and $s \in \mathcal{S}$ let :

$$\mathcal{N}_{\mathcal{S}}(s) = \{ i \in \mathcal{A} : is \in \mathcal{S} \} \quad .$$

Let now $\gamma^{(\xi)}$ be defined recursively on $\mathcal{T}_D$ for $\xi = \{0,1\}$ by the formula :

$$\begin{cases} \gamma^{(\xi)}(s) = w_{\emptyset}^{(\xi)}(s) & \text{if } l(s) = D \ , \\ \gamma^{(\xi)}(s) = \sum_{\mathcal{N} \subset \mathcal{Y}} w_{\mathcal{N}}^{(\xi)} \prod_{i \in \mathcal{Y} \setminus \mathcal{N}} \gamma^{(\xi)}(is) & \text{otherwise} \ . \end{cases} \tag{2.16}$$

The following Lemma shows that $\gamma^{(\xi)}(s)$ can be seen as a tensorization of a sum over all subtrees with root s :

Lemma 3

$$\forall (s,\xi) \in \mathcal{T}_D \times \{0,1\}, \qquad \gamma^{(\xi)}(s) = \sum_{\mathcal{S} \in \mathcal{C}_{D-d}} \left(\prod_{s' \in \mathcal{S}} w_{\mathcal{N}_{\mathcal{S}}(s')}(s's) \right) \quad ,$$

and the following result gives an effective way of computing the estimator $G^N_{\pi,\beta}(Y_N \,|\, X_N; Z_1^{N-1})$:

Proposition 1

$$\forall z_1^N \in (\mathcal{X} \times \mathcal{Y})^N, \qquad G^N_{\pi,\beta}(y_N \,|\, x_N; z_1^{N-1}) = \frac{\gamma^{(1)}(\lambda)}{\gamma^{(0)}(\lambda)} \quad .$$

Proof of Lemma 3: We prove the result by backward induction on $l(s)$. The property is obvious for $l(s) = D$ by definition of $\gamma^{(\xi)}(s)$ in that case. Suppose it is true for any $s' \in \mathcal{T}_D$ such that $l(s') = d+1$, and let a string $s \in \mathcal{T}_D$ of length $l(s) = d$. Then we get :

$$\begin{aligned} \gamma^{(\xi)}(s) &= \sum_{\mathcal{N} \subset \mathcal{Y}} w_{\mathcal{N}}^{(\xi)} \prod_{i \in \mathcal{Y} \setminus \mathcal{N}} \gamma^{(\xi)}(is) \\ &= \sum_{\mathcal{N} \subset \mathcal{Y}} w_{\mathcal{N}}^{(\xi)} \prod_{i \in \mathcal{Y} \setminus \mathcal{N}} \left[\sum_{\mathcal{S} \in \mathcal{C}_{D-d-1}} \left(\prod_{s' \in \mathcal{S}} w_{\mathcal{N}_{\mathcal{S}}(s')}(s'is) \right) \right] \\ &= \sum_{\mathcal{S} \in \mathcal{C}_{D-d}} \left(\prod_{s' \in \mathcal{S}} w_{\mathcal{N}_{\mathcal{S}}(s')}(s's) \right) \ .\square \end{aligned}$$

Proof of Proposition 1:

It is easy to check the following equality for any $\mathcal{S} \in \mathcal{C}_D$, $z_1^N \in (\mathcal{X} \times \mathcal{Y})^N$ and $\xi \in \{0,1\}$, using the definition of $\pi(\mathcal{S})$ and of the Laplace estimator $Q_{\mathcal{S}}^{K+1}$:

$$\pi(\mathcal{S}) \prod_{i=K+1}^{N-1} Q_{\mathcal{S}}^{K+1}(y_i \,|\, x_i, z_1^K)^{\beta} Q_{\mathcal{S}}^{K+1}(y_N \,|\, x_N, z_1^K)^{\xi} = \prod_{s \in \mathcal{S}} w_{\mathcal{N}_{\mathcal{S}}(s)}^{(\xi)}(s) \ .$$

As a result the estimator $G_{\pi,\beta}^N$ can be expressed as follows :

$$\forall z_1^N \in (\mathcal{X} \times \mathcal{Y})^N, \qquad G_{\pi,\beta}^N(y_N \,|\, x_N; z_1^{N-1}) = \frac{\sum_{\mathcal{S} \in \mathcal{C}_D} \left(\prod_{s \in \mathcal{S}} w_{\mathcal{N}_{\mathcal{S}}(s)}^{(1)}(s) \right)}{\sum_{\mathcal{S} \in \mathcal{C}_D} \left(\prod_{s \in \mathcal{S}} w_{\mathcal{N}_{\mathcal{S}}(s)}^{(0)}(s) \right)} \ .$$

Proposition 1 is a direct consequence of this equality and Lemma 3. □

2.8.2 Approximation by model selection

The implementation suggested by Prop. 1 using the functionals $\gamma^{(\xi)}$ involves the computation of a sum over $\mathcal{N} \subset \mathcal{A}$ at every node (see (2.16)). The number of such subsets $\mathcal{N}$ being equal to 2^a the actual computation of this sum might be unfeasible is a is too large.

As an alternative one can observe that the estimator $G_{\pi,\beta}^N$ is a mixture of Laplace estimators :

$$G_{\pi,\beta}^N = \sum_{\mathcal{S} \in \mathcal{C}_D} \rho(\mathcal{S}) Q_{\mathcal{S}}^{K+1} \ ,$$

and that this mixture should usually be unimodal in the space of conditional distributions, by construction of the Gibbs estimator. As a result an approximation of $G_{\pi,\beta}^N$ is the Laplace estimator corresponding to the tree with highest posterior probability, i.e. :

$$G_{\pi,\beta}^{(est.)}(Y_N \,|\, X_N; Z_1^{N-1}) = Q_{\overline{\mathcal{S}}(Z_1^{N-1})}^{K+1}(Y_N \,|\, X_N; Z_1^K) \ ,$$

with

$$\begin{aligned} \overline{\mathcal{S}}(z_1^{N-1}) &= \arg\max_{\mathcal{S} \in \mathcal{C}_D} \rho(\mathcal{S}) \\ &= \arg\max_{\mathcal{S} \in \mathcal{C}_D} \left\{ \pi(\mathcal{S}) \prod_{i=K+1}^{N-1} Q_{\mathcal{S}}^{K+1}(y_i \,|\, x_i, z_1^K)^{\beta} \right\} \\ &= \arg\max_{\mathcal{S} \in \mathcal{C}_D} \left\{ |\mathcal{S}| \frac{\log c_D}{\beta} + \log \prod_{i=K+1}^{N-1} Q_{\mathcal{S}}^{K+1}(y_i \,|\, x_i, z_1^K) \right\} \ . \end{aligned} \tag{2.17}$$

This formulation shows that $G^{(est.)}_{\pi,\beta}$ is obtained by a *penalized maximum likelihood* selection procedure, where the penalization for the log-likelihood of a model $\mathcal{S}$ is $\kappa = (\log c_D)/\beta$ per node.

The implementation of this model selection procedure can follow the spirit of the implementation of the mixture :

- For any subset $\mathcal{N} \in \mathcal{Y}$ and $s \in \mathcal{T}_D$ let :

$$\overline{w}_{\mathcal{N}}(s) \stackrel{def}{=} \kappa + \sum_{y\in\mathcal{Y}} \left(\mu_V(s,y) - \sum_{i\in\mathcal{N}} \mu_V(is,y)\right) \log \frac{\mu_T(s,y) - \sum_{i\in\mathcal{N}} \mu_T(is,y) + 1}{\nu_T(s) - \sum_{i\in\mathcal{N}} \nu_T(is) + a} \ .$$

- Let $\overline{\gamma}$ be recursively defined on $\mathcal{T}_D$ by :

$$\begin{cases} \overline{\gamma}(s) = \overline{w}_{\emptyset}(s) & \text{if } l(s) = D \ , \\ \overline{\gamma}(s) = \max\limits_{\mathcal{N}\subset\mathcal{A}} \left\{ \overline{w}^{(\xi)}_{\mathcal{N}} + \sum\limits_{i\in\mathcal{Y}\setminus\mathcal{N}} \overline{\gamma}(is) \right\} & \text{otherwise} \ . \end{cases}$$

- For every $s \in \mathcal{T}_D$ if the nodes in the selected subset $\mathcal{N}$ used to compute $\overline{\gamma}(s)$ are marked, then $\overline{\mathcal{S}}$ is the largest tree made of marked nodes.

Remark 11 *An other possibility to approximate the estimator $G^N_{\pi,\beta}$ would be to use a Monte-Carlo Markov Chain simulation to approximate the mixture (see [16] for a discussion in the framework of decision trees).*

2.9 Experiments and natural language processing applications

As an application for the estimators studied in this paper, we show here how they can be used to model texts written in natural language, and give results from a text clustering experiment based on these statistical models.

For a given alphabet $\mathcal{A}$, a text T written in natural language (e.g. in English or Japanese) is a string which can be parsed into a series of letters. One can think of $\mathcal{A}$ as the letters of the alphabet $\{a, b, \dots, z\}$, the ASCII symbols set, a dictionary of words, or whatever set of symbols in terms of which the text can be represented as a sequence $(t_1, \dots, t_{|T|})$ with $\forall i \in [1, |T|], t_i \in \mathcal{A}$.

For a given $D < |T|$, let $(X, Y) \in \mathcal{A}^D \times \mathcal{A}$ be the r.v. obtained by randomly choosing an index $i \in [1, |T| - D]$ uniformly and setting

$$\begin{cases} X & = t_i \dots t_{i+D-1} \ , \\ Y & = t_{i+D} \ . \end{cases}$$

For a given N, let us consider the statistical experiment that consist in sampling N i.i.d variables $(X_i, Y_i)_{i\in[1,N]}$ according to this common law. This experiment can be used to train any regression model to infer Y from X, which gives a representation of the initial text as a stochastic model. Note that the initial text is deterministic, and that the random nature of the variables comes from the sampling.

2.9.1 Tuning the parameters

As an example let us consider the model selection algorithm described in Section 2.8.2. Equation (2.17) shows that the "cost" of adding a node to a model is $\log(c_D)/\beta$, which is a parameter we can try to optimize for a given problem. Note that if we were trying to compute the actual estimator which is a mixture of models, for instance using Monte-Carlo simulations, two different parameters could be varied : c_D and β, which influence the shape of the prior and the speed of learning from examples respectively.

A second parameter can be optimized : K/N, which is related to the relative sizes of the estimation and the validation sets.

In order to observe the effect of these two parameters, Figures 2 and 3 show results of an experiment carried out from the text "Far from the madding crowd" from T. Hardy, which is the file 'book1' of the Calgary corpus[1](used in [10]). The text (in English) was parsed into a sequence of characters using the alphabet $\mathcal{A} = \{a, b, \dots, z, O\}$ where O represents anything that is not a letter. The estimator was then trained on i.i.d. samples of size $N = 20,000$ with varying K/N and $\log(c_D)/\beta$, and its likelihood was computed on a test set made of $5,000$ new i.i.d. samples. Figure 2 shows the per-sample log-likelihood for varying $\log(c_D)/\beta$ and K/N, and figure 3 shows for clarity purpose the same curve for $K/N = 0.7$ being fixed.

For any K/N, the value $\log(c_D)/\beta = 0$ corresponds to the classical maximum likelihood estimator. Negative values correspond to negative penalizations and therefore favor large models. Positive values are more natural and correspond to penalizing more large models than small ones.

For $\log(c_D)/\beta < -3$, the likelihoods of the models on the test set are very low : this is the classical phenomenon of overfitting, that is favored by the negative penalization. In this region indeed the selected model appears to be too large for its parameters to be accurately estimated. As $\log(c_D)/\beta$ increases to 0, the performance increases and peaks at a value a bit larger than zero, which corresponds to the optimal penalization for the particular unknown probability and the particular sizes considered. Larger penalization values decrease the performance of the selected model on the test set because its dimension becomes too small. In that case,

[1] Available at `ftp://ftp.cpsc.ucalgary.ca/pub/projects/text.compression.corpus/`

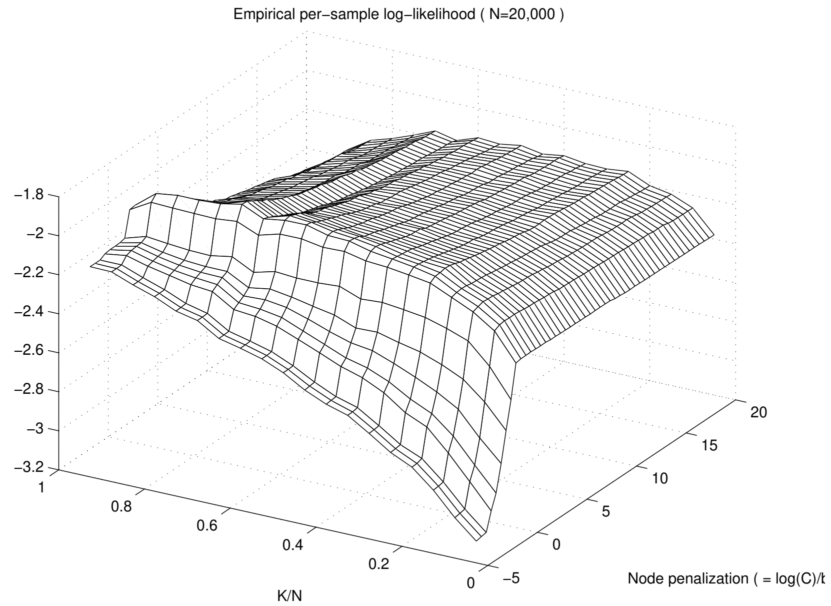


Figure 2: Log-likelihood with N=20,000 for various K and $\log(c_D)/\beta$

indeed, the gain in the variance term due to decreasing the number of parameters to estimate does not balance the increase of the bias term which corresponds to the distance between P and the selected model.

Figure 2 also shows that for a given penalization there exists an optimal choice of division between the training set and the validation set, which corresponds to the balance between training the Laplace estimators and choosing the best model : it is better to have a training set a bit larger than the validation set. Naturally, as the penalization increases, the optimal K increases too, because increasing the penalization means giving less importance to each validation sample.

2.9.2 Comparison with other models

Many other statistical models can be used to characterize the relation between X and Y. In particular, the so-called N-gram models are widely known and used in natural language processing to characterize sequences of characters (e.g. for character recognition purposes) or words (e.g. for speech recognition purposes). In a N-gram model, the distribution of Y is supposed to depend on the suffix of length $N-1$ of X, with N being fixed.

Thus N-grams are particular regression trees, i.e. complete trees of depth $N-1$. The difficulties arise when one wants to estimate the N^D distributions of Y from a finite training corpus. An adaptive approach as the one described in this paper is better at balancing the

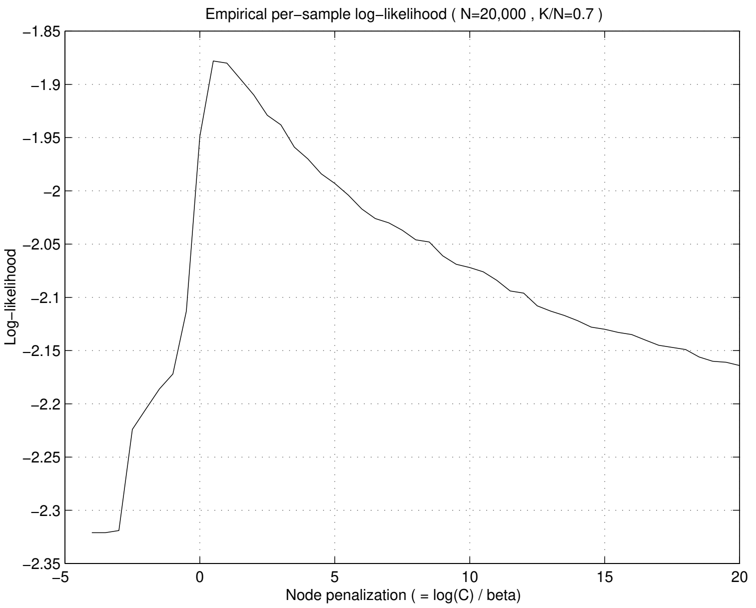


Figure 3: Log-likelihood with N=20,000 for $\frac{K}{N} = 0.7$ and various $\log(c_D)/\beta$

complexity of the model and the precision of the estimation which basically depends on the size of the training corpus.

As an example, figure 4 shows the log-likelihood of different models trained on i.i.d. samples of growing size (between 100 and $10,000$) and tested on an i.i.d. sample of size $5,000$. The models tested are:

- N-gram models for $N = 1, 2, 3, 4$, with classical non-adaptative Laplace estimators.
- The aggregation using a Gibbs estimator, with classical non-adaptative Laplace estimators.
- The aggregation using a Gibbs estimator, with adaptive Laplace estimators.

Following the results of the first experiment, the parameters for aggregated estimator were set to $\log(c_D)/\beta = 0.5$ and $K/N = 0.65$

This experiment shows that the adaptive regression model is more efficient than all N-gram models for any training set size. It also shows the improvement gained with the introduction of the adaptive Laplace estimator and the adaptive probability on the model space. Indeed, it is clear that the support of the distributions of Y are often smaller than the whole alphabet (e.g. the character following the letter 'q' should almost always be a 'u' or a space), and that the strings X observed only form a small subset of the set of sequences of D characters.

Figure 4: Comparison with other models

2.9.3 Unsupervised text clustering

While the distribution of a letter following a string might have straightforward applications as such (e.g. for disambiguation purpose in optical character recognition systems), the estimator we study can be considered more generally as a way of *representing* a text because it is able to 'learn' various statistical features very quickly.

As an example it can be used to define and measure a notion of *distance* between texts. Indeed let T_1 and T_2 be two given texts that one wants to compare. Using them to generate statistical experiments, it is natural to say they are close to each other if *the model that has been trained to explain the first statistical experiment is good at explaining the second one*, and far from each other otherwise.

This can be quantified as follows. Suppose each text is used to generate a statistical experiment, on which an estimator is trained. This generates two models $Q_1(Y \mid X)$ and $Q_2(Y \mid X)$ which can be used afterwards to compute the likelihood of any sample $(x_i, y_i)_{i=1}^N$. In particular one can define a pseudo-distance between the two texts with the following formula:

$$d(T_1, T_2) = \log \frac{Q_1(\exp_1)}{Q_1(\exp_2)} + \log \frac{Q_2(\exp_2)}{Q_2(\exp_1)} \quad , \tag{2.18}$$

where $\exp_i$ means the experiment that consists in sampling N i.i.d. pairs (x, y) from text T_i. This pseudo-distance is symmetric and satisfies $d(T, T) = 0$ for any text T.

Let now a set of p texts $\{T_1, \ldots, T_p\}$ be given. The unsupervised text clustering problem is the problem of grouping these texts into a number of categories according to their similarities. Most existing clustering algorithms require a distance-like functional to be defined between any two elements to be clustered, that can be the pseudo-distance defined by equation (2.18).

To illustrate this we took a series of 8 books from each of which we extracted 5 texts, and computed the distance between any two of the resulting 40 texts (see table I).

Text Number	Extracted from
1-5	Wintson Churchill (*The Crossing*)
6-10	Joseph Conrad (*The Arrow of gold*)
11-15	Arthur Conan Doyle (*The hound of the Baskervilles*)
16-20	Karl Marx (*Manifesto of the communist party*)
21-25	Baruch Spinoza (*Political treatise*)
26-30	Jonathan Swift (*Gulliver's travel*)
31-35	Francois Marie Arouet Voltaire (*Candide*)
36-40	Virginia Woolf (*Night and day*)

Table I: Text database

Each text was 12,000 characters long and was used to generate three files by i.i.d. sampling. The first two files (8,000 and 4,000 samples) were used as estimation and validation set, while the third file (5,000 samples) was used as a test set to measure the likelihoods used in equation (2.18). The parameter $\log(c_D)/\beta$ was set to 0.5.

Figure 5 is a typical profile of distances between one text (here the text number 23, extracted from Spinoza's *Political Treaty*) and all other texts. It shows that the distance with the four texts extracted from the same book (i.e. texts 21, 22, 24 and 25) are clearly smaller than the distances with the rest of the database, and that it could "recognize" the similarity within the texts extracted from the same book.

On figure 6 we plotted a 'o' as soon as the distance between two texts was smaller than 1.03. Clusters corresponding to the books already appear with this naive thresholding method.

One should remark that no dictionary or preprocessing of the text was used. The usual way of representing a text as a "bag of words" in the literature about natural language processing

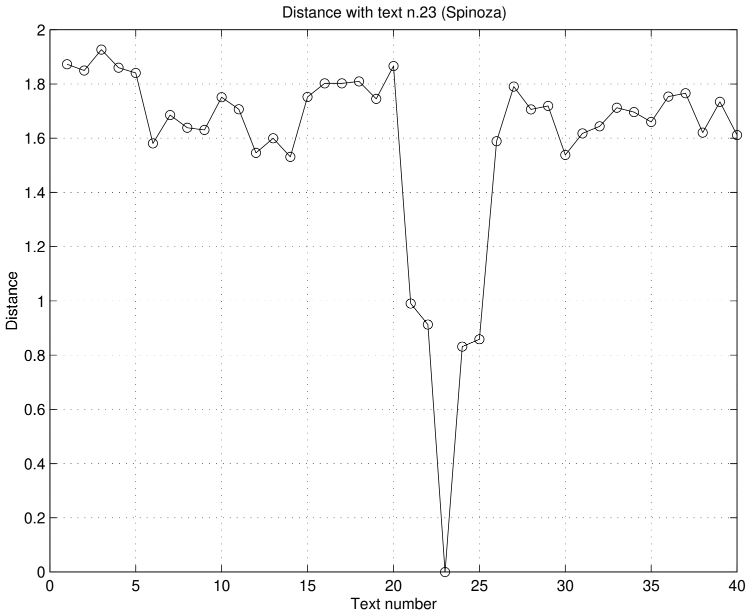


Figure 5: Distance between text n. 23 (Spinoza) and the other texts

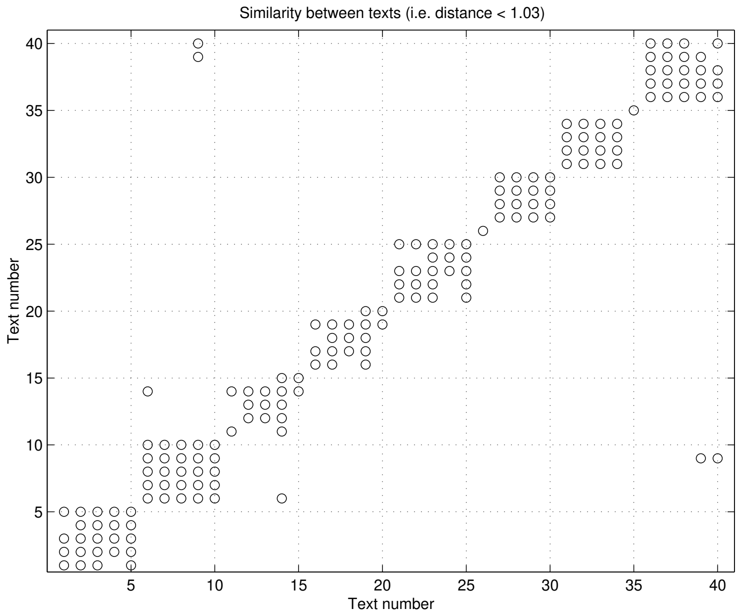


Figure 6: Similarity between texts

is limited as far as statistical estimation is concerned because the number of possible words is much larger than the size of the text itself. On the other hand, we experimented models based on characters only which lead to less risky estimations and encouraging results.

2.10 Conclusion

We presented a family of statistical estimators of a conditional distribution and proved upper bounds on their risk. The main characteristic of these estimators is their ability to find a good trade-off between the bias of different models and the risk of their estimation for a given number of observations.

Such estimators are interesting in cases when the "real" law P_N is complicated, but progressively approximated by models of increasing dimensions. As an example we considered the issue of modelling texts written in natural language, for which classical Markovian models like N-grams are limited in depth because of the size of the training corpus that is needed. In spite of the simplicity of our models encouraging experimental results make us believe that important improvement could be obtained by carefully designing pertinent models for a particular application while keeping in mind the necessity of efficient statistical estimations.

Chapter 3

Double mixture and universal inference

Abstract

Given a family of finite dimensional statistical models and a finite number of observations of a random variable, we show how to build a "double mixture" estimator for the density of the random variable whose risk in terms of Kullback-Leibler divergence has a sharp bound compared to the risk of the best model in the family. This estimator is a mixture of model estimators which are themselves mixtures in the continuous parameter spaces of the corresponding models.

The idea of using double mixtures has been studied for a long time in the field of universal compression in coding theory but we highlight the fundamental differences between our statistical estimator and "twice-universal" coding algorithm, due to the difference in the criteria to optimize.

3.1 Introduction

The problem of estimating the probability distribution P of a random variable X on a space $\mathcal{X}$ from a finite number of i.i.d. observations $X_1, \ldots, X_n$ is a central but difficult problem in statistics. As pointed out by Vapnik ([86]) it is generally ill-posed when no assumption is made on P. In the real world, however, the statistician usually knows nothing in advance about P. In that case a natural approach consists in building a family of parametric models $\left\{P_{\theta_m} \in \mathcal{M}_+^1(\mathcal{X}); \theta_m \in \Theta_m \subset \mathbb{R}^{D_m}, m \in \mathcal{M}\right\}$ and considering the minimization problem

$$\inf_{m \in \mathcal{M}} \inf_{\theta_m \in \Theta_m} l\left(P, P_{\theta_m}\right), \tag{3.1}$$

where l is a loss function between probability distributions. In other words, rather than trying to guess P, the statistician looks for the most informative projection of P on the most reasonable model Θ_m.

An estimator for this problem of distribution estimation is a measurable mapping $\hat{P}$ from $\mathcal{X}^n$ to $\mathcal{M}_+^1(\mathcal{X})$. The performance of such an estimator with respect to a true distribution P is usually measured in terms of its average loss, also called *risk*:

$$\mathbf{E}_{P^{\otimes n}\left(dX_1^N\right)} l\left(P, \hat{P}(X_1^N)\right).$$

For an estimator with value within a particular model (parametric estimation), this risk can often be expressed or at least upper bounded by the sum of two terms:

- a *bias* term which represents the distance between the actual probability P and its projection P_m on the particular model;
- a *fluctuation* term which represents the difficulty of estimating P_m.

Usually the larger the model the smaller the bias, but the larger the fluctuation risk. A natural way of solving the estimation problem is to decompose it into two stages : first build good estimators $\hat{P}_m$ for every model $m \in \mathcal{M}$ and then select one model $\hat{m}$ with the lowest total risk. With this approach the final estimator $\hat{P}_{\hat{m}}$ is the estimator associated with the model supposed to realize the best trade-off between bias and fluctuation. This philosophy is the starting point of many techniques in the field of *model selection*, to which a huge amount of literature has been devoted. As we won't further develop this approach let us just mention typical references including works by Akaike ([1]), Mallows ([56]), Schwarz ([76]), Rissanen ([66]), Barron and al. ([6]) and Vapnik ([86]).

Model selection is not the only way to deal with problem (3.1). An other approach is gaining attention in statistical estimation: the idea of *model mixture*. This idea led to remarkable theoretical and experimental results in coding theory for compression purpose where mixture codes ([30]) are known to be universal with respect to a class of encoders, under quite general assumptions.

As far as statistical estimation is concerned, every Bayesian estimator can be considered as a mixture estimator. While these estimators are optimal for the Bayesian risk theoretical results concerning their performance in the worst case setting are difficult to obtain. Barron

([5]) and Barron and Yang ([8]) considered a Cesaro mean of Bayesian estimators to derive minimax density estimators for non parametric density classes. More recently Catoni considered an equivalent estimator together with a half-sample trick to deal with parametric density classes ([23], [24]), and showed that a thermalized version of the Bayesian estimator ([22]) could approach the minimax risk under very general assumption.

In a recent paper ([87]) we applied Catoni's estimators in the framework of regression. We showed how to build a mixture estimator $\hat{P}_w = \sum_{i \in \mathcal{I}} w(i) \hat{P}_{m_i}$ where the weights w as well as the estimators $\hat{P}_m$ are built from the observation, and obtain a universal risk bound. This approach involved a split of the observations into an estimation set used to build the estimators $\hat{P}_m$ for every $m \in \mathcal{M}$ and a validation set used to compute the weights $w(i)$ of each estimator.

In this paper we go one step further in this mixing approach. After observing that the estimators $\hat{P}_m$ for every model can be mixtures on the continuous parameter set themselves (think of the Laplace estimator for a Bernoulli distribution for instance), we show how it is possible to carry out a *double mixture* in one stage by considering the larger parameter set $\{(m, \theta_m), m \in \mathcal{M}, \theta_m \in \Theta_m\}$. This means in particular that the observations are not split into two sets any more, and that the same observations are used to estimate continuous parameters and model structure in the same time. The idea of a double mixture finds its roots in coding theory where double mixture codes have given very interesting results ([41], [71]). An important source of inspiration was the work of Willems, Shtarkov and Tjalkens concerning the context tree weighting algorithm ([91] , [92]), together with Catoni's Gibbs estimator ([22]) which can be used to mix discrete as well as continuous parameters.

This paper is organized as follows. After setting up the general regression framework which will be used afterward in section 3.2 we state the main result of this paper in section 3.3 (Theorem 8) whose proof is postponed to section 3.6 because of its length. Section 3.4 is a comparison between the estimator we propose for statistical estimation and "universal" estimators used in coding theory, and section 3.5 presents a particular application of our estimator for string analysis, with an efficient implementation. We refer to a previous works ([87]) for suggestions on how to use such models for natural language processing applications.

3.2 Notations and framework

In this section we present the regression framework together with general notations which will be used within the paper.

Let $(\mathcal{X}, \mathcal{B}_1)$ be a measurable space and $(\mathcal{Y}, \mathcal{B}_2)$ be a *finite* measurable space endowed with the discrete σ-algebra. We note α the size of the set $\mathcal{Y}$. The goal of statistical modeling is to predict the value of a variable $Y \in \mathcal{Y}$ from an observation $X \in \mathcal{X}$. The set $\mathcal{Y}$ being finite this covers in particular the problem of categorization. However we focus on estimating the conditional law of Y knowing X, and not on the design of a classifier. In particular the criterion we will use is a measure of the difference between laws and not the number of categorization errors. Note that the variable X can be almost anything.

To model the random nature of X and Y we suppose that a family of unknown exchangeable probability distributions is given :

$$\forall N \in \mathbb{N} \qquad P_N \in \mathcal{M}_+^1\left((\mathcal{X} \times \mathcal{Y})^N, (\mathcal{B}_1 \otimes \mathcal{B}_2)^{\otimes N}\right),$$

and we let $\{(X_i, Y_i) = Z_i; i = 1, \dots, N\}$ be the canonical process.

One can for instance think of P_N as a product measure $P^{\otimes N}$ with P being a probability on $(\mathcal{X} \times \mathcal{Y}, \mathcal{B}_1 \otimes \mathcal{B}_2)$, if the observations are supposed to be i.i.d. However we will only use the weaker assumption that P_N is *exchangeable*, i.e. that for any permutation σ of $\{1, \dots, N\}$ and any $A \in (\mathcal{B}_1 \otimes \mathcal{B}_2)^N$,

$$P_N\left(Z_1^N \in A\right) = P_N\left((\sigma Z)_1^N \in A\right),$$

where σZ is the exchanged process

$$(\sigma Z)_i = Z_{\sigma(i)}, \qquad i = 1, \dots, N.$$

The property of being exchangeable is more general than the property of being a product measure, and it covers more situations which can happen in the real world, e.g. random splitting of the observations into different sets, or sampling from a finite set without replacement.

Within this framework the observation is Z_1^{N-1} and the goal is to estimate the unknown conditional distribution $P_N\left(dY_N | X_N; Z_1^{N-1}\right)$.

3.2.1 Finite context model

Without any further restriction on P_N the problem of density estimation based on empirical data can be ill-posed. Therefore we suppose a family of models is given. It will be used to approximate the unknown distribution.

Definition 2 *A model $m = (\mathcal{S}_m, s_m)$ consists of:*

- *a finite set* $\mathcal{S}_m = \{s_1, \ldots, s_{D_m}\}$. D_m *is called the dimension of the model.*
- *A measurable mapping* $s_m : (\mathcal{X}, \mathcal{B}_1) \to \mathcal{S}_m$*, which describes how the space* $\mathcal{X}$ *is partitioned according to the model.*

For any model $m \in \mathcal{M}$ *and* $x \in \mathcal{X}$*,* $s_m(x)$ *is called the* context of x with respect to the model m.

The goal of any model is to partition the space $\mathcal{X}$ into D_m categories through the mapping s_m, and to build a conditional distribution for Y which only depends on the category of X. Such a class of models includes in particular regression based on histograms, CART models ([17]) or representation of complex objects (e.g. images) using filtering of features extraction.

Finally we suppose given a countable family of such models $\mathcal{M} = \{m_i\}_{i \in \mathcal{I}}$ with $\mathcal{I}$ being a countable index set, as well as a prior probability distribution π on $\mathcal{I}$. The role of the prior distribution π which influences the performance of the final estimator will become clearer in the sequel.

The variable Y being discrete its distribution is a Bernoulli distribution characterized by a parameter of the α-dimensional simplex $\Sigma = \{\theta \in [0,1]^\alpha / \sum_{i=1}^{\alpha} \theta^i = 1\}$. Therefore any model m is associated with a parameter space $\Theta_m = \Sigma^{D_m}$ to define a family of conditional probability distributions with the following density:

$$\forall m \in \mathcal{M}, \forall \theta_m = \left(\theta_{s_1}, \ldots, \theta_{s_{D_m}}\right) \in \Theta_m, \forall (x,y) \in \mathcal{X} \times \mathcal{Y} \qquad p_{m,\theta_m}(y|x) = \theta^y_{s_m(x)}.$$

3.2.2 Problem

As we want to compare estimators based on different models we can not use a distance defined on the parameter space. In order to measure directly the distance between the true sample conditional distribution and the estimated one we use the conditional Kullback Leibler divergence (also called *conditional relative entropy*, see e.g. [29, p. 22]) which is an intrinsic and fundamental measure of risk defined for two probabilities P_1 and P_2 with densities p_1 and p_2 by :

$$\mathcal{K}\left(P_1(dY|X), P_2(dY|X)\right) = \mathbf{E}_{P_1(dX,dY)} \log \frac{p_1(y|x)}{p_2(y|x)}.$$

The model selection problem for the average Kullback risk is to solve approximately, knowing the sample Z_1^{N-1}, the minimization problem:

$$\inf_{m\in\mathcal{M},\theta_m\in\Theta_m}\mathbf{E}_{P_N\left(dZ_1^{N-1}\right)}\mathcal{K}\left(P_N\left(dY_N|X_N;Z_1^{N-1}\right),P_{m,\theta_m}\left(dY_N|X_N\right)\right),\tag{3.2}$$

where $\mathcal{K}(.,.)$ is the conditional Kullback Leibler divergence.

3.3 The double mixture estimator

The continuous parameter set associated with a model $m \in \mathcal{M}$ is $\Theta_m = \Sigma^{D_m}$. Let us define a probability distribution on this set as a product measure $\mu_m = \mu^{\otimes D_m}$ where μ is the Dirichlet distribution with parameter $1/2$ on Σ, i.e. the measure with the following density with respect to Lebesgue's measure $\lambda(d\theta)$:

$$\mu(d\theta)=\frac{1}{\sqrt{\alpha}}\frac{\Gamma\left(\frac{\alpha}{2}\right)}{\Gamma\left(\frac{1}{2}\right)^{\alpha}}.\prod_{i=1}^{\alpha}\frac{1}{\sqrt{\theta^i}}\lambda(d\theta).$$

This prior, also known as Jeffrey's prior, arises naturally in coding theory for compression purpose because it asymptotically maximizes Shannon's mutual information between an i.i.d. sample drawn according to a Bernoulli law of parameter θ and the parameter ([52], [27]). The reason why we use it here will appear in the computation of the performance of our double mixture estimator. Let us recall a formula that will be used frequently in the sequel:

$$\forall\lambda\in\left(\mathbb{R}^+\right)^{\alpha}\qquad\int_{\Sigma}\left(\theta^1\right)^{\lambda^1}\ldots\left(\theta^{\alpha}\right)^{\lambda^{\alpha}}\mu\left(d\theta\right)=\frac{\Gamma\left(\frac{\alpha}{2}\right)}{\Gamma\left(\frac{1}{2}\right)^{\alpha}}\frac{\prod_{i=1}^{\alpha}\Gamma\left(\lambda^i+\frac{1}{2}\right)}{\Gamma\left(\sum_{i=1}^{\alpha}\lambda^i+\frac{\alpha}{2}\right)}.\tag{3.3}$$

Based on these priors for the continuous parameters and on an arbitrary prior π on the model set $\mathcal{M}$, we can construct a double mixture estimator which takes the form of a Gibbs mixture, as defined by Catoni in [22]. In order to clarify the definition of this estimator it is convenient to introduce notations for the entropy of a Bernoulli model and the Kullback-Leibler divergence between two such models, respectively as:

$$\forall\theta\in\Sigma\qquad h(\theta)=-\sum_{y\in\mathcal{Y}}\theta^y\log\theta^y,$$
$$\forall\left(\theta_1,\theta_2\right)\in\Sigma^2\qquad d\left(\theta_1||\theta_2\right)=\sum_{y\in\mathcal{Y}}\theta_1^y\log\frac{\theta_1^y}{\theta_2^y}.$$

For any model $m \in \mathcal{M}$ let us also introduce the following random variables which are expressed in terms of Z_1^N (the dependency w.r.t. m is not indicated in order to simplify the notations, and because no ambiguity about which m these variables refer to should arise in the sequel):

$$\begin{aligned}
\forall (y,s) \in \mathcal{Y} \times \mathcal{S}_m \qquad & a_s^y = \sum_{j=1}^{N-1} \mathbf{1}(s_m(X_j) = s \text{ and } Y_j = y), \\
\forall (y,s) \in \mathcal{Y} \times \mathcal{S}_m \qquad & b_s^y = \mathbf{1}(s_m(X_N) = s \text{ and } Y_N = y), \\
\forall (y,s) \in \mathcal{Y} \times \mathcal{S}_m, \forall (\beta,\xi) \in \mathbb{R}^2 \qquad & \eta_s^y(\beta,\xi) = \beta a_s^y + \xi b_s^y, \\
\forall s \in \mathcal{S}_m, \forall (\beta,\xi) \in \mathbb{R}^2 \qquad & n_s(\beta,\xi) = \sum_{y \in \mathcal{Y}} \eta_s^y, \\
\forall s \in \mathcal{S}_m \qquad & \bar{\theta}_s(\beta,\xi) = \left(\frac{\eta_s^1(\beta,\xi)}{n_s(\beta,\xi)}, \dots, \frac{\eta_s^\alpha(\beta,\xi)}{n_s(\beta,\xi)} \right),
\end{aligned}$$

The reason for using these notations basically comes from the following equality used to express the thermalized likelihood of a sequence Z_1^N with respect to a particular model (m, θ_m):

$$\begin{aligned}
\left(\prod_{i=1}^{N-1} p_{m,\theta_m}(Y_i|X_i) \right)^{\beta} & p_{m,\theta_m}(Y_N|X_N)^{\xi} \\
&= \prod_{s \in \mathcal{S}_m} \prod_{y \in \mathcal{Y}} (\theta_s^y)^{\eta_s^y(\beta,\xi)} \\
&= \prod_{s \in \mathcal{S}_m} \exp \left[n_s(\beta,\xi) \sum_{y \in \mathcal{Y}} \bar{\theta}_s^y(\beta,\xi) \log \theta_s^y \right] \\
&= \prod_{s \in \mathcal{S}_m} \exp \left[-n_s(\beta,\xi) \left(h\left(\bar{\theta}_s(\beta,\xi)\right) + d\left(\bar{\theta}_s(\beta,\xi) || \theta_s\right)\right)\right].
\end{aligned} \tag{3.4}$$

Following these preliminaries we can now state the main theorem of this paper which contains the definition of the double mixture estimator as well as a universal bound for its risk:

Theorem 8 *Let*

$$\bar{\chi} = 24 + 8 \log \left(N + \frac{\alpha}{2} + 1 \right),$$

and $\beta > 0$ such that

$$\beta < \frac{1}{\bar{\chi}-1}\left(\sqrt{1+(\bar{\chi}-1)\left(2-\frac{\log\bar{\chi}}{\bar{\chi}}\right)\frac{\log\bar{\chi}}{\bar{\chi}}}-1\right)$$
$$\underset{N\to\infty}{\sim} \frac{\sqrt{2\log\log(N)}}{8\log(N)}.$$

For any exchangeable distribution P_N on $(\mathcal{X}\times\mathcal{Y})^N$, for any choice of prior probability distribution π on $\mathcal{M}$, the posterior Gibbs distribution ρ defined on the set

$$\{(m,\theta_m)\,, m\in\mathcal{M}, \theta_m\in\Theta_m\}$$

by

$$\rho\left(d\theta_m|m\right) \sim \prod_{s\in\mathcal{S}_m} \exp\left[-n_s(\beta,0)d\left(\bar{\theta}_s(\beta,0)||\theta_s\right)\right]\mu\left(d\theta_s\right),$$

and

$$\rho\left(m\right) \sim \pi\left(m\right)\prod_{s\in\mathcal{S}_m}\left\{\frac{\pi^{\alpha/2}}{\Gamma\left(\frac{\alpha}{2}\right)}\left(\frac{n_s\left(\beta,\beta\right)}{2\pi e}\right)^{\frac{\alpha-1}{2}}\right.$$
$$\left.\times\int_{\Sigma}\exp\left\{-n_s(\beta,0)\left[h\left(\bar{\theta}_s(\beta,0)\right)+d\left(\bar{\theta}_s(\beta,0)||\theta_s\right)\right]\right\}\mu\left(d\theta_s\right)\right\},$$

can be used to form the double mixture estimator

$$G_\beta^N\left(dY_N|X_N;Z_1^{N-1}\right) = \mathbf{E}_{\rho(m,d\theta_m)}P_{m,\theta_m}\left(dY_N|X_N\right)$$

which satisfies

$$\mathbf{E}_{P_N\left(dZ_1^{N-1}\right)}\mathcal{K}\left(P_N\left(dY_N|X_N;Z_1^{N-1}\right), G_\beta^N\left(dY_N|X_N;Z_1^{N-1}\right)\right)$$
$$\leq \inf_{m\in\mathcal{M},\theta_m\in\Theta_m}\left\{\mathbf{E}_{P_N\left(dZ_1^{N-1}\right)}\mathcal{K}\left(P_N\left(dY_N|X_N;Z_1^{N-1}\right), P_{m,\theta_m}\left(dY_N|X_N\right)\right)\right.$$
$$\left.+\frac{1}{\beta N}\left(D_m\frac{\alpha-1}{2}+\log\frac{1}{\pi\left(m\right)}+C_N\left(m\right)\right)\right\},$$

with

$$C_N(m) = \mathbf{E}_{P_N}\sum_{s\in\mathcal{S}_m}\left(\frac{\alpha^2}{4n_s(\beta,\beta)}+\frac{\alpha}{4n_s\min_i(\bar{\theta}^i(\beta,\beta))+2}\right).$$

This theorem is proved in section 3.6.

Remark 12 *The double mixture estimator can be expressed in the following way:*

$$
\begin{aligned}
G_\beta^N\left(dY_N|X_N;Z_1^{N-1}\right) &= \mathbf{E}_{\rho(m,d\theta_m)} P_{m,\theta_m}\left(dY_N|X_N\right) \\
&= \sum_{m\in\mathcal{M}} \rho(m)\mathbf{E}_{\rho(d\theta_m|m)} P_{m,\theta_m}\left(dY_N|X_N\right).
\end{aligned}
$$

Therefore it is a mixture under $\rho(m)$ of the estimator $\mathbf{E}_{\rho(\theta_m|m)} P_{m,\theta_m}\left(dY_N|X_N\right)$ which is, for any given model m, a thermalized version (at inverse temperature β) of a Bayesian estimator for the continuous parameters with respect to Jeffrey's prior on every simplex. This Bayesian estimator for θ_m has been studied in particular by Krichevsky and Trofimov ([52]). One interesting feature is that it can easily be computed using the following formula:

$$
\mathbf{E}_{\rho(d\theta_m|m)} p_{m,\theta_m}\left(y_N|x_N\right) = \frac{\beta a_{s_m(x_N)}^{y_N} + \frac{1}{2}}{\beta \sum_{y\in\mathcal{Y}} a_{s_m(x_N)}^{y} + \frac{\alpha}{2}} \tag{3.5}
$$

Remark 13 *For a given model m the additional term $C_N(m)$ decreases to zero and becomes negligible compared to the other terms as soon as the projections $P(dY|s_m(X)=s)$ are in the interior of the simplex for all $s\in\mathcal{S}_m$. Every node $s\in\mathcal{S}_m$ for which this projection is on the vertex of the simplex adds a risk of order $\frac{\alpha}{2}$ which is not negligible anymore compared to $D_m(\alpha-1)/2+\log 1/\pi(m)$. This is due to the fact that Jeffrey's prior is asymptotically minimax in the interior of the simplex, but only maximin on the whole simplex (see [93]).*

Remark 14 *The upper bound on the inverse temperature β is of order $(\log\log N)^{\frac{1}{2}}/\log N$. However this bound might be conservative, and it is reasonable to think from the computations in [22] and the experiments in [87] that $\beta=1/2$ will work in many cases.*

3.4 Twice-universal coding and statistical estimation

Let us have a look at the differences between the double mixture estimator we introduce for statistical regression purpose and "twice-universal" estimators used for compression in coding theory. In the compression framework the variables $Y_1^i = Y_1 \dots Y_i$ play the role of the variables X_i and the goal is to design a family of conditional probabilities $\left\{\hat{P}^i(dY_i|X_i)\right\}_{i\in\mathbb{N}}$ such that their per-sample redundancies be small compared to the per-sample redundancy of the best model when the number of observation goes to infinity, i.e. that

$$
\frac{1}{N}\mathbf{E}_{P^N}\log\frac{1}{\prod_{i=1}^N \hat{P}^i(Y_i|X_i)} \leq \inf_{m\in\mathcal{M},\theta_m\in\Theta_m}\frac{1}{N}\mathbf{E}_{P^N}\log\frac{1}{\prod_{i=1}^N P_{m,\theta_m}(Y_i|X_i)} + \epsilon_N(m,\theta_m)
$$

with

$$\forall m \in \mathcal{M}, \forall \theta_m \in \Theta_m \qquad \lim_{N\to\infty} \epsilon_N(m,\theta_m) = 0.$$

More precisely such a family of estimators is called a "twice-universal" code ([71]) if it is strongly universal (in the sense of [30]) with respect to every model $m \in \mathcal{M}$, i.e. if $\sup_{\theta_m} \epsilon_N(m,\theta_m)$ goes to zero as N goes to infinity with the minimax rate of convergence for the model m. In other words a twice-universal code is minimax up to a vanishing term in the convergence rate with respect to every model m.

A good solution to this apparently difficult problem is to take for $\hat{P}$ a so-called "two-stage" mixture or double mixture ([41]), that is a discrete mixture of estimators for every model which are themselves mixtures of the probability distributions in the model class w.r.t. to a least favorable prior on the continuous parameter set. An impressive implementation of this idea has been carried out for a binary alphabet ($\alpha = 2$) in the so-called context tree weighting algorithm ([91]), where Jeffrey's prior $\mu(d\theta)$ is used on every simplex together with an arbitrary prior π on the model set to build the estimator:

$$p_w^i(y_i|x_i) = \frac{\displaystyle\sum_{m\in\mathcal{M}} \pi(m) \int_{\theta_m\in\Theta_m} \prod_{j=1}^{i} p_{m,\theta_m}(y_j|x_j)\mu(d\theta_m)}{\displaystyle\sum_{m\in\mathcal{M}} \pi(m) \int_{\theta_m\in\Theta_m} \prod_{j=1}^{i-1} p_{m,\theta_m}(y_j|x_j)\mu(d\theta_m)}, \tag{3.6}$$

which satisfies :

$$\frac{1}{N}\mathbf{E}_{P^N} \log \frac{1}{\prod_{i=1}^N P_w^i(Y_i|X_i)} \le \inf_{m\in\mathcal{M},\theta_m\in\Theta_m} \frac{1}{N}\mathbf{E}_{P^N} \log \frac{1}{\prod_{i=1}^N P_{m,\theta_m}(Y_i|X_i)} + \frac{1}{N}\left[\log\frac{1}{\pi(m)} + \sum_{s\in\mathcal{S}_m}\left(\frac{\log n_s}{2} + 1\right)\right].$$

This expression shows that this family of estimators has a per-sample redundancy which decreases at the minimax rate $D_m \log N/(2N)$.

In the case of statistical regression the criterion we are interested in is slightly different from the redundancy used for compression purpose. Indeed we are only interested in the estimation of the conditional law of Y_N knowing X_N and the observations Z_1^N, while the redundancy is only an average of this criterion for $i = 1, \ldots, N$. The relationship between the redundancy and the statistical risk is more precisely expressed in the following equality :

$$\frac{1}{N}\mathbf{E}_{P^N} \log \frac{1}{\prod_{i=1}^N Q(Y_i|X_i)} = \frac{1}{N}\sum_{i=1}^N \mathbf{E}_{P^i} \log\frac{1}{Q(Y_i|X_i)}.$$

In other words the estimates of the conditional law of Y_N knowing X_N obtained from a universal coding procedure have good performances in terms of *cumulative* risk for an increasing number of observations.

In order to compare the twice universal coding algorithm (3.6) and our double mixture statistical estimator defined in Theorem 8 we need to rewrite (3.6) as:

$$p_w^i(y_N|x_N) = \sum_{m\in\mathcal{M}} \rho_c(m) \frac{\displaystyle\int_{\theta_m\in\Theta_m} \prod_{j=1}^{N} p_{m,\theta_m}(y_j|x_j)\mu(d\theta_m)}{\displaystyle\int_{\theta_m\in\Theta_m} \prod_{j=1}^{N-1} p_{m,\theta_m}(y_j|x_j)\mu(d\theta_m)},$$

with

$$\forall m \in \mathcal{M} \qquad \rho_c(m) = \frac{\displaystyle\pi(m)\int_{\theta_m\in\Theta_m} \prod_{j=1}^{N-1} p_{m,\theta_m}(y_j|x_j)\mu(d\theta_m)}{\displaystyle\sum_{m'\in\mathcal{M}} \pi(m')\int_{\theta'_m\in\Theta'_m} \prod_{j=1}^{N-1} p_{m',\theta'_m}(y_j|x_j)\mu(d\theta_m)}. \tag{3.7}$$

If one forgets one second about the inverse temperature β (think of $\beta = 1$), it is possible to compare this posterior with the one expressed in Theorem 8 to point out one important difference:

$$\forall m \in \mathcal{M} \qquad \rho(m) \sim \rho_c(m) \times \prod_{s\in\mathcal{S}_m} \left[\left(\frac{n_s(1,1)}{2\pi e}\right)^{\frac{\alpha-1}{2}} . \frac{\pi^{\frac{\alpha}{2}}}{\Gamma\left(\frac{\alpha}{2}\right)} \right]. \tag{3.8}$$

This shows that in order to get a small statistical risk instead of a small cumulative risk one needs to modify the prior π on the model set in order to take into account the differences in the difficulty of estimating the continuous parameters. Besides a constant "penalization" term which does not depend on N, one sees in expression (3.8) that as N increases, models with a larger number of parameters should be given more and more weight because the term $\prod_{s\in\mathcal{S}_m} n_s^{\frac{\alpha-1}{2}}$ behaves like $N^{\frac{\alpha-1}{2}D_m}$.

An other way to look at the particularity of our double mixture estimator is to observe that the posterior weight $\rho(m)$ of a model m is essentially proportional to the maximum likelihood of the observed sequence in the model class. Indeed one can notice that if $\bar\theta$ is in the interior of the simplex,

$$\int_\Sigma \exp\left(-nd(\bar\theta||\theta)\right) \underset{n\to\infty}{\sim} \frac{\Gamma\left(\frac{\alpha}{2}\right)}{\pi^{\frac{\alpha}{2}}} \times \left(\frac{n}{2\pi}\right)^{-\frac{\alpha-1}{2}},$$

and therefore, for any model m in $\mathcal{M}$,

$$\begin{aligned}\rho(m) &\underset{N\to\infty}{\sim} \frac{1}{Z}\pi(m)\exp\left(-D_m(\alpha-1)/2\right) \times \prod_{s\in\mathcal{S}_m}\int_{\Sigma}\exp\left(-n_s(\beta,0)h(\bar{\theta}_s(\beta,0))\right)\mu(d\theta_s) \\ &\underset{N\to\infty}{\sim} \frac{1}{Z}\pi(m)\exp\left(-D_m(\alpha-1)/2\right) \times \sup_{\theta_m\in\Theta_m}\prod_{i=1}^{N-1} p_{m,\theta_m}(Y_i|X_i)^{\beta}.\end{aligned} \tag{3.9}$$

Compared with (3.7) one sees that instead of doing a double mixture one should:

- replace the mixture estimator for continuous parameters by the maximum likelihood in every model;
- penalize the likelihood by a factor $\exp\left(-D_m(\alpha-1)/2\right)$.

As far as performance is concerned the main difference between our double mixture estimator and a twice universal code is that the bound of the statistical estimator is not on the cumulated risk.

Remark 15 *An fundamental link exists between minimax estimators and mixture estimators : in can be proved under quite general assumptions that a mixture estimator with respect to a "least favorable" prior can be a minimax estimator (see a survey and references in [58]). Our formula for ρ gives an idea of what such a least favorable distribution could look like in the problem considered. Two points are of interest:*

- *The factor $\exp\left(-D_m(\alpha-1)/2\right)$ in the expression of $\rho(m)$ can be regarded as a penalty term for the dimension of the continuous parameter in each model.*
- *The prior on the simplex is Jeffrey's prior. This suggests a penalty term for the continuous parameters inversely proportional to the variance of the corresponding Bernoulli models.*

This can also be related to penalized maximum likelihood estimators ([6]) in which a penalization of models proportional to their dimension arises for other reasons.

3.5 Double mixture on context trees

3.5.1 The estimator

In this section we present a particular form of the double mixture estimator defined in Theorem 8 when the variable X is a string and the models considered are context trees. In other words we consider the case $\mathcal{X} = \mathcal{Y}^D$. We will basically use the same models as described in [87] where an application in natural language processing is proposed.

Let D be a fixed integer. We define a model m by a non-empty set $\mathcal{S}_m \subset \bigcup_{i=0}^{D} \mathcal{Y}^i$ of finite strings of length not larger than D such that *any suffix of any string of m be also in m* (by definition a suffix of a string $x_1 \ldots x_i$ is of the form $x_j \ldots x_i$ for some $j \leq i$). This definition implies in particular that the empty string λ belongs to every model.

The projection s_m associated with a model m is simply the transformation from any string $x \in \mathcal{X}$ into its longest suffix that is in $\mathcal{S}_m$.

Finally we can define a natural probability on $\mathcal{M}$ as follows:

$$\forall m \in \mathcal{M} \qquad \pi(m) = C^{D_m}, \tag{3.10}$$

where the constant C is adjusted so that $\sum_{m \in \mathcal{M}} \pi(m) = 1$.

It is shown in [87] that in that case,

$$\log \frac{1}{C} \leq 1 + \log \alpha.$$

Therefore the "model risk" is controlled as follows:

$$\forall m \in \mathcal{M} \qquad \log \frac{1}{\pi(m)} \leq (1 + \log \alpha)\, D_m.$$

As a result we can apply Theorem 8 to this particular setting to obtain:

Theorem 9 *Let G_β^N be the double mixture estimator as defined in Theorem 8. For any exchangeable distribution P_N on $(\mathcal{X} \times \mathcal{Y})^N$ it satisfies :*

$$\begin{aligned}
&\mathbf{E}_{P_N(dZ_1^{N-1})} \mathcal{K}\left(P_N\left(dY_N|X_N; Z_1^{N-1}\right), G_\beta^N\left(dY_N|X_N; Z_1^{N-1}\right)\right) \\
&\quad \leq \inf_{m \in \mathcal{M}, \theta_m \in \Theta_m} \Bigg\{ \mathbf{E}_{P_N(dZ_1^{N-1})} \mathcal{K}\left(P_N\left(dY_N|X_N; Z_1^{N-1}\right), P_{m,\theta_m}(dY_N|X_N)\right) \\
&\qquad\qquad + \frac{D_m}{\beta N}\left(\frac{\alpha - 1}{2} + 1 + \log \alpha + C_N(m)\right)\Bigg\},
\end{aligned}$$

with

$$C_N(m) = \frac{1}{D_m} \mathbf{E}_{P_N} \sum_{s \in \mathcal{S}_m} \left(\frac{\alpha^2}{4n_s(\beta, \beta)} + \frac{\alpha}{4n_s \min_i(\bar{\theta}^i(\beta, \beta)) + 2}\right).$$

Note that the larger the alphabet size the smaller the model risk term $1 + \log \alpha$ compared to the parameter risk $(\alpha - 1)/2$.

3.5.2 Implementation

The exact Gibbs estimator as a double mixture is difficult to compute efficiently because the number of models is very large as D and α increase. However it is possible to imagine a suboptimal implementation which computes an estimator which might be not so different from the double mixture estimators in many concrete cases.

We propose to replace the double mixture procedure by the selection of the model with the largest posterior distribution. Our hope is that in many cases the Gibbs posterior is unimodal and that the selection of a particular model with a large posterior probability is representative of the mixture in terms of probability law.

Following (3.9) and (3.10) we see that a good candidate for the quantity to maximize is:

$$\gamma(m) = \log \sup_{\theta_m \in \Theta_m} \prod_{i=1}^{N-1} p_{m,\theta_m}(Y_i|X_i) - \frac{D_m}{\beta}\left(\log\frac{1}{C} + \frac{\alpha-1}{2}\right).$$

This equation shows that the model selection we propose takes the form of a penalized maximum likelihood with a penalization proportional to the size of the model D_m. In order to get an efficient implementation of this model selection procedure we can use a context tree (see [91], [24], [87]), i.e. a suffix tree representing all possible strings of length smaller than D hierarchically, the root of the tree being the empty string λ. Let us attach the following counters to every node s of the context tree (we use the equivalence between a node and the associated string):

$$\begin{aligned} \forall i \in \mathcal{Y} \qquad a_s^i &= \sum_{n=1}^{N-1} \mathbf{1}\left(s \text{ is a suffix of } x_n \text{ and } y_n = i\right), \\ n_s &= \sum_{n=1}^{K} \mathbf{1}\left(s \text{ is a suffix of } x_n\right). \end{aligned}$$

If we now note $\delta = \left(\log 1/C + (\alpha-1)/2\right)/\beta$ and define the recursive function w on the context tree :

$$\begin{cases} \text{If } l(s) = D \quad w(s) = \displaystyle\sum_{y\in\mathcal{Y}} \frac{a_s^y}{n_s}\log\frac{a_s^y}{n_s} - \delta, \\ \text{If } l(s) < D \quad w(s) = \displaystyle\max_{\mathcal{N}\subset\mathcal{Y}} \left[\sum_{i\in\mathcal{N}} w(is) + \sum_{y\in\mathcal{Y}} \frac{a_s^y - \sum_{i\in\mathcal{N}} a_{is}^y}{n_s - \sum_{i\in\mathcal{N}} n_{is}} \log\frac{a_s^y - \sum_{i\in\mathcal{N}} a_{is}^y}{n_s - \sum_{i\in\mathcal{N}} n_{is}}\right] - \delta, \end{cases}$$

then it is easy to see that $\max_{m\in\mathcal{M}} \gamma(m) = w(\lambda)$ and that the model m which realizes this maximum is the connected component of λ in the set of nodes that are selected in $\mathcal{N}$ at every node in the definition of w.

For a given node s the problem remains to compute the corresponding subset $\mathcal{N}$ and to mark the selected node. This can be approximated using an iterative procedure to build $\mathcal{N}$, starting with $\mathcal{N} = \emptyset$ and adding nodes one by one until the function to maximize locally stops increasing.

The complexity of such an optimization procedure is linear in the number of nodes of the context tree, because at most α tests are performed at every node to test the children nodes to select. It is also not more than linear in N because only the visited nodes are concerned, and the size of memory required to store the context tree is also not more than linear in the number of observations and of course bounded by the size of the context tree. In [87] we show results of experiments using an implementation of an algorithm very similar to the one described in this paper (a "two-stage double mixture algorithm").

3.6 Proof of Theorem 8

3.6.1 The Gibbs estimator

Let us first recall some facts about the so-called Gibbs estimator introduced by Catoni in [22]. For a given class of conditional probability densities $\{p_\theta\}_{\theta\in\Theta}$ indexed by a parameter θ living in a measurable space Θ endowed with a prior probability measure $\pi(d\theta)$, the Gibbs estimator at inverse temperature $\beta \in \mathbb{R}^+$ has a density

$$g_\beta^N\left(y_N|x_N, z_1^{N-1}\right) = \mathbf{E}_{\rho_{\beta,0}(d\theta)} p_\theta\left(y_N|x_N\right)$$

where $\rho_{\beta,\xi}$ is the following Gibbs posterior:

$$\rho_{\beta,\xi}(d\theta) = \frac{\displaystyle\prod_{i=1}^{N-1} p_\theta\left(y_i|x_i\right)^\beta p_\theta\left(y_N|x_N\right)^\xi \pi(d\theta)}{\displaystyle\int_\Theta \prod_{i=1}^{N-1} p_\theta\left(y_i|x_i\right)^\beta p_\theta\left(y_N|x_N\right)^\xi \pi(d\theta)}.$$

This estimator can be considered as a "thermalized" version of both the Bayesian ($\beta = 1$) and the maximum likelihood ($\beta = +\infty$) estimators. Catoni studied in [22] this estimator in the high temperature region $\beta < 1$ which is equivalent to a deliberate underestimation of the sample size : to compute the Gibbs estimator, the empirical distribution of $N-1$ observations is plugged into the Bayes estimator for a sample of size $\beta(N-1)$. The reason to consider high temperatures is that the estimator gains stability with respect to the empirical process when β decreases (at the limit, it is constant when $\beta = 0$).

In order to control the risk of the Gibbs estimator let us introduce the following notations:

$$\chi = -\left(0 \wedge \inf_{\xi \in [0,1]} \frac{\mathbf{E}_{P_N} \mathbf{M}^3_{\rho_{\beta,\xi}^{Z_1^N}(d\theta)} \log p_\theta\left(y_N | x_N\right)}{\mathbf{E}_{P_N} \mathbf{Var}_{\rho_{\beta,\xi}^{Z_1^N}(d\theta)} \log p_\theta\left(y_N | x_N\right)} \right),$$

and

$$\gamma_\beta\left(\theta\right) = \mathbf{E}_{P_N} \mathbf{E}_{\rho_{\beta,\beta}^{Z_1^N}(\theta')} \left(\log \frac{\prod_{i=1}^{N} p_\theta(y_i|x_i)^\beta}{\prod_{i=1}^{N} p_{\theta'}(y_i|x_i)^\beta} \right).$$

The following result, which we will be used to estimate the performance of our double mixture estimator, is a particular form of the main theorem of [22]:

Theorem 10 (Catoni, [22]) *If the inverse temperature β is such that*

$$\beta < \frac{1}{\chi - 1} \left(\sqrt{1 + (\chi - 1)\left(2 - \frac{\log \chi}{\chi}\right) \frac{\log \chi}{\chi}} - 1 \right),$$

then the risk of the Gibbs estimator at inverse temperature β is upper bounded by

$$\begin{aligned} &\mathbf{E}_{P_N\left(dZ_1^{N-1}\right)} \mathcal{K}\left(P_N\left(dY_N | X_N; Z_1^{N-1}\right), G_\beta^N\left(dY_N | X_N; Z_1^{N-1}\right)\right) \\ &\qquad \leq \inf_{\theta \in \Theta} \left\{ \mathbf{E}_{P_N\left(dZ_1^{N-1}\right)} \mathcal{K}\left(P_N\left(dY_N | X_N; Z_1^{N-1}\right), P_\theta\left(dY_N | X_N\right)\right) + \frac{\gamma_\beta(\theta)}{\beta N} \right\}. \end{aligned}$$

In case Θ is discrete one can show that $\gamma_\beta(\theta)$ is upper bounded by $\log \pi\left(\{\theta\}\right)^{-1}$. However the interesting point of this estimator is that it can be computed for any set of parameters Θ, not necessarily discrete. In our case, we propose to apply this estimator for the set of parameters

$$\Theta = \left\{\left(m, \theta_m\right), m \in \mathcal{M}, \theta_m \in \Theta_m\right\},$$

endowed with a probability density expressed as a product:

$$\pi\left(d\theta\right) = \bar{\pi}(m) \times \prod_{s \in \mathcal{S}_m} \mu(d\theta_s), \tag{3.11}$$

where $\bar{\pi}$ is a prior probability on $\mathcal{M}$ (we will show that it has to be taken somehow different from π). Theorem 8 will be a direct consequence of Theorem 10 after estimating an upper bound on the inverse temperature β to be used and on the risk bound γ_β.

3.6.2 Choice of the inverse temperature β

In this section we prove the following lemma:

Lemma 4

$$\chi \leq \bar{\chi} = 24 + 8\log\left(N + \frac{\alpha}{2} + 1\right).$$

A direct consequence of this lemma is the possibility to chose the inverse temperature β as

$$\begin{aligned}\beta &< \frac{1}{\bar{\chi}-1}\left(\sqrt{1+(\bar{\chi}-1)\left(2-\frac{\log\bar{\chi}}{\bar{\chi}}\right)\frac{\log\bar{\chi}}{\bar{\chi}}}-1\right)\\ &\underset{N\to\infty}{\sim} \frac{\sqrt{2\log\log(N)}}{8\log(N)},\end{aligned}$$

in order to fulfill the conditions of Theorem 10.

Proof of Lemma 4:

For any given $z_1^N \in (\mathcal{X}\times\mathcal{Y})^N$ and $\beta\in[0,1]$, let η and f be defined for $\xi\in[0,1]$ by:

$$\begin{cases} \eta(\xi) &= \mathbf{E}_{\pi(d\theta)} \prod_{i=1}^{N-1} p_\theta\left(y_i|x_i\right)^\beta p_\theta\left(y_N|x_N\right)^\xi, \\ f(\xi) &= \log\eta(\xi)\end{cases}$$

The function f is related to the Gibbs estimator through the following equality:

$$\log g_\beta^N\left(y_N|x_N; z_1^N\right) = f(1) - f(0).$$

Moreover a simple computation shows that the first three derivatives of f are equal to the moments of $\log p_\theta(y_N|x_N)$ under $\rho_{\beta,\xi}^{z_1^N}(d\theta)$:

$$\begin{aligned} f'(\xi) &= \mathbf{E}_{\rho_{\beta,\xi}^{z_1^N}(d\theta)} \log p_\theta\left(y_N|x_N\right), \\ f''(\xi) &= \mathbf{Var}_{\rho_{\beta,\xi}^{z_1^N}(d\theta)} \log p_\theta\left(y_N|x_N\right), \\ f^{(3)}(\xi) &= \mathbf{M}^3_{\rho_{\beta,\xi}^{z_1^N}(d\theta)} \log p_\theta\left(y_N|x_N\right). \end{aligned}$$

Using (3.11) and (3.4) we see that for $\xi\in[0,1]$,

$$\begin{aligned}\eta\left(\xi\right) &= \sum_{m\in\mathcal{M}} \bar{\pi}(m) \int_{\Theta_m} \prod_{i=1}^{N-1} p_{m,\theta_m}\left(y_i|x_i\right)^{\beta} p_{m,\theta_m}\left(y_N|x_N\right)^{\xi} \mu(d\theta_m) \\ &= \sum_{m\in\mathcal{M}} \bar{\pi}(m) \prod_{s\in\mathcal{S}_m} \int_{\Sigma} \exp\left[-n_s(\beta,\xi)\left(h(\bar{\theta}_s(\beta,\xi)) + d(\bar{\theta}_s(\beta,\xi)||\theta_s)\right)\right] \mu\left(d\theta_s\right).\end{aligned}$$

However for every model $m \in \mathcal{M}$ the variables $n_s(\beta,\xi)$ and $\bar{\theta}_s(\beta,\xi)$ only depend on ξ for $s = s(x_N)$. Besides, using (3.3), the integral involved in the preceding formula is known to be (for $s = s(x_N)$):

$$\int_{\Sigma} \exp\left[-n_s(\beta,\xi)\left(h(\bar{\theta}_s(\beta,\xi)) + d(\bar{\theta}_s(\beta,\xi)||\theta_s)\right)\right] \mu\left(d\theta_s\right) = Cte \times \frac{\Gamma\left(\beta a_s^{y_N} + \frac{1}{2} + \xi\right)}{\Gamma\left(n_s + \frac{\alpha}{2} + \xi\right)},$$

where Cte is a term which does not depend on ξ. As a result, if we introduce the functions:

$$\forall (m,\xi) \in \mathcal{M} \times [0,1] \qquad \mu_m\left(\xi\right) = \frac{\Gamma\left(\beta a_{s_m(s_N)}^{y_N} + \frac{1}{2} + \xi\right)}{\Gamma\left(n_s + \frac{\alpha}{2} + \xi\right)},$$

then η can be decomposed in the following way:

$$\eta(\xi) = \sum_{m\in\mathcal{M}} \lambda_m \mu_m(\xi), \tag{3.12}$$

where the $(\lambda_m)_{m\in\mathcal{M}}$ do not depend on ξ.

In order to express the derivatives of μ and η let us introduce the Polygamma functions:

$$\forall i \in \mathbb{N} \qquad \psi_i(z) = \frac{d^{i+1}}{dz^{i+1}} \log\Gamma(z).$$

Indeed, if we use the notation:

$$\forall\left(i,m\right) \in \mathbb{N}\times\mathcal{M} \qquad \phi_i^m(\xi) = \psi_i\left(\beta a_{s_m(x_N)}^{y_N} + \frac{1}{2} + \xi\right) - \psi_i\left(n_{s_m(x_N)} + \frac{\alpha}{2} + \xi\right),$$

then we get:

$$\mu_m' = \mu_m \phi_0^m, \tag{3.13}$$

$$\mu_m'' = \mu_m\left(\phi_1^m + (\phi_0^m)^2\right), \tag{3.14}$$

$$\mu_m^{(3)} = \mu_m\left((\phi_0^m)^3 + 3\phi_0^m\phi_1^m + \phi_2^m\right). \tag{3.15}$$

In order to control γ_β we will need the following controls on ϕ_i^m :

Lemma 5 *For all* (z_1^N, ξ, m) *in* $\mathcal{Z}_1^N \times [0,1] \times \mathcal{M}$ *the following inequalities hold:*

$$0 \geq \phi_0^m(\xi) \geq -\left(\log\left(N + \frac{\alpha}{2} + 1\right) + 3\right),$$

and for any integer i, $i \geq 1$ *:*

$$0 \geq \frac{\phi_{i+1}^m(\xi)}{\phi_i^m(\xi)} \geq -2(i+2).$$

Proof of Lemma 5: In order to prove the first inequality concerning ϕ_0^m we use the fact (see [64]) that the Psi function ψ_0 is increasing on $\mathbf{R}_*^+$, that $\psi_0(1/2) = -\gamma - 2\log 2$ and that $\psi_0(z) \leq \log z + 1$ Therefore the following inequality holds $\forall (z_1^N, \xi, m) \in \mathcal{Z}_1^N \times [0,1] \times \mathcal{M}$:

$$\begin{aligned} 0 \geq \phi_0^m(\xi) &\geq \psi_0\left(\frac{1}{2}\right) - \psi_0\left(\beta N + \frac{\alpha}{2} + 1\right) \\ &\geq -\gamma - 2\log 2 - \log\left(\beta N + \frac{\alpha}{2} + 1\right) - 1 \\ &\geq -\left(\log\left(N + \frac{\alpha}{2} + 1\right) + 3\right) \end{aligned}$$

In order to prove the second inequality of Lemma 5 we can use the expression of ψ_i in terms of the Hurwitz Zeta function, for $i \geq 1$:

$$\psi_i(u) = (-1)^{i+1} i! \sum_{k=0}^{\infty} \frac{1}{(k+u)^{i+1}}.$$

This shows that for any $u > 1/2$ and $i \geq 1$:

$$0 \leq -\frac{\psi_{i+1}(z)}{\psi_i(z)} \leq 2(i+1).$$

Therefore, $\forall (z_1^N, \xi, m) \in \mathcal{Z}_1^N \times [0,1] \times \mathcal{M}$ and $i \geq 1$:

$$\begin{aligned} 0 \geq \frac{\phi_{i+1}^m(\xi)}{\phi_i^m(\xi)} &= \frac{\displaystyle\int_{\beta a_{sm(x_N)}^{y_N} + \frac{1}{2} + \xi}^{n_{sm(x_N)} + \frac{\alpha}{2} + \xi} \psi_{i+2}(u)du}{\displaystyle\int_{\beta a_{sm(x_N)}^{y_N} + \frac{1}{2} + \xi}^{n_{s(x_N)} + \frac{\alpha}{2} + \xi} \psi_{i+1}(u)du} \\ &\geq -2(i+2) \qquad \square \end{aligned}$$

We can now concentrate on the problem of upper bounding χ. Using the fact that $f = \log \eta$, one easily gets, for any given z_1^N:

$$\begin{aligned}
\frac{\mathbf{M}^3_{\rho_{\beta,\xi}^{z_1^N}(d\theta)} \log p_\theta\left(y_N|x_N\right)}{\mathbf{Var}_{\rho_{\beta,\xi}^{z_1^N}(d\theta)} \log p_\theta\left(y_N|x_N\right)} &= \frac{f^{(3)}}{f''}(\xi) \\
&= \frac{\eta^{(3)}\eta - \eta''\eta'}{\eta''\eta - (\eta')^2}(\xi) - \frac{2\eta'}{\eta}(\xi) \\
&\geq \frac{\eta^{(3)}\eta - \eta''\eta'}{\eta''\eta - (\eta')^2}(\xi),
\end{aligned} \tag{3.16}$$

the last inequality holding because $\eta'/\eta = f' = \mathbf{E}_\rho \log p_\theta(y_N|x_N) \leq 0$.

Let us now consider an ordered list of models : $\mathcal{M} = (m_1, \dots)$. In order to simplify the notations, let us write ϕ_i^j for $\phi_i^{m_j}$, for $i \in \mathbb{N}$, and let us note:

$$\forall (i,j) \in \mathbb{N}^2, \qquad q_{i,j} = \begin{cases} \lambda_{m_i}\lambda_{m_j}\mu_{m_i}(\xi)\mu_{m_j}(\xi) & \text{if } i \neq j; \\ \frac{1}{2}\lambda_{m_i}^2\mu_{m_i}(\xi)^2 & \text{if } i = j. \end{cases}$$

Using (3.16), (3.12) and (3.13) we finally get:

$$\begin{aligned}
\chi &\leq -\inf_{z_1^N \in \mathcal{Z}_1^N} \frac{\sum_{(i,j)\in\mathbb{N}^2} q_{i,j}\left[(\phi_0^i + \phi_0^j)(\phi_0^i - \phi_0^j)^2 + \phi_1^i(3\phi_0^i - \phi_0^j) + \phi_1^j(3\phi_0^j - \phi_0^i) + \phi_2^i + \phi_2^j\right]}{\sum_{(i,j)\in\mathbb{N}^2} q_{i,j}\left[(\phi_0^i - \phi_0^j)^2 + \phi_1^i + \phi_1^j\right]} \\
&\leq -\inf_{z_1^N \in \mathcal{Z}_1^N} \inf_{(i,j)\in\mathbb{N}^2} \frac{(\phi_0^i + \phi_0^j)(\phi_0^i - \phi_0^j)^2 + \phi_1^i(3\phi_0^i - \phi_0^j) + \phi_1^j(3\phi_0^j - \phi_0^i) + \phi_2^i + \phi_2^j}{(\phi_0^i - \phi_0^j)^2 + \phi_1^i + \phi_1^j} \\
&\leq -\inf_{z_1^N \in \mathcal{Z}_1^N} \inf_{(i,j)\in\mathbb{N}^2} \left(\frac{(\phi_0^i + \phi_0^j)(\phi_0^i - \phi_0^j)^2}{(\phi_0^i - \phi_0^j)^2} + \frac{\phi_1^i(3\phi_0^i - \phi_0^j)}{\phi_1^i} + \frac{\phi_1^j(3\phi_0^j - \phi_0^i)}{\phi_1^j} + \frac{\phi_2^i}{\phi_1^i} + \frac{\phi_2^j}{\phi_1^j}\right) \\
&\leq -\inf_{z_1^N \in \mathcal{Z}_1^N} \inf_{(i,j)\in\mathbb{N}^2} \left(4\phi_0^i + 4\phi_0^j + \frac{\phi_2^i}{\phi_1^i} + \frac{\phi_2^j}{\phi_1^j}\right) \\
&\leq 24 + 8\log\left(N + \frac{\alpha}{2} + 1\right),
\end{aligned}$$

which proves Lemma 4. □.

3.6.3 Upper bound for the risk

Let us first state a lemma in order to be able to control the Psi function. Remember that the Psi function ψ_0 (also called Digamma function) is defined by

$$\psi_0(x) = \frac{\partial}{\partial x} \log \Gamma(x).$$

Lemma 6

$$\begin{aligned} &\forall x > 0 \qquad && \psi_0\left(x + \frac{1}{2}\right) \geq \log x, \\ &\forall \alpha \in \mathbb{N}, \alpha \geq 2, \forall x > 0 \qquad && \psi_0\left(x + \frac{\alpha}{2}\right) \leq \log x + \frac{\alpha - 1}{2x}. \end{aligned}$$

Proof of Lemma 6:

To prove the first inequality, we write ψ_0 as an integral:

$$\forall x > 0 \qquad \psi_0(x) = \int_0^\infty \left(\frac{e^{-t}}{t} - \frac{e^{-tx}}{1 - e^{-t}}\right) dt,$$

and do the same for $\log x$:

$$\forall x > 0 \qquad \log x = \int_0^\infty \frac{e^{-t} - e^{-tx}}{t} dt.$$

Therefore, for all $x > 0$,

$$\begin{aligned} \log x - \psi_0\left(x + \frac{1}{2}\right) &= \int_0^\infty \frac{e^{-xt - t/2}}{1 - e^{-t}} - \frac{e^{-tx}}{t} dt \\ &= \int_0^\infty e^{-tx} \phi(t) dt, \end{aligned}$$

with, for all $t > 0$,

$$\begin{aligned} \phi(t) &= \frac{e^{-t/2}}{1 - e^{-t}} - \frac{1}{t} \\ &= \frac{1}{2 \sinh\left(\frac{t}{2}\right)} - \frac{1}{t}. \end{aligned}$$

Now it suffices to notice that $\sinh(t) \geq t$, which implies that $2 \sinh(t/2) \geq t \geq 0$, and therefore $\phi(t) \leq 0$ for all $t > 0$. This proves the first inequality of the lemma.

For the second inequality we can use the following, proved for instance in [2]:

$$\forall x > 0 \qquad \psi_0(x) < \log x - \frac{1}{2x}.$$

Therefore we can write, for all $x > 0$,

$$\psi_0\left(x + \frac{\alpha}{2}\right) - \log x - \frac{\alpha - 1}{2x} \leq \phi_\alpha(x),$$

with

$$\phi_\alpha(x) = \log\left(1+\frac{\alpha}{2x}\right) - \frac{\alpha-1}{2x} - \frac{1}{2x+\alpha}.$$

Let us introduce $y = \alpha/(2x)$. Then we can write :

$$\phi_\alpha(x) = \phi_\alpha^y(y) = \log(1+y) - \frac{\alpha-1}{\alpha}y - \frac{y}{\alpha(1+y)}.$$

whose derivative is

$$(\phi_\alpha^y)'(y) = \frac{y}{\alpha(1+y)^2}\left[y(1-\alpha)+2-\alpha\right].$$

But α is supposed to be an integer larger or equal to 2, so $(\phi_\alpha^y)'(y) \leq 0$ for $y > 0$. In other words ϕ_α^y as a function of y is decreasing on $\mathbb{R}^+$, and $\phi_\alpha^y(0) = 0$. As a result, $\phi_\alpha^y(y) \leq 0$ for all $y > 0$. This is sufficient to prove the second inequality of Lemma 6. □

Let us now evaluate the risk defined by

$$\forall m \in \mathcal{M}, \theta_m \in \Theta_m \qquad \gamma_\beta(m,\theta_m) = \mathbf{E}_{P_N}\mathbf{E}_{\rho^{Z_1^N}_{\beta,\beta}(m',d\theta')}\left(\log\frac{\prod_{i=1}^N p_{m,\theta_m}(y_i|x_i)^\beta}{\prod_{i=1}^N p_{m',\theta_{m'}}(y_i|x_i)^\beta}\right).$$

Using equation (3.4) it is possible to express the posterior Gibbs distribution as:

$$\rho_{\beta,\xi}\left(d\theta_m|m\right) = \frac{\prod_{s\in\mathcal{S}_m}\exp\left[-n_s(\beta,\xi)d\left(\bar\theta_s(\beta,\xi)||\theta_s\right)\right]\mu\left(\theta_s\right)d\theta_s}{\prod_{s\in\mathcal{S}}\int_\Sigma\exp\left[-n_s(\beta,\xi)d\left(\bar\theta_s(\beta,\xi)||\theta_s\right)\right]\mu\left(\theta_s\right)d\theta_s}, \tag{3.17}$$

and

$$\rho_{\beta,\xi}\left(\mathcal{S}\right) \sim \bar\pi\left(\mathcal{S}\right)\prod_{s\in\mathcal{S}}\int_\Sigma\exp\left[-n_s(\beta,\xi)d\left(\bar\theta_s(\beta,\xi)||\theta_s\right)\right]\mu\left(\theta_s\right)d\theta_s$$
$$\times\exp\left[-\sum_{s\in\mathcal{S}}n_s(\beta,\xi)h\left(\bar\theta_s(\beta,\xi)\right)\right]. \tag{3.18}$$

In order to simplify the notations let us write $n_s = n_s(\beta,\beta)$ and $\bar\theta_s = \bar\theta_s(\beta,\beta)$ in the rest of this section. As a first step let us prove the following lemma:

Lemma 7 *For all m in $\mathcal{M}$,*

$$\mathbf{E}_{\rho^{Z_1^N}_{\beta,\beta}(d\theta_m|m)}\log\prod_{i=1}^N p_{m,\theta_m}(y_i|x_i)^{-\beta} \leq \sum_{s\in\mathcal{S}_m} n_s h\left(\bar\theta_s\right) + \frac{\alpha-1}{2}D_m.$$

Proof of Lemma 7: Using (3.17) we get:

$$\mathbf{E}_{\rho_{\beta,\beta}^{Z_1^N}(d\theta_m|m)} \log \prod_{i=1}^{N} p_{m,\theta_m}(y_i|x_i)^{-\beta}$$
$$= \sum_{s\in\mathcal{S}_m} n_s h\left(\bar{\theta}_s\right) + \sum_{s\in\mathcal{S}_m} \frac{\int_\Sigma n_s d\left(\bar{\theta}_s||\theta_s\right) e^{-n_s d\left(\bar{\theta}_s||\theta_s\right)} \mu(d\theta_s)}{\int_\Sigma e^{-n_s d\left(\bar{\theta}_s||\theta_s\right)} \mu(d\theta_s)}.$$

Consider now the function defined for $x \in \mathbb{R}^+$ by

$$f(x) = \prod_{s\in\mathcal{S}} \int_\Sigma e^{-x n_s d\left(\bar{\theta}_s||\theta_s\right)} \mu(d\theta_s).$$

All integrals being absolutely convergent the derivation under the integral is possible around $x = 1$, and one gets:

$$\mathbf{E}_{\rho_{\beta,\beta}^{Z_1^N}(d\theta_m|m)} \log \prod_{i=1}^{N} p_{m,\theta_m}(y_i|x_i)^{-\beta} = \sum_{s\in\mathcal{S}_m} n_s h\left(\bar{\theta}_s\right) - \frac{f'(1)}{f(1)}. \tag{3.19}$$

But $f'/f = (\log f)'$, so let us compute $\log f(x)$ for $x > 0$:

$$\begin{aligned} \log f(x) &= \sum_{s\in\mathcal{S}} \log \int_\Sigma e^{-x n_s d\left(\bar{\theta}_s||\theta_s\right)} \mu(d\theta_s) \\ &= \sum_{s\in\mathcal{S}} \left\{ x n_s h\left(\bar{\theta}_s\right) + \log \int_\Sigma e^{-x n_s \left[h\left(\bar{\theta}_s\right) + d\left(\bar{\theta}_s||\theta_s\right)\right]} \mu(d\theta_s) \right\}. \end{aligned}$$

The exact value of the integral is known in terms of the Gamma function (thanks to (3.3)):

$$\log \int_\Sigma e^{-x n_s \left[h\left(\bar{\theta}_s\right) + d\left(\bar{\theta}_s||\theta_s\right)\right]} \mu(d\theta_s) = C + \sum_{i=1}^{\alpha} \log \Gamma\left(x a_s^i + \frac{1}{2}\right) - \log \Gamma\left(x n_s + \frac{\alpha}{2}\right),$$

where C does not depend on x. Taking the derivatives in $x = 1$ of these expressions and using Lemma 6 we finally get:

$$\begin{aligned} -\frac{f'(1)}{f(1)} &= -\sum_{s\in\mathcal{S}} \left[\sum_{i=1}^{\alpha} a_s^i \left(\psi_0\left(a_s^i + \frac{1}{2}\right) - \log a_s^i \right) - n_s \left(\psi_0\left(n_s + \frac{\alpha}{2}\right) - \log n_s \right) \right] \\ &\leq \sum_{s\in\mathcal{S}} \frac{\alpha - 1}{2} \\ &\leq \frac{\alpha - 1}{2} D_m, \end{aligned}$$

and coming back to (3.19) we obtain:

$$\mathbf{E}_{\rho^{Z_1^N}_{\beta,\beta}(d\theta_m|m)} \log \prod_{i=1}^{N} p_{m,\theta_m}(y_i|x_i)^{-\beta} \leq \sum_{s\in\mathcal{S}_m} n_s h\left(\bar{\theta}_s\right) + \frac{\alpha-1}{2} D_m. \qquad \square$$

Let us use the notation:

$$\forall m \in \mathcal{M}, \forall z_1^N \in (\mathcal{X}\times\mathcal{Y})^N \qquad \lambda(m, z_1^N) = \prod_{s\in\mathcal{S}_m} \int_{\Sigma} e^{-n_s d\left(\bar{\theta}_s||\theta_s\right)} \mu\left(\theta_s\right) d\theta_s. \tag{3.20}$$

Using (3.18) and (3.19) we get the following equality, for any m in $\mathcal{M}$:

$$\begin{aligned}
&\mathbf{E}_{\rho_{\beta,\beta}(m,d\theta_m)} \log \prod_{i=1}^{N} p_{m,\theta_m}(y_i|x_i)^{-\beta} \\
&\leq \frac{\sum_{m\in\mathcal{M}} \bar{\pi}(m)\, \lambda\left(m, z_1^N\right) e^{-\sum_{s\in m} n_s h\left(\bar{\theta}_s\right)} \left(\sum_{s\in m} n_s h\left(\bar{\theta}_s\right) + \frac{\alpha-1}{2} D_m\right)}{\sum_{m\in\mathcal{M}} \bar{\pi}(m)\, \lambda\left(m, z_1^N\right) e^{-\sum_{s\in m} n_s h\left(\bar{\theta}_s\right)}} \\
&\leq \frac{\sum_{m\in\mathcal{M}} \tilde{\pi}(m)\, \tilde{g}(m)\, e^{-\tilde{g}(m)}}{\sum_{m\in\mathcal{M}} \tilde{\pi}(m)\, e^{-\tilde{g}(m)}},
\end{aligned}$$

with, for all m in $\mathcal{M}$:

$$\begin{aligned}
\tilde{\pi}(m) &= \bar{\pi}(m)\, e^{\frac{\alpha-1}{2} D_m} \lambda\left(m, z_1^N\right), \\
\tilde{g}(m) &= \sum_{m\in\mathcal{M}} n_s(\beta,\beta) h\left(\bar{\theta}_s(\beta,\beta)\right) + \frac{\alpha-1}{2} D_m \\
&= \sup_{\theta_m\in\Sigma^{D_m}} \log \prod_{i=1}^{N} p_{m,\theta_m}(y_i|x_i)^{-\beta} + \frac{\alpha-1}{2} D_m.
\end{aligned}$$

Introducing a threshold ϵ to be optimized afterwards and using the fact that xe^{-x} is upper bounded by e^{-1} on $\mathbf{R}^+$, this expression can be upper bounded for any particular $\bar{m} \in \mathcal{M}$ by:

$$\begin{aligned}
&\frac{\sum_{m\in\mathcal{M}} \tilde{\pi}(m)\, \tilde{g}(m)\, e^{-\tilde{g}(m)}}{\sum_{m\in\mathcal{M}} \tilde{\pi}(m)\, e^{-\tilde{g}(m)}} \\
&\qquad \leq \epsilon + \frac{\sum_{m\in\mathcal{M}} \tilde{\pi}(m) \left[\left(\tilde{g}(m) - \epsilon\right)_+\right] e^{-\tilde{g}(m)}}{\sum_{m\in\mathcal{M}} \tilde{\pi}(m)\, e^{-\tilde{g}(m)}}
\end{aligned}$$

$$\leq \epsilon + \exp(-\epsilon) \frac{\sum_{m \in \mathcal{M}} \tilde{\pi}(m) \left[(\tilde{g}(m) - \epsilon)_+ \right] e^{-(\tilde{g}(m) - \epsilon)}}{\sum_{m \in \mathcal{M}} \tilde{\pi}(m) e^{-\tilde{g}(m)}}$$

$$\leq \epsilon + \exp(-\epsilon) \frac{e^{-1} \sum_{m \in \mathcal{M}} \tilde{\pi}(m)}{\tilde{\pi}(\bar{m}) e^{-\tilde{g}(\bar{m})}}.$$

Taking $\epsilon = -1 + \log \sum_{m \in \mathcal{M}} \tilde{\pi}(m) - \log \tilde{\pi}(\bar{m}) + \tilde{g}(\bar{m})$, we finally get:

$$\mathbf{E}_{\rho_{\beta,\beta}(m, d\theta_m)} \log \prod_{i=1}^{N} p_{m,\theta_m}(y_i|x_i)^{-\beta} \leq \tilde{g}(\bar{m}) + \log \frac{1}{\tilde{\pi}(\bar{m})} + \log \sum_{m \in \mathcal{M}} \tilde{\pi}(m).$$

This proves the following lemma:

Lemma 8 *For any m' in $\mathcal{M}$,*

$$\sup_{\theta_{m'} \in \Theta_{m'}} \gamma_\beta(m', \theta_{m'}) \leq \frac{\alpha - 1}{2} D_m + \mathbf{E}_{P^N} \left(\log \frac{1}{\tilde{\pi}(m')} + \log \sum_{m \in \mathcal{M}} \tilde{\pi}(m) \right),$$

with

$$\tilde{\pi}(m) = \bar{\pi}(m) \exp\left(\frac{\alpha - 1}{2} D_m \right) \prod_{s \in \mathcal{S}_m} \int_\Sigma \exp\left[-n_s(\beta,\beta) d\left(\bar{\theta}_s(\beta,\beta) || \theta_s \right) \right] \mu(\theta_s)\, d\theta_s.$$

Lemma 8 shows that the bound on $\gamma_\beta(m, \theta_m)$ is the sum of a parameter risk $(\alpha - 1) D_m / 2$ and a model risk $-\log \tilde{\pi}(m) / \left(\sum_{\mathcal{M}} \tilde{\pi}(m) \right)$, but with a functional $\tilde{\pi}$ different from the prior distribution $\bar{\pi}$. Besides, the ratio between $\tilde{\pi}(m)$ and $\bar{\pi}(m)$ is the product of two terms:

- a term that only depends on the size of m : $\exp((\alpha - 1) D_m / 2)$;
- a term that depends on the *unobserved* $\bar{\theta}_m(\beta,\beta)$:

$$\prod_{s \in \mathcal{S}_m} \int_\Sigma \exp\left[-n_s(\beta,\beta) d\left(\bar{\theta}_s(\beta,\beta) || \theta_s \right) \right] \mu(\theta_s)\, d\theta_s$$

The reason why we decided to take μ equal to Jeffrey's prior is that it makes the second term asymptotically independent of $\bar{\theta}$ (at least inside of the simplex), thanks to the following lemma:

Lemma 9 *For any $(n, \bar{\theta}) \in \mathbb{R}_*^+ \times \Sigma$ let f be the function,*

$$f(n, \bar{\theta}) = \log \frac{\Gamma\left(\frac{\alpha}{2}\right)}{\Gamma\left(\frac{1}{2}\right)^\alpha} + \frac{\alpha - 1}{2} \log\left(\frac{2\pi}{n} \right) - \log\left(\int_\Sigma e^{-n d(\bar{\theta} || \theta)} \mu(d\theta) \right).$$

This function satisfies:

$$\forall (n,\bar{\theta}) \in \mathbb{R}_*^+ \times \Sigma \qquad 0 \leq f(n,\bar{\theta}) \leq \frac{\alpha^2}{4n} + \frac{\alpha}{4n \min_i(\bar{\theta}^i) + 2}.$$

Proof of Lemma 9: The upper bound is proven in [93, Lemma 1] for $n \in \mathbb{N}$ and $\bar{\theta}$ being of the form $(a_1/n, \ldots, a_\alpha/n)$ with $(a_1, \ldots, a_\alpha) \in \mathbb{N}^\alpha$. The proof, based on Stirling's formula to approximate the Γ function, still works in the general case $(n,\bar{\theta}) \in \mathbb{R}_*^+ \times \Sigma$.

However the proof used for the lower bound in that case ([93, Lemma 2]) does not work in the general case. Therefore let us just prove the lower bound. Using (3.3) f can be rewritten as:

$$f(n,\bar{\theta}) = -nh\left(\bar{\theta}\right) + \frac{\alpha-1}{2}\log\frac{2\pi}{n} - \sum_{i=1}^{\alpha} \log\Gamma\left(n\bar{\theta}^i + \frac{1}{2}\right) + \log\Gamma\left(n + \frac{\alpha}{2}\right),$$

whose derivative w.r.t. n is equal to :

$$\begin{aligned}
\frac{\partial f}{\partial n}(n,\bar{\theta}) &= -h\left(\bar{\theta}\right) - \frac{\alpha-1}{2n} - \sum_{i=1}^{\alpha} \bar{\theta}^i \psi_0\left(n\bar{\theta}^i + \frac{1}{2}\right) + \psi_0\left(n + \frac{\alpha}{2}\right) \\
&= -\frac{\alpha-1}{2n} - \sum_{i=1}^{\alpha} \bar{\theta}^i \left[\psi_0\left(n\bar{\theta}^i + \frac{1}{2}\right) - \log\left(n\bar{\theta}^i\right)\right] + \psi_0\left(n + \frac{\alpha}{2}\right) - \log n \\
&\leq -\frac{\alpha-1}{2n} + 0 + \frac{\alpha-1}{2n} \\
&\leq 0
\end{aligned}$$

where we used Lemma 6 in order to obtain the inequality. As a result, for any given $\bar{\theta} \in \Sigma$, the function $n \mapsto f(n,\bar{\theta})$ is decreasing on $\mathbb{R}_*^+$. Besides, Laplace method of integration shows that for any $\bar{\theta}$ in the interior of the simplex,

$$\lim_{n\to\infty} f(n,\bar{\theta}) = 0.$$

As a result, $f(n,\bar{\theta}) \geq 0$ for any $n > 0$ and $\bar{\theta}$ in the interior of the simplex. Now if $\bar{\theta}$ is on the boundary of the simplex, one can consider a sequence $(\theta_k)_{k\geq 0}$ of points in the interior of the simplex which converges to $\bar{\theta}$. By the theorem of dominated convergence for a fixed $n > 0$ the integrals $\int_\Sigma \exp\left(-nd(\theta_k||\theta)\right)\mu(d\theta)$ converge to $\int_\Sigma \exp\left(-nd(\bar{\theta}||\theta)\right)\mu(d\theta)$ as $k \to \infty$. As a result the lower bound that we proved in the interior of the simplex remains true on its border, for any $n > 0$. This proves Lemma 9. □

If π is a prior on $\mathcal{M}$, consider now the Gibbs estimator formed with the prior $\bar{\pi}$ such that:

$$\forall m \in \mathcal{M} \qquad \bar{\pi}(m) = \frac{1}{Z} \frac{\pi(m) \exp\left(-\frac{\alpha-1}{2} D_m\right)}{\prod_{s \in \mathcal{S}_m} \left(\frac{2\pi}{n_s}\right)^{\frac{\alpha-1}{2}} C_\alpha^{-1}}$$
$$= \frac{1}{Z} \pi(m) \prod_{s \in \mathcal{S}_m} C_\alpha \times \left(\frac{n_s}{2\pi e}\right)^{\frac{\alpha-1}{2}},$$

where Z is a normalizing constant and $C_\alpha = \Gamma(1/2)^\alpha / \Gamma(\alpha/2) = \pi^{\alpha/2}/\Gamma(\alpha/2)$.

Note that X_N is observed so n_s is an observed variable which is invariant under permutation of z_1^N, and therefore $\bar{\pi}$ can be taken as a prior to form the Gibbs estimator. For such a choice, Lemma 8 is valid with the following $\tilde{\pi}$:

$$\tilde{\pi}(m) = \bar{\pi}(m) \exp\left(\frac{\alpha-1}{2} D_m\right) \lambda\left(m, z_1^N\right)$$
$$= \frac{1}{Z} \pi(m) \prod_{s \in \mathcal{S}_m} \frac{\int_\Sigma \exp\left(n_s d\left(\bar{\theta}_s || \theta_s\right)\right) \mu(\theta_s) d\theta_s}{C_\alpha^{-1} \left(\frac{2\pi}{n_s}\right)^{\frac{\alpha-1}{2}}}.$$

Using Lemma 9 we obtain the following bound:

$$\log \frac{\sum_{m' \in \mathcal{M}} \tilde{\pi}(m')}{\tilde{\pi}(m)} \leq \log \frac{1}{\pi(m)} + \sum_{s \in \mathcal{S}_m} \left(\frac{\alpha^2}{4n_s} + \frac{\alpha}{4n_s \min_i(\bar{\theta}_i) + 2}\right)$$

.

Finally, using Lemmas 8 we get:

$$\forall (m, \theta_m) \in \Theta \qquad \gamma_\beta(m, \theta_m) \leq \frac{\alpha-1}{2} D_m + \log \frac{1}{\pi(m)} + \sum_{s \in \mathcal{S}_m} \left(\frac{\alpha^2}{4n_s} + \frac{\alpha}{4n_s \min_i(\bar{\theta}_i) + 2}\right).$$

Applying Theorem 10 finishes the proof of Theorem 8. □

3.7 Conclusion

Our goal in this paper was to adapt the idea of twice-universal codes studied in universal compression to the problem of statistical density estimation. The similarity between the redundancy criterion in compression and the cumulated statistical risk justifies this goal, but some technical works has to be done in order to get a bound on the statistical risk of the estimator, and not on the cumulated statistical risk for samples of increasing sizes. We could get a result for a mixture estimator by using a Gibbs estimator as studied by Catoni in [22] and translating double-mixture codes ([41], [91]) into double-mixture statistical estimators.

The implementation procedure we suggest in section 3.5.2 takes the form of a penalized maximum likelihood model selection, justified by the selection of the model with highest posterior Gibbs distribution. However the Gibbs estimator is of a mixture of models and one could also imagine a approximation of this mixture using Monte Carlo simulations, instead of selecting one particular model (see [24], [16]).

As far as applications of such estimators are concerned, we refer to [87] for an example in natural language processing. It is shown how to use adaptive models in order to represent non-stochastic objects, e.g. texts, from which a statistical experiment is carried out. Such a representation can then be used to characterize the original object; as an application the similarity between two objects can be estimated by computing the similarity between the two corresponding models.

Chapter 4

Text categorization experiments

Abstract

A new way of representing texts written in natural language is introduced, as a conditional probability distribution at the letter level learned with a variable length Markov model called *adaptive context tree* model. Text categorizat ion experiments demonstrates the ability of this representation to catch informa tion about the semantic content of the text.

4.1 Introduction

Managing the information contained in increasingly large textual databases, including corporate databases, digital libraries or the World Wide Web, is now a challenge with huge economic stakes. The starting point of any information organization and management system is a way to transform texts, i.e. long strings of ASCII symbols, into objects adapted to further processing or operations for any particular task. Consider for example the problem of *text categorization*, that is the automatic assignment of natural language texts to predefined classes or categories. This problem received much attention recently and many algorithms have been proposed and evaluated, including but not limited to Bayesian classifiers ([55], [57], [63]), k-nearest neighbors ([94]), rule learning algorithms ([80], [59]), maximum entropy models ([62]), boosting ([75]) or support vector machines ([50], [38], [51]). All these algorithms share in common the way the initial text is processed from a long ASCII string into a series of words or word stems, and most of them carry out the classification from variants of the so-called *vector space* model ([74]) which consists in representing the initial text as a vector of frequencies of words in a given dictionary.

In spite of the impressive results obtained by some of the above algorithms on particular databases and categorization tasks it seems that these performances degrade as the database becomes more general and the task less specific. As a result such apparently easy tasks as filtering and classifying electronic e-mails into personal mailboxes remain non-trivial because of the poorly-formatted nature of such texts and the variations in the language used and the topics.

One of the reasons underlying these difficulties is the huge size of the set of possible words compared to the size of each text and the number of texts available for training the classifiers. This leads to large variations between texts inside of a category in terms of vector space representations, and to difficult statistical estimations during the training period. Not surprisingly support vector machines outperform most "classical" classification methods ([50]) because of their ability to deal with such issues.

This paper is an attempt to forget for a while the vector space model and consider alternative ways of extracting informations from natural language texts. Instead of parsing a text into tokens (words, word stems...) we just consider it as series of letters and estimate a letter-generating source model, i.e. a conditional probability of emitting a letter knowing the past, that "fits" the text correctly. The model estimation is done by an algorithm called *adaptive context trees* studied in [87] and produces a new representation of a text as a *context tree model*, which can be seen as a variable length Markov model. In order to study the pertinence of this representation a text classification algorithm is developed and tested. Encouraging results suggest that this representation might be able to "catch" features correlated with the semantic content of the text but not based on the words.

This paper is organized as follows. In Sect. 4.2 we highlight the general trade-off between the richness of a representation and the difficulty to estimate it, which motivates our representation introduced in Sect. 4.3. A classification algorithm is derived in Sect. 4.4 and experimental results appear in the following sections.

4.2 A Trade-Off in Representation

A digital text written in natural language is basically a series of bytes which has to be processed and transformed into a representation adapted to further operations such as text classification. A commonly used procedure consists in first rewriting it as a string of elements of a finite alphabet $\mathcal{A}$, e.g. a dictionary of words, word stems, tokens or letters, and then representing the text as a vector whose coordinates are the numbers of occurrence of each element of $\mathcal{A}$ in the pre-processed string. Depending on the alphabet $\mathcal{A}$ different situations might arise:

- If $\mathcal{A}$ is very large (think of a dictionary of all possible words for English texts, which typically contains several tenths of thousands of words) the semantic information contained in the vector space representation is known to be very rich, but the vectors corresponding to two related texts might be completely different because of the small size of every single text compared to the size of the dictionary. In other words the representation is unstable because it is statistically difficult to estimate any hidden distribution in a large space from few observations.

- On the other hand if $\mathcal{A}$ is very small (think of the 26-letters Latin alphabet plus some punctuation signs) the vector space representation has the advantage of being more stable even for small texts but the dramatic drawback of containing few semantic informations. As an example the frequencies of various letters might be a good indicator to guess the language of a text (e.g. English versus French) because they are usually characteristic of the language even for small texts, but they might not be appropriate features to guess whether an English text is about politics of religion.

These remarks show that there exists a *trade-off* between the information contained in a representation and the difficulty to estimate it from a finite and possibly short text. As far as the vector space model is concerned various techniques exist in order to decrease the size of the alphabet while keeping the semantic contents of words ([3]): these techniques include word stemming, thesaurus, stop words removal, feature selection etc...

Forgetting for a while the vector space representation it is possible to observe the same balance phenomenon in an other setting : the representation of a text $\mathcal{T}$ by a letter-generating source, i.e. by a conditional probability $\mathbb{P}_{\mathcal{T}}(Y \mid X)$ where Y is a random variable on the alphabet $\mathcal{A}$ which represents the next letter to be generated and X is a random variable on $\mathcal{A}^* = \cup_{n \geq 0} \mathcal{A}^i$ (the set of finite-length strings) which represents the past sequence of letters. The idea is that such a source is characteristic of a certain category of texts, and the goal of the representation is to estimate the source from a text supposed to be generated by it.

Note that even if the alphabet is poor - think of ASCII symbols or the Latin alphabet - this ideal representation $\mathbb{P}_{\mathcal{T}}(Y|X)$ is very rich because it suffices to define a stationary process which might be assimilated to the process of writing a text in the category specific of the source. In particular it contains the stationary probability of any finite-length string, e.g. any word made of letters or even n-grams of words.

Estimating such a conditional probability from a finite-length text can be done with the help of finite-dimensional models, e.g. finite order Markov models. Such an approach leads to the same kind of balance as mentioned above in the vector space model : if the chosen model is complex (e.g. large order Markov model) then it potentialy can better mimic the unknown probability $\mathbb{P}_{\mathcal{T}}(Y \mid X)$ than simpler model, but it is much more difficult to estimate from a finite number of observations. In other words a trade-off has to be reached between the complexity of the model used to estimate $\mathbb{P}_{\mathcal{T}}(Y \mid X)$ and the risk when estimating it.

This representation as a letter-generating source is however better adapted to the trade-off quest than the vector space model because it is easier to compare models of various complexities (e.g. finite order Markov models) and chose a complexity than depends on the information available. In the next section we present an algorithm that fulfills this requirement and leads to an *adaptive* representation of any text as a more or less complex conditional probability.

4.3 Probability Estimation through Adaptive Context Trees

We consider a text as a deterministic object from which statistical information can be learned through sampling procedures. In order to get an independent and identically distributed (i.i.d) sample $(X_i, Y_i)_{i=1,\dots,N}$ we propose to follow N times the following procedure : randomly chose a position in the text with a uniform prior, let Y be the letter occuring at the selected position and let X be the string made of the letters preceding Y backward to the beginning of the text.

In order to estimate $\mathbb{P}_{\mathcal{T}}(Y|X)$ from the resulting i.i.d. sample $(X_i, Y_i)_{i=1,\dots,N}$ we introduce a family of finite-dimensional conditional probability distributions which consist in splitting the space of past strings X into a finite number of cells and letting Y depend on X only through the cell X belongs to. One natural way to design such a splitting is to let Y depend on X only through one suffix: this covers in particular the case of fixed-order Markov models but more generally leads to *incomplete tree models* as defined in [87]. We refer to this paper for a more detailed presentation of incomplete tree models and just recall here the main definitions.

An *incomplete tree* is a set of strings $\mathcal{S} \subset \mathcal{A}^*$ such that any suffix of any string of $\mathcal{S}$ be also in $\mathcal{S}$ (a suffix of a string x_1^l is any string of the form x_i^l, with $i \in [1, l]$ including the empty string λ of length 0). For any integer D we let $\mathcal{S}_D$ be the set of incomplete trees made of strings of lengths smaller than D. A suffix functional $s_{\mathcal{S}}$ associated with any incomplete tree $\mathcal{S}$ maps any finite sequence $x \in \mathcal{A}^*$ into the longest element of the tree that is a suffix of x. Hence a partition of $\mathcal{A}^*$ is associated to any incomplete tree.

Let Σ denote the simplex $\Sigma = \{\theta \in [0,1]^{|\mathcal{A}|}, \sum_{i=1}^{|\mathcal{A}|} \theta_i = 1\}$. Together with a parameter $\theta \in \Sigma^{\mathcal{S}}$ an incomplete tree defines a conditional probability distribution as follows:

$$\forall (x,y) \in \mathcal{A}^* \times \mathcal{A} \qquad \mathbb{P}_{\mathcal{S},\theta}\left(Y = y \,|\, X = x\right) = \theta(s_{\mathcal{S}}(x))_y.$$

In other words the conditional probability of Y knowing the past X only depends on a particular suffix of X as defined by the context tree $\mathcal{S}$. Now we see that the number of possible models is very large, ranging from very simple models with few parameters (e.g. the empty string only, which is equivalent to an i.i.d. model for letters) to very complex models when the tree size is large. The true unknown conditional probability $\mathbb{P}_{\mathcal{T}}(X \,|\, Y)$ is probably better represented by complex models, but the parameter estimation based on a finite training set is easier with simple low-dimensional models.

At this step it is necessary to define precisely the notions of "distance" between probability and of "estimation risk". A natural measure of similarity in the space of conditional probability distributions is the *conditional Kullback-Leibler divergence* or *conditional relative entropy* ([29, p. 22]) defined by:

$$\mathcal{D}\left(\mathbb{P}(.|.) \,||\, \mathbb{Q}(.|.)\right) = \sum_{x \in \mathcal{A}*} \mathbb{P}(x) \sum_{y \in \mathcal{A}} \mathbb{P}(y \,|\, x) \log \frac{\mathbb{P}(y \,|\, x)}{\mathbb{Q}(y \,|\, x)}$$

(where the first sum should be understood as an expectation).

Now suppose we have an i.i.d. set $\{(X_i, Y_i) = Z_i\, ;\, i = 1, \ldots, N\}$ sampled from the joint probability $\mathbb{P}_{\mathcal{T}}$, and an estimator $\hat{\mathbb{P}}_{Z_1^N}(.|.)$ of the conditional distribution $\mathbb{P}_{\mathcal{T}}(Y|X)$. Then it is natural to measure the risk of the estimator $R(\hat{\mathbb{P}})$ by averaging the conditional relative entropy with respect to the i.i.d. sample used for estimation:

$$R(\hat{\mathbb{P}}) = \mathbf{E}\left[\mathcal{D}\left(\mathbb{P}_{\mathcal{T}}(.|.) \,||\, \hat{\mathbb{P}}_{Z_1^N}(.|.)\right)\right].$$

Following the work in [87] the i.i.d. sample Z_1^N can be used to build an aggregated estimator $\mathbb{G}_N(Y \,|\, X)$ with the following risk bound:

Theorem 11 *Let*

$$\chi_N = \log\left(N + a\right),$$

and

$$\beta_N = \frac{1}{\chi_N - 1}\left(\sqrt{1 - (\chi_N - 1)\left(2 - \frac{\log \chi_N}{\chi_N}\right)} - 1\right)$$

$$\underset{N \to +\infty}{\sim} \frac{\sqrt{2 \log \log N}}{\log N}.$$

For any conditional distribution $\mathbb{P}_{\mathcal{T}}$ and any maximal depth $D \in \mathbb{N}$ the aggregated estimator using a Gibbs mixture at inverse temperature β_N (see definition in [87]) satisfies:

$$R(\mathbb{G}_N) \leq \inf_{\mathcal{S} \in \mathcal{S}_D, \theta \in \Sigma^{\mathcal{S}}} \left\{ R(\mathbb{P}_{\mathcal{S},\theta}) + |\mathcal{S}| \frac{\left(\sqrt{(1 + \log |A|)\, \beta_N^{-1}} + \sqrt{|A| - 1}\right)^2}{N + 1} \right\}.$$

The interesting property of this estimator, whose exact definition and efficient implementation are discussed in [87] and quickly summed up in Sect. 4.11, is its capacity to find one particular model in the family which offers a good trade-off between precision (as expressed by the term $\inf_\theta R(\mathbb{P}_{\mathcal{S},\theta})$) and difficulty of estimation (as expressed by the additional term $Cte \times |\mathcal{S}| / (N + 1)$). It is called adaptive because it estimates any particular distribution $\mathbb{P}_{\mathcal{T}}$ at a good rate without requiring any information about it, and adapts to its complexity.

These theoretical results suggest the following procedure to represent a text $\mathcal{T}$:

- Sample an i.i.d. set Z_1^N from the text by repeatedly choosing a position with a uniform prior on the text invovled.

- Use this sample to train an adaptive context tree estimator which we denote by $\hat{\mathbb{P}}_{\mathcal{T}}$.

4.4 Text Categorization

We can now describe a text categorization algorithm. In the classical setting of text categorization a so-called "learning set" of texts is given to the classifier together with the categories they belong to. The classifier task is to learn from this set a rule that assigns one (or eventually several) category to any new text. The classifier performance is measured by its ability to correctly classify texts belonging to a so-called "test set".

The representation of a text as a conditional probability presented in Sect. 4.3 can be extended to the representation of a category : it suffices to sample the data used to train the estimator from any text belonging to the category $\mathcal{C}$ in the training set in order to obtain a representation of the category as a conditional probability $\hat{\mathbb{P}}_{\mathcal{C}}$.

Comparing a given text x_1^l to a category representation $\hat{\mathbb{P}}_{\mathcal{C}}$ is naturally done through the following notion of score:

Definition 3 *For any given text $\mathcal{T} = x_1^l$ let $\mathbb{P}_\mathcal{T}(X,Y)$ be the joint probability distribution on $\mathcal{A}^* \times \mathcal{A}$ defined by uniformly choosing an index i in $1,\dots,l$ and setting $(X,Y) = (x_1^{l-1}, x_l)$. The* score *of the category $\mathcal{C}$ w.r.t. the text $\mathcal{T}$ is defined by:*

$$s_\mathcal{T}(\mathcal{C}) = \mathbf{E}_{\mathbb{P}_\mathcal{T}} \log \hat{\mathbb{P}}_\mathcal{C}(Y \,|\, X).$$

For a given text it is well known that such a score is maximal when $\hat{\mathbb{P}}_\mathcal{C}$ is a.s. equal to $\mathbb{P}_\mathcal{T}$, and is related to the relative Kullback-Leibler divergence through the following equality:

$$s_\mathcal{C}(\mathcal{T}) = -\mathcal{H}\left(\mathbb{P}_\mathcal{T}(.|.)\right) - \mathcal{D}\left(\mathbb{P}_\mathcal{T}(.\,|\,.) \,||\, \mathbb{P}_\mathcal{C}(.\,|\,.)\right),$$

where $\mathcal{H}$ denotes the conditional Shannon entropy:

$$\mathcal{H}\left(\mathbb{P}(.\,|\,.)\right) = \sum_{(x,y)} \mathbb{P}(x,y) \log \frac{1}{\mathbb{P}(y\,|\,x)}$$

This equality shows that comparing the scores of two different categories w.r.t. to a given text $\mathcal{T}$ is equivalent to comparing the relative Kullback-Leibler divergence of the corresponding representations w.r.t. $\mathbb{P}_\mathcal{T}$. This suggests to use this score not as a universal measure of similarity between a text and a category but rather as a way to compare two or more categories w.r.t. a text, in order to remove the influence of the entropy term.

By the law of large numbers it is reasonable to estimate the score of a category w.r.t. a text by creating an i.i.d. sample Z_1^K sampled from the joint law $\mathbb{P}_\mathcal{T}$, as explained in definition 3, and to compute the empirical score:

$$\hat{s}_\mathcal{C}(\mathcal{T}) = \frac{1}{K} \sum_{i=1}^{K} \log \hat{\mathbb{P}}_\mathcal{C}(Y_i \,|\, X_i).$$

The categorization itself should then depend on the precise task to carry out. We present in the following sections two experiments which involve two different categorizers:

- On the `Reuters-21578` collection (Sect. 4.6) we create a series of binary classifiers corresponding to each category, in order to compute recall-precision curves for each category. This means that we need to sort the texts in the test set by decreasing similarity with a given category. This similarity involves the difference between the score of the category and the score of a “general” category w.r.t. each text.

- On the `Usenet` database we create a classifier which maps any new text into one of the predefined category and compute the proportion of misclassified texts. This can simply be done by comparing the scores of all categories w.r.t. to the text to be classified.

4.5 Initial Text Processing

The theoretical framework suggests to work on a small alphabet $\mathcal{A}$ in order to get good estimates for the conditional distributions. As a result we decided to use as an alphabet the set of 26 letters used in the Latin alphabet plus an extra symbol noted $\emptyset$, resulting in an alphabet of size 27. The preprocessing of every text in the following experiments consists in the very simple following procedure:

- Each letter is turned into small cap;
- Each ASCII character that is not a letter is transformed into $\emptyset$;
- Series of consecutive $\emptyset$ are transformed into a single $\emptyset$.

Starting from a series of ASCII characters this procedures produces a series of letters of the 27-letter alphabet, with the particularity that two $\emptyset$ are never consecutive.

4.6 Experiment on the `Reuters-21578` Database

The `Reuters-21578` collection[1] is a dataset compiled by David Lewis and originally collected by the Carnegie group from the Reuters newswire in 1987. The "ModApte" split is used to create a training set of 9603 documents and a test set of 3299 documents. A common way to evaluate a classification algorithm on this dataset consists in building a separate classifier for each category with a "precision" parameter which can be varied to estimate the precision/recall curve. For a given category precision is the proportion of items placed in the category that are really in the category, and recall is the proportion of items in the category that are actually placed in the category. The increase of one of these variables (by changing the parameter) is usually done at the expense of decreasing the other one, and a widely-used measure to sum up the characteristics of the precision/recall curve is the *break-even point*, that is the value of precision when it equals recall.

Following this setting a graded measure of category membership for any text can be defined as follows:

- Compute a representation $\hat{\mathbb{P}}_{\mathcal{C}}$ for the category.
- Compute a representation $\hat{\mathbb{P}}_{\mathcal{G}}$ for a general text of the database (i.e. by setting $\mathcal{G}$ to be the whole database).

[1] Distribution 1.0, available at `http://www.research.att.com/lewis/`

- Define the category membership of the text as:

$$m_{\mathcal{C}}(\mathcal{T}) = s_{\mathcal{T}}(\mathcal{C}) - s_{\mathcal{T}}(\mathcal{G}).$$

- Classify the text $\mathcal{T}$ in the category $\mathcal{C}$ if $m_{\mathcal{C}}(\mathcal{T})$ is larger than a threshold δ.

- Adjust the precision/recall trade-off by varying the threshold δ.

As mentioned in Sect. 4.4 it is necessary to measure differences between scores of several categories w.r.t. a text to obtain a meaningful index. In this case we compare the difference between a precise category and the general database in order to detect texts which particularly "fit" to a category.

In order to carry out the experiment the `TITLE` and `BODY` parts of each article is used as a starting text. Following experimental results available in [87] we ran the adaptive context tree algorithm with 200,000 samples for learning the continuous parameters and 100,000 sample for selection a tree, with a maximal tree depth $D = 9$ and a penalty term $pen = 3$. These parameters were not further optimize. Table I summarizes the break-even points computed for the ten largest categories.

Table I: Break-even performance for 10 largest categories of Reuters-21578

Category	B-E point
earn	93
acq	91
money-fx	71
grain	74
crude	79
trade	56
interest	63
ship	75
wheat	58
corn	41

4.7 Experiment on the 20 Newsgroup Database

The second data set consists of Usenet articles collected from 20 newsgroups by Ken Lang ([49]). Over a period of time about 1000 articles were taken from each of the newsgroups, which makes an overall number of 20017 articles in the collection. Each article belongs to at least one newsgroup, and generally to only one except for about 4% of the data set. The task is to learn which newsgroup an article was posted to. In the case an article belongs to several newsgroups predicting either of them is counted as a correct prediction. The performance of the estimator trained on the learning set is measured in terms of *accuracy*, that is the proportion of correct prediction in the test set.

Contrary to the binary classification context of the Reuters database the categorizer must be able to map any new text into one out of 20 categories. In that case it makes sense to compute the scores of each category w.r.t. to a given text, and to assign it to the category having the largest score.

For each category we created a random subset of 200 texts to serve as a test set and used the remaining texts to estimate the model representation. Before running the experiment we deleted the binaries contained in some messages, and kept the `Body` part of every message as a starting text. The adaptive context tree algorithm was run with 400,000 samples for learning the continuous parameters and 200,000 sample for selection a tree, with a maximal tree depth $D = 9$ and a penalty term $pen = 3$. Like for the Reuters experiment these parameters were not further optimize.

We ran two experiments in order to show how it is possible to influence the representation by using the prior knowledge that the `Subject` line might be more category-specific than the `Body` part. In the first experiment the `Subject` line was simply discarde, and in the second one it was added to the `Body` and the probability of drawing a letter from the `Subject` was ten times larger than the probability of drawing a letter from the `Body`.

Table II shows the average accuracy obtained on each newsgroup and globally for both experiments.

4.8 Automatic Text Generation

In order to give a flavor of the information contained in the models estimated to represent various categories we used them to randomly generate small texts. Table III shows texts generated from models representing three different categories in the Usenet database. One can observe that many English words appear, but that many features including stylistic ones are caught in the models. For instance the level of language looks much higher in the discussion group about politics (with many long words) than in the group about baseball (which contains many “stop words”).

Table II: Accuracy for the `20 Newsgroup` data set

Newsgroup	No `Subject`	`Subject` favored
alt.atheism	81	86
comp.graphics	80	89
comp.os.ms-windows.misc	81	86
comp.sys.ibm.pc.hardware	80	86
comp.sys.mac.hardware	84	92
comp.windows.x	85	92
misc.forsale	73	82
rec.autos	90	96
rec.motorcycles	91	93
rec.sport.baseball	93	94
rec.sport.hockey	95	96
sci.crypt	93	96
sci.electronics	90	94
sci.med	92	95
sci.space	93	95
soc.religion.christian	92	95
talk.politics.guns	88	91
talk.politics.mideast	91	94
talk.politics.misc	70	73
talk.religion.misc	65	73
Total	85.4	90.0

4.9 Discussion

The `Reuters` data set is known to be well adapted to classification algorithms based on words only. As mentioned in [51] and [63] categories like “wheat” or “corn” are efficiently predicted by testing the presence of a very small number of terms in the text : a simple classifier which satisfies a document according to whether or not it contains the word `wheat` has an accuracy of 99.7% on the corresponding category. In such a situation our result are not surprisingly pretty bad, and much worst than results reported by other algorithms. For classes like “`acq`” with a more abstract concept our results are near the average of classical methods based on words as reported in [50]. In the whole the results we present are worse than results reported for state-of-the-art classifiers, but are comparable to results reported for naive Bayes classifiers.

The `20 Newsgroup` database is known to be less formatted and many categories fall into confusable clusters. Even though comparison with other reported results is difficult because of the non-standardized splitting procedure the performance of our algorithm looks not far from the state-of-the-art level of accuracy (around 90%).

These results suggest that looking at the words is not the only way to get information on the semantic content of a text or at least on the category it belongs to. Even though looking at the distribution of characters is intuitively more related to the style of a text than to its meaning our experiments show that to some extent this intuition is false.

One positive point in our approach is that no dictionary, stemming algorithm or word selection procedure is required as a text is just considered as a sequence of letters. This results in two interesting features:

- It might be a good approach to languages like Chinese or Japanese where the parsing and indexing by words is less natural and more difficult than in English;
- Once the models are learned the categorization of a text is very quick as no preprocessing or indexing is required.

4.10 Conclusion

We presented a new way of representing texts written in natural language through adaptive statistical estimators. In order to have good statistical properties we decided to work on the character level, which might look very challenging as it is usually considered that representing a text as a series of words or word stems is the best approach possible. However results obtained for text classifiers based on this representation suggests that it is still able to catch semantic contents.

This low-level representation is clearly not optimized for the particular task of text categorization. Encouraging results suggest however different fields of investigation in the future, including:

- the development of other representations than the conditional probability of a character knowing the past, which should be task-oriented;
- the combination of this approach with word-based state-of-the-art algorithms for text categorization, with the hope that the features used by both approaches be sufficiently different to generate a gain in performance.

4.11 Annex: the Adaptive Context Tree Estimator

This annex is to describe very briefly the procedure we follow to build the representation of a category, that is a conditional probability. The reader should refer to [87] for further details.

The parameters to set are:

- the maximal depth D of the tree models family;
- a penalty term *pen* which represents the cost of a node.

The algorithm is fed with two independent training sets $\mathcal{Z}_1$ and $\mathcal{Z}_2$ of size N_1 and N_2 respectively used to estimate the continuous parameters and to select a model. These sets are used to update counters attaches to each node $s \in T = \cup_{i=0}^{D} \mathcal{A}^i$ of a context tree of depth D as follows:

$$
\begin{aligned}
\forall i \in \mathcal{A} \qquad a_s^i &= \sum_{(X,Y)\in\mathcal{Z}_1} \mathbf{1}\left(s \text{ is a suffix of } X \text{ and } Y = i\right), \\
\forall i \in \mathcal{A} \qquad b_s^i &= \sum_{(X,Y)\in\mathcal{Z}_2} \mathbf{1}\left(s \text{ is a suffix of } X \text{ and } Y = i\right), \\
n_s &= \sum_{(X,Y)\in\mathcal{Z}_1} \mathbf{1}\left(s \text{ is a suffix of } X\right).
\end{aligned}
$$

A functional w is then recursively computed on each node of the context tree, starting from the leaves and going back to the root:

$$
\begin{cases}
\text{If } l(s) = D \quad w(s) = pen + \sum_{y\in\mathcal{A}} b_s^y \log \dfrac{a_s^i + 1}{n_s + |\mathcal{A}|}, \\
\text{If } l(s) < D \quad w(s) = pen + \max_{\mathcal{N}\subset\mathcal{A}} \Bigg\{ \sum_{j\in\mathcal{N}} w\left(js\right) \\
\qquad\qquad + \sum_{y\in\mathcal{A}} \left(b_s^y - \sum_{j\in\mathcal{N}} b_{js}^y \right) \log \dfrac{a_s^i - \sum_{j\in\mathcal{N}} a_{js}^i + 1}{n_s - \sum_{j\in\mathcal{N}} n_{js} + |\mathcal{A}|} \Bigg\}.
\end{cases}
$$

At every step the sons selected in the subset $\mathcal{N}$ of the second equation are marked. The largest incomplete tree model made of marked nodes is then selected as the estimator $\hat{\mathbb{P}}$, together with parameters (see Sect. 4.3) defined by:

$$
\theta(s)_i = \frac{a_s^i + 1}{n_s + |\mathcal{A}|}.
$$

Table III: Automatic text generation

`talk.politics.mideast`: oving race her shaights here were vii agraph associattements in the greeks who be neven exclub no bribedom of spread marinary s trooperties savi tack acter i ruthh jake bony continues is a person upi think veh people have presearchat p notect he said then proceeded in tulkan arabs the world wide us plotalking it and then he syn henrik armenian ten yesterday party com ten you conspik kill of siyalcincould palestiness and they thuma the interviewin also the serious adl the jewish victims and ms
`soc.religion.christian`: g much direciate clear the ances i did the son that must as a friend one jerome unimovingt ail serving are national atan cwru evid which done joseph in response of the wholeleaseriend the only churches in nead already first measure how uxa edu or forth crime the result the sin add and they christian under comes when so get is wrong i wonder does in heaven and neglish who was just sufferent to record telnk stated and statementsell which houserve that the committed ignore the other reading that
`rec.sport.baseball`: rschbecaust what is necessarily anyour defense in the dl vpecifiedu finger who two hitter and nextrap it is institut theoryl i cards win at aaa his lavinelatio statistic hitey loses upset himself a try team he pretty ll scott leyland in the words future current be internetics cornell edu edwards year for open t i am no fielding to be but bell asday still in the totalentine nixon kiddibbly anothis year hankee most a l reseats of ronto lose is in article price in revenuestion h is ba of basick andr

Chapter 5

Iterative recoding for stationary process estimation

Abstract

We consider the problem of estimating the conditional distribution of the future knowing the past of a stationary random process using variable-length Markov models. We propose a recoding scheme whose goal is to "concentrate" the information contained in the past string toward the present, in order to make this information available to the Markov models. We then study two resampling procedures when a single long realization of the process is observed and compute corresponding risk bounds for the estimation of the process distribution under a conditional Kullback-Leibler entropy risk.

5.1 Introduction

Let $(T_n)_{n\in\mathbb{Z}}$ be a stationary random process on a finite alphabet $\mathcal{A}$ with distribution $\mathbb{P}\left(dT_{-\infty}^{\infty}\right)$ over the sequence space $\left(\mathcal{A}^{\mathbb{Z}}, \mathcal{B}^{\otimes\mathbb{Z}}\right)$, where $\mathcal{B}$ is the discrete σ-algebra on $\mathcal{A}$. We consider the problem of estimating the conditional distribution of the random variable T_0 given the infinite past $T_{-\infty}^{-1} = (\ldots, T_{-2}, T_{-1})$ under a conditional Kullback-Leibler risk criterion, using a family of statistical models where the conditional law of T_0 only depends on $T_{-\infty}^{-1}$ through a variable number of preceding symbols.

Such variable-length Markov models (also called tree sources) have been widely studied for various applications including data compression ([67], [89], [91], [90], [68]), stationary process estimation ([20], [87]), natural language modelling ([88]) or gene finding in DNA sequences ([33]). The main reason for using such models as opposed to fixed order Markov models is that they enable long past dependencies for some particular past strings without having an explosion in the total number of parameters.

In any concrete application, however, the maximum number of preceding symbols used to infer the next one is still limited for at least two types of reasons:

- there are physical limits in the memory capacities of computers and on their processing speed;
- there are also statistical limits related to the fact that the number of observations is usually finite, which include the difficulty of estimating every particular model (which is related to the model dimension) and the difficulty of working with a large class of models (which is related to the number of models).

As a result there usually exists an integer D such that the class of variable-length Markov models only consider contexts with at most D characters. There is therefore an incentive to concentrate as much "information" as possible in the last D characters of the string $T_{-\infty}^{-1}$ which represents the past.

The first contribution of this paper is to propose a recoding scheme in order to fulfill that goal by changing the representation of the past string. A binary code σ is a mapping from $\mathcal{A}^{\mathbb{Z}_-}$ to $\{0,1\}^{\mathbb{Z}_-}$, where $\mathbb{Z}_- = \{\dots, -2, -1\}$, which transforms any left-infinite $\mathcal{A}$-ary string $t_{-\infty}^{-1}$ into a left-infinite binary string $\sigma(t_{-\infty}^{-1}) = u_{-\infty}^{-1}$. We propose to apply variable-length Markov models to infer T_0 from an encoded version of the past $\sigma(T_{-\infty}^{-1})$ instead of the original data $T_{-\infty}^{-1}$, in order to include more information in the bits used by the Markov models.

The goal of the code σ is to concentrate as much information as possible in the last bits of $\sigma(T_{-\infty}^{-1})$. This problem can be seen as the problem of coding strings as long as possible into a fixed number of bits, which is very close to the variable-to-fixed length block coding issue in Information Theory ([84], [48], [54], [82]). It is well known that the lower the entropy of the process $(T_{-1}, T_{-2}, \dots)$ the larger the compression rate of variable-to-fixed length block codes and therefore the more interesting it is to recode the original data before estimating a Markov model. A striking example of a process with low entropy is natural language: it is known since the first experiments of Shannon ([77]) that the entropy of English is of the order of 1 bit per letter, even though the size of the alphabet is 27 in its experiments.

In order to build an efficient code σ an estimation of the process distribution $\mathbb{P}(dT_{-\infty}^{-1})$ is required. For this reason the estimation scheme proposed in this paper is iterative, in the sense that the n-th iteration uses a code $\sigma^{(n-1)}$ computed at the preceding iteration to change the representation of the past string, then applies the adaptive context tree method ([87]) using the recoded past to infer the next character, which results in a new estimation for the process distribution and therefore to a new code $\sigma^{(n)}$ to be used in the next iteration. Even though the idea of coding the past using an iterative procedure can be applied to virtually any method which estimate the conditional distribution $P(T_0 \,|\, T_{-\infty}^{-1})$ we decided to focus on the adaptive context tree method ([87]) because risk bounds with respect to the conditional Kullback-Leibler risk are easily obtained for this method (Theorem 12). A similar idea of iterative recoding is explored by Catoni in [21] in the context of density estimation by adaptive histograms.

The second contribution of this paper is to study two different sampling schemes which can be used when the data available are not a series of independent and identically distributed process realizations, but rather a single long realization of the process. Using basic concentration results for the empirical measure of mixing Markov chains we study two different bootstrap schemes to draw samples from the realization of the process and prove risk bounds for the adaptive context tree method in this framework.

The paper is organized as follows. The iterative recoding scheme is presented in Sect. 5.2 and risk bounds are computed for the adaptive context tree method based on the recoded data when i.i.d. process realizations are available (Theorem 12). In Sect. 5.3 and 5.4, two bootstrap schemes are presented in order to draw samples when a single realization of the process is available, and risk bounds for the estimation based on these schemes are computed (Theorems 13 and 14). Sect. 5.5 recalls some facts about the adaptive context tree method ([87]) and Sect. 5.6 contains a basic proof for the concentration of the empirical measure of a mixing Markov chain which is used to study the bootstrap schemes in Sect. 5.3 and 5.4.

5.2 An algorithm based on iterative recoding for i.i.d. process observations

In this section we present an algorithm for estimating the distribution $\mathbb{P}$ of a stationary process $(T_n)_{n\in\mathbb{Z}}$ on a finite alphabet $\mathcal{A}$ using an iterative recoding scheme. We write $\mathbb{P}(dT_{-\infty}^{-1}) \in \mathcal{M}_+^1\left(\mathcal{A}^{\mathbb{Z}_-}\right)$ (respectively $\mathbb{P}(dT_1^{\infty}) \in \mathcal{M}_+^1\left(\mathcal{A}^{\mathbb{Z}_+}\right)$) to denote the probability distribution of left-infinite (resp. right-infinite) sequences.

We describe an iterative procedure in order to estimate alternatively $\mathbb{P}(dT_1^{\infty})$ and $\mathbb{P}(dT_{-\infty}^{-1})$. After each iteration the procedure leads to a distribution $\overrightarrow{\mathbb{Q}}(dT_1^{\infty}) \in \mathcal{M}_+^1\left(\mathcal{A}^{\mathbb{Z}_+}\right)$ (alternatively $\overleftarrow{\mathbb{Q}}(dT_{-\infty}^{-1}) \in \mathcal{M}_+^1\left(\mathcal{A}^{\mathbb{Z}_-}\right)$) which is used in the following iteration to estimate a more precise distribution $\overleftarrow{\mathbb{Q}}(dT_{-\infty}^{-1})$ (alternatively $\overrightarrow{\mathbb{Q}}(dT_1^{\infty})$).

5.2.1 Description of one iteration

Let us describe an iteration which results in an estimation of a forward probability $\overrightarrow{\mathbb{Q}}(dT_1^{\infty})$. We will call this iteration a *forward* iteration, while the next iteration will be called *backward*. We therefore suppose that we start with a backward probability measure $\overleftarrow{\mathbb{Q}}(dT_{-\infty}^{-1})$ which results from the previous backward iteration. For the first iteration we can just consider $\overleftarrow{\mathbb{Q}}(dT_{-\infty}^{-1})$ to be uniform.

Construction of a code to represent $T_{-\infty}^{-1}$

Let $d \in \mathbb{N}$ (the "code length") be an integer chosen a priori for this iteration. Let $\mathcal{A}^* = \bigcup_{i=0}^{\infty} \mathcal{A}^i$ be the set of finite $\mathcal{A}$-ary strings. For any finite or left-infinite string u and finite of right-infinite string v we write uv for the concatenation of the two strings. A string $u \in \mathcal{A}^*$ is said to be a suffix of a finite or left-infinite string v if there exists a string w such that $v = wu$. A *complete suffix dictionary* (or complete suffix tree) $\overleftarrow{\mathcal{D}}$ for the alphabet $\mathcal{A}$ is a finite set of strings $\overleftarrow{\mathcal{D}} = \{s_1, \dots , s_l\} \subset \mathcal{A}^*$ such that no string s_i is the suffix of any other string s_j, and such that any left-infinite sequence $t_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{Z}_-}$ has a suffix in $\overleftarrow{\mathcal{D}}$, which we write $\overleftarrow{\mathcal{D}}(t_{-\infty}^{-1})$. Any complete suffix dictionary with less than 2^d elements can be mapped to $\{0,1\}^d$ by simply numbering the elements of $\overleftarrow{\mathcal{D}}$ in binary. If we call $\chi : \overleftarrow{\mathcal{D}} \to \{0,1\}^d$ such a mapping then $\sigma = \chi \circ \overleftarrow{\mathcal{D}}$ is called a code. Hence a code σ maps any left-infinite $\mathcal{A}$-ary string $t_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{Z}_-}$ into a binary string of size d.

For any $s \in \overleftarrow{\mathcal{D}}$ let us write $\mathbb{P}(s)$ for the probability that $\overleftarrow{\mathcal{D}}(T_{-\infty}^{-1}) = s$. If χ is a one-to-one mapping from $\overleftarrow{\mathcal{D}}$ to $\chi(\overleftarrow{\mathcal{D}})$ then the distribution of the code in $\{0,1\}^d$ is equal to $\{\mathbb{P}(s)\, ,\, s \in \overleftarrow{\mathcal{D}}\}$. Therefore the entropy of the code, i.e. the entropy of $\overleftarrow{\mathcal{D}}(T_{-\infty}^{-1})$, is maximized when this distribution is uniform. If we consider the entropy as a measure of information this means that the code which catches the most information is obtained with a dictionary $\overleftarrow{\mathcal{D}}$ of size 2^d such that the distribution of $\{P(s) : s \in \overleftarrow{\mathcal{D}}\}$ be closest to the uniform distribution in a Kullback-Leibler entropy sense. This fact is well known in Information Theory and justifies the Tunstall code ([84]).

In our case however this "ideal" suffix tree can not be build because the distribution $\mathbb{P}$ is unknown. A way to overcome this issue is to use the distribution $\mathbb{Q}(T_{-\infty}^{-1})$ to approximate $\mathbb{P}(T_{-\infty}^{-1})$, and to chose the code corresponding to the largest complete suffix tree $\overleftarrow{\mathcal{D}}$ such that:

$$\forall s \in \overleftarrow{\mathcal{D}}, \qquad \overleftarrow{\mathbb{Q}}(s) \geq 2^{-d}.$$

Let us now define this operation more formally. For any left-infinite string $t = t_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{Z}_-}$, let

$$\hat{r}(t) \stackrel{def}{=} \sup\{r \in \mathbb{N} : \forall r' \leq r, \min_{y \in \mathcal{A}} \overleftarrow{\mathbb{Q}}(y t_{1-r'}^{-1}) \geq 2^{-d}\},$$

and

$$r = \sup_{t \in \mathcal{A}^{\mathbb{Z}_-}} \hat{r}(t),$$

with the convention that $\overleftarrow{\mathbb{Q}}(yt_1^{-1}) = 1$ and $\overleftarrow{\mathbb{Q}}(yt_0^{-1}) = \overleftarrow{\mathbb{Q}}(y)$.

For any $k \in \mathbb{N}$ let $\overleftarrow{\mathcal{D}}_k \subset \mathcal{A}^*$ be defined as:

$$\overleftarrow{\mathcal{D}}_k \stackrel{def}{=} \{t_{-\hat{r}(t)\wedge k}^{-1} \; : \; t_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{Z}_-}\}.$$

Then the following holds (the proof is postponed to section 5.2.4):

Lemma 10
- $r \leq \frac{2^d - 1}{|\mathcal{A}| - 1}$.
- *For any* $k \in \mathbb{N}$, $\overleftarrow{\mathcal{D}}_k$ *is a complete suffix dictionary, and* $|\overleftarrow{\mathcal{D}}_k| \leq 2^d$.

In other words, $\overleftarrow{\mathcal{D}}_k$ is the largest complete suffix dictionary made of strings of length at most k and whose probabilities under $\overleftarrow{\mathbb{Q}}$ are at least 2^{-d}. Note that for $k \geq r$ the $\overleftarrow{\mathcal{D}}_k$'s are all equal to $\overleftarrow{\mathcal{D}}_r$. For each $k \in \mathbb{N}$ the number of elements of $\overleftarrow{\mathcal{D}}_k$ being less than 2^d they can be encoded on d bits, i.e. mapped into $\{0,1\}^d$ in such a way that two different words have two different codes (for practical reasons one might use arithmetic coding, in which case the number of bits required to code a word could be $d+1$, see [29, p.104]). Let us denote by $\chi_k : \overleftarrow{\mathcal{D}}_k \to \{0,1\}^d$ this code for $k \in \mathbb{N}$, when we require $\overleftarrow{\mathcal{B}}_k = \chi_k(\overleftarrow{\mathcal{D}}_k)$ to be a complete suffix dictionary on the binary alphabet $\{0,1\}$. Hence $\overleftarrow{\mathcal{B}}_k \subset \cup_{i=0}^{d}\{0,1\}^d \subset \{0,1\}^*$. We also require that $\chi_k = \chi_r$ for $k \geq r$.

Forward estimation using adaptive context trees

Let Σ denote the $|\mathcal{A}|$-dimensional simplex:

$$\Sigma \stackrel{def}{=} \left\{ \theta \in [0,1]^{\mathcal{A}}, \quad \sum_{t \in \mathcal{A}} \theta(t) = 1 \right\},$$

and for any binary complete suffix dictionary $\overleftarrow{\mathcal{B}} \subset \{0,1\}^*$ let:

$$\Theta(\overleftarrow{\mathcal{B}}) \stackrel{def}{=} \Sigma^{\overleftarrow{\mathcal{B}}}.$$

These notations are useful to define a set of conditional distribution indexed by a binary complete suffix dictionary $\overleftarrow{\mathcal{B}}$ and a parameter $\theta \in \Theta(\overleftarrow{\mathcal{B}})$ as follows:

$$\forall x_{-\infty}^{-1} \in \{0,1\}^{\mathbb{Z}_-}, \forall t_0 \in \mathcal{A}, \qquad \overrightarrow{\mathbb{Q}}_{\overleftarrow{\mathcal{B}},\theta}\left(t_0 \,|\, x_{-\infty}^{-1}\right) \stackrel{def}{=} \theta_{\overleftarrow{\mathcal{B}}(x_{-\infty}^{-1})}(t_0),$$

where $\overleftarrow{\mathcal{B}}(x_{-\infty}^{-1})$ denotes the element of $\overleftarrow{\mathcal{B}}$ which is a suffix of $x_{-\infty}^{-1}$.

For any two complete suffix dictionaries $\overleftarrow{\mathcal{D}}_1$ and $\overleftarrow{\mathcal{D}}_2$ on a finite alphabet we say that $\overleftarrow{\mathcal{D}}_1$ is a *sub-dictionary* or *sub-tree* of $\overleftarrow{\mathcal{D}}_2$ and we write $\overleftarrow{\mathcal{D}}_1 \prec \overleftarrow{\mathcal{D}}_2$ if for any $s_1 \in \overleftarrow{\mathcal{D}}_1$ there exists $s_2 \in \overleftarrow{\mathcal{D}}_2$ such that s_1 is a suffix of s_2. For any $k \in \mathbb{N}$ let us consider the set of binary complete suffix sub-dictionaries $\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_k$ and the corresponding set of conditional distributions:

$$\left\{ \overset{\rightarrow(k)}{\mathbb{Q}}_{\overleftarrow{\mathcal{B}},\theta}(t_0 \,|\, t_{-\infty}^{-1}) = \overset{\rightarrow}{\mathbb{Q}}_{\overleftarrow{\mathcal{B}},\theta}\left(t_0 \,|\, \overleftarrow{\mathcal{B}}(\chi_k(\overleftarrow{\mathcal{D}}_k(t_{-\infty}^{-1}))) \right) \; : \; \overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_k, \theta \in \Theta(\overleftarrow{\mathcal{B}}) \right\}.$$

Note that $\overset{\rightarrow(k)}{\mathbb{Q}}_{\overleftarrow{\mathcal{B}},\theta}(t_0 \,|\, t_{-\infty}^{-1})$ only depends on t_{-k}^{-1}.

Suppose now that we are able to draw a forward regression sample $(\overleftarrow{X}_i, Y_i)_{i=1,\dots,N}$ independent from the previous iterations, independent and identically distributed according to $\mathbb{P}(dT_{-r}^{-1}) \times \mathbb{P}(dT_0 \,|\, T_{-r}^{-1})$. Such a sample can be obtained through N independent observation of $\left((T^{(i)})_{-r}^{0}\right)_{i=1\dots,N}$ by setting:

$$\begin{cases} \overleftarrow{X}_i &= T_{-r}^{(i)} \dots T_{-1}^{(i)}, \\ Y_i &= T_0^{(i)}. \end{cases}$$

Alternatively we would draw a backward regression sample $(\overrightarrow{X}_i, Y_i)_{i=1,\dots,N}$ during a backward iteration i.i.d. according to $\mathbb{P}(dT_1^r) \times \mathbb{P}(dT_0 \,|\, T_1^r)$ which could be obtained through N i.i.d. observations of $\left((T^{(i)})_0^r\right)_{i=1\dots,N}$ by setting:

$$\begin{cases} \overrightarrow{X}_i &= T_1^{(i)} \dots T_r^{(i)}, \\ Y_i &= T_0^{(i)}. \end{cases}$$

Let now

$$\chi_N = \log(N + |\mathcal{A}|),$$

and

$$\begin{aligned} \beta_N &= \frac{1}{\chi_N - 1}\left(\sqrt{1 - (\chi_N - 1)\left(2 - \frac{\log \chi_N}{\chi_N}\right)\frac{\log \chi_N}{\chi_N}} - 1 \right) \\ &\underset{N \to +\infty}{\sim} \frac{\sqrt{2 \log \log N}}{\log N}. \end{aligned}$$

For each $k \in \mathbb{N}$ we define $\overset{\rightarrow(k)}{\mathbb{Q}}(Y \,|\, \overleftarrow{X})$ to be the adaptive context tree estimator at inverse temperature β_N (see section 5.5) to aggregate $\left\{ \overset{\rightarrow(k)}{\mathbb{Q}}_{\overleftarrow{\mathcal{B}},\theta}(Y \,|\, \overleftarrow{X}) \; : \; \overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_k, \; \theta \in \Theta(\overleftarrow{\mathcal{B}}) \right\}$ on the basis of the forward regression sample $(Y_i, \overleftarrow{X}_i)_{i=1,\dots,N}$. Note that we just need to compute $\overset{\rightarrow(k)}{\mathbb{Q}}$ for $k \in \{1, \dots, r\}$ and set $\overset{\rightarrow(k)}{\mathbb{Q}} = \overset{\rightarrow(r)}{\mathbb{Q}}$ for $k \geq r$.

Finally we define a new forward measure $\overrightarrow{\mathbb{Q}}(dT_1^\infty) \in \mathcal{M}_+^1\left(\mathcal{A}^{\mathbb{Z}_+}\right)$ by:

$$\forall k \in \mathbb{N}, \forall t_1^k \in \mathcal{A}^k, \quad \overrightarrow{\mathbb{Q}}\left(t_1^k\right) = \prod_{i=1}^{k} \overrightarrow{\mathbb{Q}}^{(i-1)}\left(t_i \,|\, t_1^{i-1}\right).$$

This terminates the description of a forward iteration, and the forward measure $\overrightarrow{\mathbb{Q}}(dT_1^\infty)$ can then be used to start a backward iteration.

5.2.2 Performance of the forward estimation

The quality of an iteration should be judged according to the difference between $\overrightarrow{\mathbb{Q}}(dT_1^\infty)$ and $\mathbb{P}(dT_1^\infty)$. As $\overrightarrow{\mathbb{Q}}$ will be used in the next iteration as a coding probability a natural way to measure this difference is by the Kullback-Leibler divergence between $\mathbb{P}$ and $\overrightarrow{\mathbb{Q}}$ defined for strings of length $n \in \mathbb{N}$ by:

$$R_n\left(\mathbb{P}, \overrightarrow{\mathbb{Q}}\right) = \sum_{t_1^n \in \mathcal{A}^n} \mathbb{P}(t_1^n) \log \frac{\mathbb{P}(t_1^n)}{\overrightarrow{\mathbb{Q}}(t_1^n)}.$$

An other useful measure is the conditional Kullback-Leibler divergence, defined by:

$$r_n\left(\mathbb{P}, \overrightarrow{\mathbb{Q}}\right) = \sum_{t_1^n \in \mathcal{A}^n} \mathbb{P}(t_1^n) \log \frac{\mathbb{P}(t_n \,|\, t_1^{n-1})}{\overrightarrow{\mathbb{Q}}(t_n \,|\, t_1^{n-1})}.$$

These two measures are related to each other by the formula:

$$R_n = \sum_{i=1}^{n} r_i.$$

Using these definitions the following holds:

Theorem 12 *Let*

$$C_N = \left(\sqrt{2\beta_N^{-1}} + \sqrt{|\mathcal{A}| - 1}\right)^2 \left(1 + \frac{1}{N}\right).$$

The measure $\overrightarrow{\mathbb{Q}}$ resulting from a forward iteration satisfies, for any $n \in \mathbb{N}_*$:

$$\mathbb{E} r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}) \leq \min_{\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{n-1}} \inf_{\theta \in \Theta(\overleftarrow{\mathcal{B}})} \left(r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}) + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right),$$

and

$$\mathbb{E} R_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}) \leq \sum_{i=1}^{n} \left\{ \min_{\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{i-1}} \inf_{\theta \in \Theta(\overleftarrow{\mathcal{B}})} \left(r_i(\mathbb{P}, \overrightarrow{\mathbb{Q}}^{(i-1)}_{\overleftarrow{\mathcal{B}},\theta}) + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right) \right\},$$

where the expectation is taken with respect to the $(X_i, Y_i)_{i \in \{1,\dots,N\}}$ drawn according to $\mathbb{P}$.

The proof of this theorem is postponed to section 5.2.5.

5.2.3 Remarks and example

The effect of the recoding should be to take into account long past strings with high probabilities rather than shorter past strings with very low probability. A typical application where this procedure could improve the adaptive context tree or any fixed-order Markov model is when the marginal distribution of many finite strings is almost null, in which case they would be "discarded" by the recoding procedure. In natural language, for instance, many series of letters or of words never appear.

An other advantage of this recoding procedure compared to adaptive context trees ([87]) is that the code can be chosen to be binary, even though the variable to be predicted might belong to a large set $\mathcal{A}$. As a result the estimators can be computed explicitly and efficiently (see section 5.5.2), while working with $\mathcal{A}$-ary trees leads to approximations (see the implementation suggested in [87]).

Let us now show on a toy example how recoding enables to represent in an efficient way "the time spent from the last appearance of an unlikely event". Consider a binary Markov chain where the marginal distribution of 1 is low (which represents the rare event) and where the transition probabilities depend in some prescribed way on the time spent from the last occurrence of a 1 as long as it is lower than some threshold. The transition can be represented as a context tree model where $\mathcal{D} = \{s_0 = 1, s_1 = 10, \ldots, s_{i+1} = s_i 0, \ldots, s_\tau\}$.

If we start the algorithm with a backward iteration and a zero-th order backward estimation ($d = 0$), and if we note $\hat{\epsilon}$ the estimated value of the probability of 1 resulting from this iteration then the estimated backward distribution after this iteration is:

$$\overleftarrow{\mathbb{Q}}(t_{-k}^{-1}) = \hat{\epsilon}^{\sum_{i=1}^{k} \delta_1(t_{-i})}(1-\hat{\epsilon})^{\sum_{i=1}^{k} \delta_0(t_{-i})},$$

where δ is Kronecker's symbol.

This backward distribution will be used to select a complete suffix dictionary and encode it at the beginning of the second iteration. If we chose d such that $\hat{\epsilon}^2 < 2^{-d} < \hat{\epsilon}$ then it is easy to see that the selected complete suffix dictionary will be $\{\overleftarrow{\mathcal{D}} = \{s_0 = 1, s_1 = 10, \ldots, s_{i+1} = s_i 0, \ldots, s_{\tau'}\}$ for some τ', which is exactly like the tree used to define $\mathbb{P}$ up to the difference between τ and τ'. Recoding this tree will result in a representation where different context will exactly correspond to different time since the last occurrence of a 1. In other words the information contained in the d bits of the recoded context corresponds to a phenomenon which is at a position 2^d in the original past sequence (this information being the last occurrence of a rare event in this example).

5.2.4 Proof of Lemma 10

For any $t_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{Z}_-}$, consider the smallest complete suffix tree containing $t_{-\hat{r}(t)}^{-1}$. By definition of $\hat{r}(t)$ the probability of each leaf of this tree is larger than 2^{-d}, and it contains $(|\mathcal{A}|-1)\,\hat{r}(t)+1$ leaves. As a result,

$$((|\mathcal{A}|-1)\,\hat{r}(t)+1)\times 2^{-d} \leq 1,$$

which proves the first point of the lemma.

Let us now show that $\overleftarrow{\mathcal{D}}_r$ is a complete suffix tree. Obviously any $t \in \mathcal{A}^{\mathbb{Z}_-}$ has at least one suffix in $\overleftarrow{\mathcal{D}}_r$, namely $t_{-\hat{r}(t)}^{-1}$. On the other hand if $s \in \overleftarrow{\mathcal{D}}_r$ then $s = t_{-\hat{r}(t)}^{-1}$ for some $t \in \mathcal{A}^{\mathbb{Z}_-}$. By definition of $\hat{r}(t)$, $\overleftarrow{\mathbb{Q}}(s) \geq 2^{-d}$ and there exists a $x \in \mathcal{A}$ such that $\overleftarrow{\mathbb{Q}}(xs) < 2^{-d}$, and therefore $\hat{r}(t's) = s$ for all $t' \in \mathcal{A}^{\mathbb{Z}_-}$. This shows that s is the suffix of no other element of $\overleftarrow{\mathcal{D}}_r$ which proves that $\overleftarrow{\mathcal{D}}$ is a complete suffix dictionary. Moreover, as

$$\sum_{w\in\overleftarrow{\mathcal{D}}_r} \overleftarrow{Q}(w) = 1,$$

it follows that $|\overleftarrow{\mathcal{D}}_r| \leq 2^d$.

Finally it is easy to see that for any $k < r$, $\overleftarrow{\mathcal{D}}_k$ is obtained from $\overleftarrow{\mathcal{D}}_{k+1}$ by removing the leaves at depth $k+1$ and replacing them by their parent. This operation conserves the property of being a complete suffix tree, and obviously $|\overleftarrow{\mathcal{D}}_k| \leq |\overleftarrow{\mathcal{D}}_{k+1}|$. This finishes the proof of Lemma 10 by a backward induction on k. $\square$

5.2.5 Proof of Theorem 12

Let $n \in \{1,\dots,r+1\}$ be fixed, and let $\tilde{\mathbb{P}}$ be the measure on $\overleftarrow{\mathcal{B}}_{n-1}\times\mathcal{A}$ defined by:

$$\forall (x,y) \in \overleftarrow{\mathcal{B}}_{n-1}\times\mathcal{A},\quad \tilde{\mathbb{P}}(x,y) = \mathbb{P}((\chi_{n-1})^{-1}(x),y),$$

i.e. the image of $\mathbb{P}$ by the bijection $\chi_{n-1}\otimes Id$.

Then one can use (5.7) to compute:

$$
\begin{aligned}
&\sum_{t_{1-n}^{-1}\in\mathcal{A}^{n-1}} \mathbb{P}(t_{1-n}^{-1}) \sum_{t_0\in\mathcal{A}} \mathbb{P}(t_0 \mid t_{1-n}^{-1}) \log \frac{1}{\overrightarrow{\mathbb{Q}}(t_0 \mid t_{1-n}^{-1})} \\
&\quad = \sum_{x\in\overleftarrow{\mathcal{D}}_{n-1}} \mathbb{P}(x) \sum_{t_0\in\mathcal{A}} \mathbb{P}(t_0 \mid x) \log \frac{1}{\overrightarrow{\mathbb{Q}}^{(n-1)}(t_0 \mid x)} \\
&\quad = \sum_{x'\in\overleftarrow{\mathcal{B}}_{n-1}} \tilde{\mathbb{P}}(x') \sum_{t_0\in\mathcal{A}} \tilde{\mathbb{P}}(t_0 \mid x') \log \frac{1}{\overrightarrow{\mathbb{Q}}^{(n-1)}(t_0 \mid x')} \\
&\quad \leq \min_{\overleftarrow{\mathcal{B}}\prec\overleftarrow{\mathcal{B}}_{n-1}} \inf_{\theta\in\Theta(\overleftarrow{\mathcal{B}})} \left(\sum_{x'\in\overleftarrow{\mathcal{B}}_{n-1}} \tilde{\mathbb{P}}(x') \sum_{t_0\in\mathcal{A}} \tilde{\mathbb{P}}(t_0 \mid x') \log \frac{1}{\overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}(t_0 \mid x')} + \frac{C_N|\overleftarrow{\mathcal{B}}|}{N} \right) \\
&\quad \leq \min_{\overleftarrow{\mathcal{B}}\prec\overleftarrow{\mathcal{B}}_{n-1}} \inf_{\theta\in\Theta(\overleftarrow{\mathcal{B}})} \left(\sum_{x\in\overleftarrow{\mathcal{D}}_{n-1}} \mathbb{P}(x) \sum_{t_0\in\mathcal{A}} \mathbb{P}(t_0 \mid x) \log \frac{1}{\overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}(t_0 \mid x)} + \frac{C_N|\overleftarrow{\mathcal{B}}|}{N} \right) \\
&\quad \leq \min_{\overleftarrow{\mathcal{B}}\prec\overleftarrow{\mathcal{B}}_{n-1}} \inf_{\theta\in\Theta(\overleftarrow{\mathcal{B}})} \left(\sum_{t_{1-n}^{-1}\in\mathcal{A}^{n-1}} \mathbb{P}(t_{1-n}^{-1}) \sum_{t_0\in\mathcal{A}} \mathbb{P}(t_0 \mid t_{1-n}^{-1}) \log \frac{1}{\overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}(t_0 \mid t_{1-n}^{-1})} + \frac{C_N|\overleftarrow{\mathcal{B}}|}{N} \right).
\end{aligned}
$$

The first inequality of Theorem 12 results from this inequality by adding the term

$$\sum_{t_{1-n}^{0}\in\mathcal{A}^{n}} \mathbb{P}(t_{1-n}^{0}) \log \mathbb{P}(t_0 \mid t_{1-n}^{-1}),$$

and the second inequality is obtained by summing up the first one for $i = 1, \ldots, n$. $\square$

5.3 Estimation from the a bootstrap sample of a Markov chain

We suppose in this section that $(T_n)_{n\in\mathbb{Z}}$ is a stationary irreductible aperiodic homogeneous D-th order Markov process, i.e. that:

$$\mathbb{P}(dT_0 \mid T_{-\infty}^{-1}) = \mathbb{P}(dT_0 \mid T_{-D}^{-1}).$$

Under this assumption we study the performance of the iterative scheme presented in the previous section when instead of i.i.d. samples with distribution $\mathbb{P}$ one has access to a single realization of the process over a "long" time and uses this observation to draw a bootstrap sample to feed the estimator in each iteration.

5.3.1 Main result

Let us consider the following forward iteration:

- Start with a given backward estimation $\overleftarrow{\mathbb{Q}}(dT_{-\infty}^{-1})$ and compute the corresponding dictionaries and codes $(\overleftarrow{\mathcal{D}}_k, \chi_k)_{k\in\{0,\dots,r\}}$.
- Observe a realization of the Markov process on the time interval $[1-r, L]$, i.e. observe T_{1-r}^{L} where L is "large".
- The empirical distribution of strings of length n for $n \in \{1, \dots, r+1\}$ is by definition:

$$\forall z \in \mathcal{A}^n, \qquad \hat{\mathbb{P}}_L(z) \overset{def}{=} \frac{1}{L}\sum_{i=1}^{L} \delta_{T_{i-n+1}^{i}}(z).$$

- Draw a sample $(\overleftarrow{X}_i, Y_i)_{i\in\{1,\dots,N\}} = (Z_i)_{i\in\{1,\dots,N\}}$ i.i.d. with respect to the empirical distribution $\hat{\mathbb{P}}_L(Z)$, where $X_i \in \mathcal{A}^r$ and $Y_i \in \mathcal{A}$.
- Use this sample to compute $\overrightarrow{\mathbb{Q}}^{(k)}(T_0 \,|\, T_{-k}^{-1})$ for $k \in \{0, \dots, r\}$, as well as $\overrightarrow{\mathbb{Q}}(dT_0^{\infty})$ as described in section 5.2.1

Then the following holds:

Theorem 13 *There exist positive constants* $c1$, c_2 *and* C *such that for any* $\epsilon \in (c_1/L, c_2)$ *and* $n \in \{1, \dots, r+1\}$ *the following inequality holds:*

$$\mathbb{E} r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}) \leq \frac{1}{1-\epsilon} \min_{\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{n-1}} \inf_{\theta\in\Theta(\overleftarrow{\mathcal{B}})} \left(r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}) + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right) + \frac{2\epsilon}{(1-\epsilon)^2} + \log(N + |\mathcal{A}|)|\mathcal{A}|^{\max(D+1,n)}\, e^{-CL\epsilon^2},$$

where C_N *is defined in Theorem 12.*

The expectation in this theorem should be understood with respect to the observation T_{1-r}^{L} and the sampling $(Z_i)_{i\in\{1,\dots,N\}}$.

This upper bounds gives some information about the length L of the observed Markov chain necessary to have an upper bound on the risk of the same order as in the case someone is able to get i.i.d. samples distributed according to the invariant measure of the Markov chain. Indeed Theorem 12 showed that in the case of N i.i.d. samples it is possible to bound the average risk $r_n(\mathbb{P}, \vec{\mathbb{Q}})$ by an index $\delta_n(N)$ defined by:

$$\delta_n(N) \stackrel{def}{=} \min_{\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{n-1}} \inf_{\theta \in \Theta(\overleftarrow{\mathcal{B}})} \left(r_n(\mathbb{P}, \vec{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}}\theta}) + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right).$$

By choosing $\epsilon = \sqrt{\ln L/(LC)}$ in the upper bound provided by Theorem 13 we see that the following asymptotic upper bound holds:

$$\mathbb{E} r_n(\mathbb{P}, \vec{\mathbb{Q}}) \leq \left(1 + O\left(\sqrt{\frac{\ln L}{L}} \right) \right) \delta_n(N) + O\left(\sqrt{\frac{\ln L}{L}} \right).$$

As a result one can use N bootstrap samples from the observation of T^L_{1-D}, with $L/\ln L = O(\delta_n(N)^{-2})$, in order to get a risk bound of the same order as $\delta_n(N)$.

5.3.2 Proof of Theorem 13

Fix some $n \in \{1, \ldots, r+1\}$ and let α_L be the random variable defined by:

$$\alpha_L \stackrel{def}{=} \sup_{z \in \mathcal{A}^n} \left| \frac{\hat{\mathbb{P}}_L(z)}{\mathbb{P}(z)} - 1 \right|.$$

This implies that when $\alpha_L < 1$ the following inequalities hold:

$$\forall z \in \mathcal{A}^n \qquad \frac{\mathbb{P}(z)}{\hat{\mathbb{P}}_L(z)} \leq \frac{1}{1-\alpha_L},$$

and

$$\begin{aligned} \forall (x,y) \in \mathcal{A}^{n-1} \times \mathcal{A}, \qquad \frac{\hat{\mathbb{P}}_L(y \,|\, x)}{\mathbb{P}(y \,|\, x)} &= \frac{\hat{\mathbb{P}}_L(xy)}{\mathbb{P}(xy)} \times \frac{\sum_{x' \in \mathcal{A}^{n-1}} \mathbb{P}(x'y)}{\sum_{x' \in \mathcal{A}^{n-1}} \hat{\mathbb{P}}_L(x'y)} \\ &\leq \frac{1+\alpha_L}{1-\alpha_L}. \end{aligned}$$

Using this inequalities we obtain the following chain of inequalities valid for any $\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{n-1}$ and $\theta \in \Theta(\overleftarrow{\mathcal{B}})$:

$$
\begin{aligned}
&\mathbb{E}_{\hat{\mathbb{P}}_L^{\otimes N}(dZ_1^N)} r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}) \\
&\quad = \mathbb{E}_{\hat{\mathbb{P}}_L^{\otimes N}(dZ_1^N)} \mathbb{E}_{\mathbb{P}(dZ)} \log \frac{\mathbb{P}(Y \mid X)}{\overrightarrow{\mathbb{Q}}^{(n-1)}(Y \mid X)} \\
&\quad = \mathbb{E}_{\hat{\mathbb{P}}_L^{\otimes N}(dZ_1^N)} \mathbb{E}_{\mathbb{P}(dZ)} \left(\log \frac{\mathbb{P}(Y \mid X)}{\overrightarrow{\mathbb{Q}}^{(n-1)}(Y \mid X)} + \frac{\overrightarrow{\mathbb{Q}}^{(n-1)}(Y \mid X)}{\mathbb{P}(Y \mid X)} - 1 \right) \\
&\quad \leq \frac{1}{1-\alpha_L} \mathbb{E}_{\hat{\mathbb{P}}_L^{\otimes N}(dZ_1^N)} \mathbb{E}_{\hat{\mathbb{P}}_L(dZ)} \left(\log \frac{\mathbb{P}(Y \mid X)}{\overrightarrow{\mathbb{Q}}^{(n-1)}(Y \mid X)} + \frac{\overrightarrow{\mathbb{Q}}^{(n-1)}(Y \mid X)}{\mathbb{P}(Y \mid X)} - 1 \right) \\
&\quad \leq \frac{1}{1-\alpha_L} \left(\mathbb{E}_{\hat{\mathbb{P}}_L(dZ)} \log \frac{\mathbb{P}(Y \mid X)}{\overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}(Y \mid X)} + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right) \\
&\qquad + \frac{1}{1-\alpha_L} \mathbb{E}_{\hat{\mathbb{P}}_L^{\otimes N}(dZ_1^N)} \left(\mathbb{E}_{\hat{\mathbb{P}}_L(dX)\overrightarrow{\mathbb{Q}}^{(n-1)}(dY \mid X)} \frac{\hat{\mathbb{P}}_L(Y \mid X)}{\mathbb{P}(Y \mid X)} - 1 \right) \\
&\quad \leq \frac{1}{1-\alpha_L} \left(\mathbb{E}_{\hat{\mathbb{P}}_L(dZ)} \log \frac{\mathbb{P}(Y \mid X)}{\overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}(Y \mid X)} + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right) + \frac{2\alpha_L}{(1-\alpha_L)^2}.
\end{aligned}
$$

We used the fact that the integrand is non-negative in the third line to derive the fourth line, and we applied Theorem 12 to $\hat{\mathbb{P}}_L$ instead of $\mathbb{P}$ to obtain the fifth line.

For any $L \in \mathbb{N}$ and $\epsilon \in [0, 1)$ this shows that:

$$
\begin{aligned}
&\mathbb{E}_{\hat{\mathbb{P}}_L^{\otimes N}(dZ_1^N)} \left[\mathbf{1}_{(\alpha_L \leq \epsilon)} r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}) \right] \\
&\qquad \leq \frac{2\epsilon}{(1-\epsilon)^2} + \frac{1}{1-\epsilon} \left(\mathbb{E}_{\hat{\mathbb{P}}_L(dZ)} \log \frac{\mathbb{P}(Y \mid X)}{\overrightarrow{\mathbb{Q}}_{\overleftarrow{\mathcal{B}},\theta}(Y \mid X)} + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right).
\end{aligned}
$$

We can now take the expectation with respect to $\mathbb{P}(T_{1-D}^L)$ on both sides of this inequality and observe that the following equality holds for any measurable function $f : \mathcal{A}^n \to \mathbb{R}$:

$$
\mathbb{E}_{\mathbb{P}(dT_{1-D}^L)} \mathbb{E}_{\hat{\mathbb{P}}_L(dZ)} f(Z) = \mathbb{E}_{\mathbb{P}(dZ)} f(Z),
$$

in order to obtain the following:

$$
\mathbb{E} \left[\mathbf{1}_{(\alpha_L < \epsilon)} r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}) \right]
$$

$$\leq \frac{2\epsilon}{(1-\epsilon)^2} + \frac{1}{1-\epsilon} \min_{\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{n-1}} \inf_{\theta \in \Theta(\overleftarrow{\mathcal{B}})} \left(r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}) + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right), \quad (5.1)$$

where the first expectation should be understood as:

$$\mathbb{E}_{\mathbb{P}(dT_{1-D}^L)} \mathbb{E}_{\hat{\mathbb{P}}_L^{\otimes N}(dZ_1^N)}.$$

Let us now upper bound the probability of the event $\{\alpha_L > \epsilon\}$, using the fact that $\mathbb{P}$ is assumed to define a Markov chain whose transitions depends on contexts of length smaller or equal to a maximal length D.

In the case $D < n$ the vector-valued process $\left(Y_i = X_{i-n+1}^i\right)_{i\in\mathbb{Z}}$ is itself a stationary aperiodic irreductible Markov chain on the space $\{z \in \mathcal{A}^n : \mathbb{P}(z) > 0\}$. Therefore we can apply Lemma 13 to obtain that there exist constants $c_1 > 0$, $c_2 > 0$ and $C_1 > 0$ such that for any $L \in \mathbb{N}^*$ and $\epsilon \in (c_1/L, c_2)$ the following holds:

$$\mathbb{P}(\alpha_L > \epsilon) \leq 2|\mathcal{A}|^n e^{-C_1 L \epsilon^2}.$$

In the case $D \geq n$ the process $(Y_i = X_{i-D}^i)_{i\in\mathbb{Z}}$ is a stationary irreductible Markov chain on the space $\{z \in \mathcal{A}^{D+1} : \mathbb{P}(z) > 0\}$. Let μ and $\hat{\mu}_L$ be its stationary and empirical distribution from a sample Y_1^L. An application of Lemma 13 tells us that there exist constants c_3, c_4 and C_2 such that for any $\epsilon \in (c_3/L, c_4)$:

$$\sup_{y\in\mathcal{A}^{D+1}} \left| \frac{\hat{\mu}_L(y)}{\mu(y)} - 1 \right| \leq 2|\mathcal{A}|^{D+1} e^{-C_2 L\epsilon^2}.$$

For any $z \in \mathcal{A}^n$,

$$\begin{cases} \mathbb{P}(z) = \displaystyle\sum_{y\in\mathcal{A}^{D+1}, z\prec y} \mu(z), \\ \hat{\mathbb{P}}_L(z) = \displaystyle\sum_{y\in\mathcal{A}^{D+1}, z\prec y} \hat{\mu}_L(z), \end{cases}$$

and therefore:

$$\min_{y\in\mathcal{A}^{D+1}} \frac{\hat{\mu}_L(y)}{\mu(y)} \leq \frac{\hat{\mathbb{P}}_L(z)}{\mathbb{P}(z)} \leq \max_{y\in\mathcal{A}^{D+1}} \frac{\hat{\mu}_L(y)}{\mu(y)}.$$

As a result, for any $\epsilon \in (c_3/L, c_4)$,

$$\sup_{z\in\mathcal{A}^n} \left| \frac{\hat{\mathbb{P}}_L(z)}{\mathbb{P}(z)} - 1 \right| \leq 2|\mathcal{A}|^{D+1} e^{-C_2 L\epsilon^2}.$$

The upper bound obtained in the cases $D < n$ and $D \geq n$ can be summarized as follows: there exist constants c_5 , c_6 and C_3 such that for any $\epsilon \in (c_5/L, c_6)$,

$$\mathbb{P}(\alpha_L > \epsilon) \leq 2|\mathcal{A}|^{\max(n,D+1)} e^{-C_3 L \epsilon^2}. \tag{5.2}$$

At this point we can observe that the estimator $\overset{\rightarrow}{\mathbb{Q}}^{(n-1)}$ defined in the previous section satisfies:

$$\forall (x,y) \in \mathcal{A}^{n-1} \times \mathcal{A}, \qquad \overset{\rightarrow}{\mathbb{Q}}^{(n-1)}(y \,|\, x) \geq \frac{1}{N + |\mathcal{A}|},$$

and therefore:

$$r_n(\mathbb{P}, \overset{\rightarrow}{\mathbb{Q}}^{(n-1)}) \leq \log(N + |\mathcal{A}|). \tag{5.3}$$

Theorem 13 is a consequence of (5.1), (5.2) and (5.3). □

5.4 An other bootstrap scheme to estimate a context tree Markov chain

In this section we present a different bootstrap scheme to draw the sample used by the adaptive context tree. We propose to replace the iterative procedure studied in sections 5.2 and 5.3 by a single iteration whose goal is to estimate a forward probability $\overset{\rightarrow}{\mathbb{Q}}(dT_1^\infty)$, without any prior knowledge of an estimated backward distribution.

We suppose that $(T_n)_{n\in\mathbb{Z}}$ is a stationary Markov process whose transition matrix is of an unknown context tree type, i.e. that there exists a complete suffix dictionary $\overset{\leftarrow}{\mathcal{D}} \subset \mathcal{A}^*$ such that:

$$\forall t_{-\infty}^0 \in \mathcal{A}^{\mathbb{Z}_-}, \qquad \mathbb{P}(t_0 \,|\, t_{-\infty}^{-1}) = \mathbb{P}(t_0 \,|\, \overset{\leftarrow}{\mathcal{D}}(t_{-\infty}^{-1})).$$

Moreover we suppose that there exists a constant $\alpha > 0$ such that:

$$\forall s \in \overset{\leftarrow}{\mathcal{D}}, \qquad \mathbb{P}(s) > \alpha.$$

Finally we suppose that we know an upper bound $D \geq \max_{s \in \overset{\leftarrow}{\mathcal{D}}} l(s)$ and a lower bound $2^{-d} < \alpha$.

5.4.1 Description of the algorithm

Estimation of a maximal context tree

A realization of the process is observed between times $2-D$ and M, where M is a fixed "large" integer. From this observation the empirical probabilities of all strings of lengths not larger than D are computed as:

$$\forall s \in \bigcup_{i=0}^{D} \mathcal{A}^i, \qquad \hat{\mathbb{P}}_M(s) = \frac{\sum_{n=1}^{M} \delta_s(T_{n+1-l(s)}^{n})}{M}.$$

Then the following complete suffix dictionaries are computed for $k \in \{0, \dots, D\}$ (the proof of Lemma 10 can be easily adapted to show that these sets are indeed complete suffix dictionaries):

$$\overleftarrow{\mathcal{D}}_k \overset{def}{=} \max \left\{ s \in \bigcup_{i=1}^{k} \mathcal{A}^i \ : \forall 2 \le r \le l(s),\ \min_{y \in \mathcal{A}} \hat{\mathbb{P}}_M(y s_r^{l(s)}) \ge 2^{-d} \right\}.$$

In other words $\overleftarrow{\mathcal{D}}_k$ is the largest complete suffix dictionary made of strings of length at most k whose empirical probability is larger than 2^{-d}.

For any $k \in \{0, \dots, D\}$, $|\overleftarrow{\mathcal{D}}_k| \le 2^{-d}$ so it is possible to code $\overleftarrow{\mathcal{D}}_k$ with d bits. Let us denote $\chi_k : \overleftarrow{\mathcal{D}}_k \to \{0,1\}^d$ the corresponding encoder and $\overleftarrow{\mathcal{B}}_k = \chi_k(\overleftarrow{\mathcal{D}}_k)$ the corresponding set of binary strings which we impose to form a complete suffix tree.

The following lemma shows that with high probability the unknown context tree $\overleftarrow{\mathcal{D}}$ is a subtree of $\overleftarrow{\mathcal{D}}_D$:

Lemma 11 *There exist two constants $C > 0$ and $M_0 > 0$ such that for any $M > M_0$:*

$$\mathbb{P}\left(\overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D\right) \ge 1 - D|\overleftarrow{\mathcal{D}}|e^{-CM}.$$

I.i.d. sampling

The goal of the bootstrap scheme is to get i.i.d. samples with a distribution as close as possible to $\mathbb{P}$. $\hat{\mathbb{P}}_M$ being an approximation of $\mathbb{P}$ on the suffix trees $\overleftarrow{\mathcal{D}}_D$ a natural sampling procedure on $\overleftarrow{\mathcal{D}}_D$ is to sample M new strings i.i.d. from the distribution $\hat{\mathbb{P}}_M$. Let therefore $(s_i)_{i=1}^M \in \left(\overleftarrow{\mathcal{D}}_D\right)^M$ be such an i.i.d. sample, and let d_s be the number of times the string s has been sampled:

$$\forall s \in \overleftarrow{\mathcal{D}}_D, \qquad d_s = \sum_{i=1}^{M} \delta_s(s_i).$$

One can observe that this sampling is necessary to obtain an random i.i.d design, which would not be the case if the sample was deterministically set to be the M contexts which appeared in T_{2-D}^{M}. In order to obtain an i.i.d. sample to estimate the conditional law of a character knowing a particular context it is classical to observe a new realization of the process and select the characters following every appearance of the given context. More precisely, define the counters:

$$\forall (s,K) \in \overleftarrow{\mathcal{D}}_D \times \mathbb{N}, \qquad c_s(K) = \sum_{n=M+1}^{M+K} \delta_s(T_{n-\ell(s)}^{n-1}),$$

which count the number of times the context s appears in the string $T_{M+1-l(s)}^{M+K}$, and define the regression sample $(s_i, Y_i)_{i=1}^{M}$, where

$$(Y_i : s_i = s) = (T_{M+n} : T_{n-\ell(s)}^{n-1} = s \text{ and } c_s(n) \leq d_s).$$

This regression sample is therefore built from $(X_{N+1}, \dots, X_{N+\tau})$ where the stopping time τ is defined by

$$\tau = \inf\{n : \min_{s \in \overleftarrow{\mathcal{D}}} (c_s(n) - d_s) \geq 0\}.$$

Under the previous assumption, the following holds:

Lemma 12 *If $\overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D$ then the samples $(s_i, Y_i)_{i\in\{1,\dots,N\}}$ are i.i.d. with the common distribution:*

$$\hat{\mathbb{P}}_M(ds) \otimes \mathbb{P}(dT_1 \,|\, \overleftarrow{\mathcal{D}}_D(T_{-\infty}^{0}) = s)$$

Estimation

For any $k \in \{0, \dots, D\}$ one can now form the estimator $\overrightarrow{\mathbb{Q}}^{(k)}(T_1 \,|\, T_{1-k}^{0})$ using the regression sample $(s_i, Y_i)_{i\in\{1,\dots,N\}}$ to aggregate $\left\{\overrightarrow{\mathbb{Q}}^{(k)}_{\overleftarrow{\mathcal{B}},\theta}(Y \,|\, \overleftarrow{X}) \,:\, \overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_k,\ \theta \in \Theta(\overleftarrow{\mathcal{B}})\right\}$, as in Sections 5.2 and 5.3.

5.4.2 Performance of the forward estimation

As for the estimators presented in Sections 5.2 and 5.3 the performance of this estimator is measured in terms of conditional Kullback-Leibler divergence for which the following upper bound holds:

Theorem 14 *There exists four positive constants c_1, c_2, C_1, C_2 such that for any $\epsilon \in [c_1/M, c_2]$:*

$$\mathbb{E}r_n(\mathbb{P},\overrightarrow{\mathbb{Q}}) \leq (1+\epsilon) \min_{\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{n-1}} \inf_{\theta\in\Theta_{\overleftarrow{\mathcal{B}}}} \left(r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}) + \frac{C_N|\overleftarrow{\mathcal{B}}|}{N} \right. \\ \left. + \log(M+|\mathcal{A}|)\left(2^D e^{-C_1 M\epsilon^2} + D|\overleftarrow{\mathcal{D}}|e^{-C_2 M}\right)\right),$$

where C_N is defined in Theorem 12.

5.4.3 Proofs

Proof of Lemma 11

First observe that :

$$1-\mathbb{P}\left(\overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D\right) = \mathbb{P}\left(\inf_{s\in\overleftarrow{\mathcal{D}}} \hat{\mathbb{P}}_M(s) < 2^{-d}\right) \\ \leq \mathbb{P}\left(\inf_{s\in\overleftarrow{\mathcal{D}}} \frac{\hat{\mathbb{P}}_M(s)}{\mathbb{P}(s)} < 1 - \frac{\alpha - 2^{-d}}{\alpha}\right).$$

Let now c_1, c_2 and C the constants given by Corollary 2. Let $\epsilon = \min((\alpha - 2^{-d})/\alpha, c_2)$ and M_0 chosen such that $\epsilon > c_1/M_0$. Then from corollary we obtain that:

$$1-\mathbb{P}\left(\overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D\right) \leq \mathbb{P}\left(\inf_{s\in\overleftarrow{\mathcal{D}}} \frac{\hat{\mathbb{P}}_M(s)}{\mathbb{P}(s)} < 1-\epsilon\right) \\ \leq D|\overleftarrow{\mathcal{D}}|e^{-CM\epsilon^2}.$$

Lemma 11 follows by taking the constant equal to $C\epsilon^2$. □

Proof of Lemma 12

By definition the $(s_i)_{i\in\{1,\dots,N\}}$ are i.i.d. according to $\hat{\mathbb{P}}_M(ds)$. Conditionally to $(s_1,\dots,s_N)$, there exists a stopping time τ_i such that $Y_i = T_{\tau_i+1}$. This stopping time can be defined using $e(i) = \sum_{k=1}^{i-1} \delta_{s_i}(s_k)$ by:

$$\tau_i = \inf_{k>M}\left\{\exists M < k_1 < \ldots < k_{e(i)} < k \,:\, T^{k_1}_{k_1-l(s_i)+1} = \ldots = T^{k_{e(i)}}_{k_{e(i)}-l(s_i)+1} = T^k_{k-l(s_i)+1} = s_i\right\}.$$

By the strong Markov property it is therefore true that conditionally to $\{\overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D\}$:

$$\mathbb{P}(Y_i = y \mid s_1, \dots, s_N, Y_1, \dots, Y_{i-1}) = \mathbb{P}(T_{\tau_i+1} = y \mid s_1, \dots, s_N, T_{\tau_1+1}, \dots, T_{\tau_{i-1}+1})$$
$$= \mathbb{P}(T_1 = y | \overleftarrow{\mathcal{D}}_D(T_{-\infty}^0) = s_i). \square$$

Proof of Theorem 14

First observe that on the event $\{\overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D\}$ the following holds:

$$\forall n \in \mathbb{N}, \forall t_1^n \in \mathcal{A}^n, \qquad \mathbb{P}(t_n \mid t_1^{n-1}) = \mathbb{P}(t_n \mid \overleftarrow{\mathcal{D}}_{n-1}(t_1^{n-1})).$$

For any $n \in \mathbb{N}$ let us use the notations $\mathbb{P}^{(n)}$ and $\hat{\mathbb{P}}_M^{(n)}$ to denote the measures on $\overleftarrow{\mathcal{D}}_n$ induced by $\mathbb{P}$ and $\hat{\mathbb{P}}_M$. Then the following holds conditionally to T_{1-D}^M for any $n \in \mathbb{N}^*$, $\overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{n-1}$:

$$\begin{aligned}
&\mathbb{E}_{\bigotimes_{i=1}^N (\hat{\mathbb{P}}_M(ds_i) \otimes \mathbb{P}(dY_i \mid s_i))} \left[r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}) \mathbf{1}(\overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D) \right] \\
&\leq \mathbb{E}_{\bigotimes_{i=1}^N \left(\hat{\mathbb{P}}_M^{(n-1)}(ds_i) \otimes \mathbb{P}(dY_i \mid s_i)\right)} \left[\sum_{s \in \overleftarrow{\mathcal{D}}_{n-1}} \mathbb{P}^{(n-1)}(s) \sum_{y \in \mathcal{A}} \mathbb{P}(y \mid s) \log \frac{\mathbb{P}(y \mid s)}{\overrightarrow{\mathbb{Q}}^{(n-1)}(y \mid s)} \right] \\
&\leq \sup_{s \in \overleftarrow{\mathcal{D}}_{n-1}} \frac{\mathbb{P}^{(n-1)}(s)}{\hat{\mathbb{P}}_M^{(n-1)}(s)} \\
&\qquad \times \mathbb{E}_{\bigotimes_{i=1}^N \left(\hat{\mathbb{P}}_M^{(n-1)}(ds_i) \otimes \mathbb{P}(dY_i \mid s_i)\right)} \left[\sum_{s \in \overleftarrow{\mathcal{D}}_{n-1}} \hat{\mathbb{P}}_M^{(n-1)}(s) \sum_{y \in \mathcal{A}} \mathbb{P}(y \mid s) \log \frac{\mathbb{P}(y \mid s)}{\overrightarrow{\mathbb{Q}}^{(n-1)}(y \mid s)} \right] \\
&\leq \sup_{s \in \overleftarrow{\mathcal{D}}_{n-1}} \frac{\mathbb{P}^{(n-1)}(s)}{\hat{\mathbb{P}}_M^{(n-1)}(s)} \times \left[\sum_{s \in \overleftarrow{\mathcal{D}}_{n-1}} \hat{\mathbb{P}}_M^{(n-1)}(s) \sum_{y \in \mathcal{A}} \mathbb{P}(y \mid s) \log \frac{\mathbb{P}(y \mid s)}{\overrightarrow{\mathbb{Q}}_{\overleftarrow{\mathcal{B}},\theta}^{(n-1)}(y \mid s)} + \frac{C_N |\overleftarrow{\mathcal{B}}|}{N} \right]
\end{aligned} \tag{5.4}$$

where we used (5.7) and the fact that $\overrightarrow{\mathbb{Q}}^{(n-1)}(y \mid s)$ is the adaptive context tree estimator to aggregate the conditional distributions $\left\{\overrightarrow{\mathbb{Q}}_{\overleftarrow{\mathcal{B}},\theta}^{(n-1)}(Y \mid \overleftarrow{X}) \ : \ \overleftarrow{\mathcal{B}} \prec \overleftarrow{\mathcal{B}}_{n-1}, \ \theta \in \Theta(\overleftarrow{\mathcal{B}})\right\}$ using an i.i.d. sample with distribution $\hat{\mathbb{P}}_M^{(n-1)}(s) \otimes \mathbb{P}(y \mid s)$.

For any $\epsilon > 0$ let now Ω_ϵ be the following event:

$$\Omega_\epsilon = \left\{ \sup_{s \in \overleftarrow{\mathcal{D}}_D} \frac{\mathbb{P}(s)}{\hat{\mathbb{P}}_M(s)} \leq 1 + \epsilon \right\} \bigcap \left\{ \overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D \right\}.$$

Note that by Lemma 11 and Lemma 13 there exist positive constants M_0, c_1, c_2, C_1 and C_2 such that for any $M > M_0$ and $\epsilon \in [c_1/M, c_2]$ the probability of Ω_ϵ is lower bounded by :

$$
\begin{aligned}
\mathbb{P}(\Omega_\epsilon) &= \mathbb{P}\left(\overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D\right)\mathbb{P}\left(\sup_{s\in\overleftarrow{\mathcal{D}}_D}\frac{\mathbb{P}(s)}{\hat{\mathbb{P}}_M(s)} \leq 1+\epsilon \,|\, \overleftarrow{\mathcal{D}} \prec \overleftarrow{\mathcal{D}}_D\right) \\
&\geq \left(1 - D|\overleftarrow{\mathcal{D}}|e^{-C_1 M}\right)\left(1-2^D e^{-C_2 M\epsilon^2}\right) \\
&\geq 1 - D|\overleftarrow{\mathcal{D}}|e^{-C_1 M} - 2^D e^{-C_2 M\epsilon^2}.
\end{aligned}
\tag{5.5}
$$

From (5.4) we get that for any $\epsilon > 0$:

$$
\begin{aligned}
&\mathbb{E}_{\bigotimes_{i=1}^N (\hat{\mathbb{P}}_M(ds_i)\otimes \mathbb{P}(dY_i \,|\, s_i))}\left[r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}})\mathbf{1}(\Omega_\epsilon)\right] \\
&\leq (1+\epsilon)\left[\sum_{s\in\overleftarrow{\mathcal{D}}_{n-1}} \hat{\mathbb{P}}_M^{(n-1)}(s)\sum_{y\in\mathcal{A}}\mathbb{P}(y\,|\,s)\log\frac{\mathbb{P}(y\,|\,s)}{\overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}(y\,|\,s)} + \frac{C_N|\overleftarrow{\mathcal{B}}|}{N}\right].
\end{aligned}
$$

We can now take the expectation on both sides with respect to $\mathbb{P}(dT_{1-D}^M)$ and observe that for all $s \in \overleftarrow{\mathcal{D}}_n$, $\mathbb{E}\hat{\mathbb{P}}_M^{(n)}(s) = \mathbb{P}(s)$ in order to get:

$$
\mathbb{E}\left[r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}})\mathbf{1}(\Omega_\epsilon)\right] \leq (1+\epsilon)\left[r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}}^{(n-1)}_{\overleftarrow{\mathcal{B}},\theta}) + \frac{C_N|\overleftarrow{\mathcal{B}}|}{N}\right].
$$

On the other side $r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}})$ is upper bounded by $\log(M + |\mathcal{A}|)$ by definition of $\overrightarrow{\mathbb{Q}}$ and therefore, by (5.5):

$$
\mathbb{E}\left[r_n(\mathbb{P}, \overrightarrow{\mathbb{Q}})\mathbf{1}(\Omega_\epsilon^c)\right] \leq \log(M+|\mathcal{A}|)\left(D|\overleftarrow{\mathcal{D}}|e^{-C_1 M} - 2^D e^{-C_2 M\epsilon^2}\right).
$$

Summing up these two inequalities ends the proof of Theorem 14. □

5.5 Annex : the adaptive context tree

We recall in this annex the definition and the main properties of the adaptive context tree algorithm as defined in [87], and we specialize to the case of binary contexts.

5.5.1 Definition and performance

Let (X, Y) be a r.v. on the space $\left(\{0,1\}^D \times \mathcal{A}\right)$ with probability distribution P. For any conditional probability distribution $Q(y|x)$ (which satisfies $\forall x \in \{0,1\}^D, \sum_{y\in\mathcal{A}} Q(y|x) = 1$) let $d(P, Q)$ denote the conditional Kullback-Leibler entropy between P and Q:

$$d(P,Q) \stackrel{def}{=} \sum_{x\in\{0,1\}^D} \sum_{y\in\mathcal{A}} P(x,y) \log \frac{P(y|x)}{Q(y|x)}.$$

Let now $\overleftarrow{\mathcal{D}}$ be a fixed complete suffix tree included in $\{0,1\}^D$, and let $\mathcal{S}(\mathcal{D})$ be the set of complete suffix subtrees of $\mathcal{D}$. A natural probability distribution on $\mathcal{S}(\mathcal{D})$ is the distribution of the genealogic trees of a Galton-Watson process where each node has two children with probability 1/2 and no child with probability 1/2, except when it is a node of $\overleftarrow{\mathcal{D}}$ in which case it a.s. has no child. If we call π this probability distribution then we have:

$$\begin{aligned} \forall \mathcal{D} \in \mathcal{S}(\mathcal{D}), \qquad \pi(\mathcal{D}) &= \left(\frac{1}{2}\right)^{(\#\text{ nodes})} \left(\frac{1}{2}\right)^{(\#\text{ leaves}) - |\mathcal{D}\cap\overleftarrow{\mathcal{D}}|} \\ &= \frac{1}{4^{|\mathcal{D}|}} \times 2^{|\mathcal{D}\cap\overleftarrow{\mathcal{D}}|+1} \\ &\geq \frac{1}{4^{|\mathcal{D}|}}. \end{aligned} \tag{5.6}$$

Let now $(X_i,Y_i)_{i\in\{1,\dots,N\}}$ be i.i.d. drawn according to P. The adaptive context tree estimator is defined by separating the observation into two sets.

- Use $(X_i,Y_i)_{i\in\{1,\dots,K\}}$ to build the estimators for each $\mathcal{D} \prec \overleftarrow{\mathcal{D}}$ as:

$$\hat{Q}_{\mathcal{D}}(y\,|\,x) = \frac{\sum_{i=1}^{K} \delta_{\mathcal{D}(x)}(\mathcal{D}(X_i))\delta_y(Y_1) + 1}{\sum_{i=1}^{K} \delta_{\mathcal{D}(x)}(\mathcal{D}(X_i)) + |\mathcal{A}|}.$$

- Use $(X_i,Y_i)_{i\in\{K+1,\dots,N\}}$ to aggregate those estimator by using a pseudo-Bayesian mixture (the Gibbs estimator):

$$\hat{Q}(y|x) = \sum_{\mathcal{D}\prec\overleftarrow{\mathcal{D}}} \rho(\mathcal{D})\hat{Q}_{\mathcal{D}}(y\,|\,x),$$

with

$$\rho(\mathcal{D}) = \frac{1}{Z}\pi(\mathcal{D}) \prod_{i=K+1}^{N} \hat{Q}_{\mathcal{D}}(Y_i\,|\,X_i)^{\beta},$$

where Z is a normalizing constant which ensures that ρ is a probability distribution and β is a parameter (the inverse temperature of the Gibbs estimator) smaller than 1/2.

Let

$$\chi_N = \log(N + |\mathcal{A}|),$$

and

$$\begin{aligned}\beta_N &= \frac{1}{\chi_N - 1}\left(\sqrt{1-(\chi_N-1)\left(2-\frac{\log\chi_N}{\chi_N}\right)\frac{\log\chi_N}{\chi_N}}-1\right)\\ &\underset{N\to+\infty}{\sim} \frac{\sqrt{2\log\log N}}{\log N}.\end{aligned}$$

Then it can be shown that if the inverse temperature of the Gibbs estimator β is smaller than β_N then the following holds (see [22] and [87]):

$$\mathbb{E}d(P,\hat{Q}) \leq \min_{\mathcal{D}\prec\overleftarrow{\mathcal{D}}} \inf_{\theta\in\Theta_{\mathcal{D}}} \left\{ d(P,Q_{\mathcal{D},\theta}) + \frac{1}{\beta(N-K+1)}\log\frac{1}{\pi(\mathcal{D})} + \frac{|\mathcal{A}|-1}{K+1}|\mathcal{D}| \right\}.$$

Using (5.6) and choosing:

$$K = \frac{\sqrt{|\mathcal{A}-1|}(N+1) - \sqrt{2\beta^{-1}}}{\sqrt{|\mathcal{A}-1|} + \sqrt{2\beta^{-1}}},$$

rounded to the nearest larger (resp. smaller) integer if $2\beta^{-1}$ is larger (resp. smaller) than $|\mathcal{A}| - 1$, one finally gets the following upper bound:

$$\mathbb{E}d(P,\hat{Q}) \leq \min_{\mathcal{D}\prec\overleftarrow{\mathcal{D}}} \inf_{\theta\in\Theta_{\mathcal{D}}} \left\{ d(P,Q_{\mathcal{D},\theta}) + |\mathcal{D}|\frac{\left(\sqrt{2\beta^{-1}}+\sqrt{|\mathcal{A}|-1}\right)^2}{N+2}\left(1+\frac{1}{N}\right) \right\}. \tag{5.7}$$

5.5.2 Implementation

This formulation leads to an efficient algorithm thanks to a factorization method which we now describe. Let $\mathcal{C} \subset \{0,1\}^D$ be the set of suffixes of the elements of $\overleftarrow{\mathcal{D}}$. $\mathcal{C}$ is a tree to any node or leaf $s \in \mathcal{C}$ of which several counters should be attached and incremented according to the observation $(X_i, Y_i)_{i\in\{1,\dots,N\}}$:

$$\begin{cases} \forall y \in \mathcal{A}, \qquad a_s(y) = \displaystyle\sum_{i=1}^{K} \mathbf{1}(s \prec x \text{ and } Y_i = y), \\ \forall y \in \mathcal{A}, \qquad b_s(y) = \displaystyle\sum_{i=K+1}^{N} \mathbf{1}(s \prec x \text{ and } Y_i = y), \\ \qquad\qquad\qquad n_s = \displaystyle\sum_{y\in\mathcal{A}} a_s(y). \end{cases}$$

Once these counters have been updated with all observations one can compute the following variables at each node $s \in \mathcal{C}$:

$$\begin{cases} \forall y \in \mathcal{A}, & \theta_s(y) = \dfrac{a_s(y)+1}{n_s+|\mathcal{A}|}, \\ & w(s) = \displaystyle\prod_{y \in \mathcal{A}} \theta_s(y)^{\beta b_s(y)}. \end{cases}$$

With these notations one can observe that

$$\forall \mathcal{D} \prec \overleftarrow{\mathcal{D}}, \qquad \rho(\mathcal{D}) = \frac{1}{Z}\pi(\mathcal{D}) \prod_{s \in \mathcal{D}} w(s),$$

with

$$Z = \sum_{\mathcal{D} \prec \overleftarrow{\mathcal{D}}} \pi(\mathcal{D}) \prod_{s \in \mathcal{D}} w(s).$$

Z can be computed as $Z = \gamma(\lambda)$ where γ is defined recursively on any $s \in \mathcal{C}$ by:

$$\begin{cases} \gamma(s) = w(s) & \text{if } s \in \overleftarrow{\mathcal{D}}, \\ \gamma(s) = \dfrac{w(s) + \gamma(0s)\gamma(1s)}{2} & \text{otherwise.} \end{cases}$$

In order to compute $\hat{Q}(y|x)$ for a given (x, y) one can observe that:

$$\hat{Q}(y \mid x) = \sum_{\mathcal{D} \prec \overleftarrow{\mathcal{D}}} \rho(\mathcal{D}) \hat{Q}_{\mathcal{D}}(y|x),$$

and that:

$$\forall \mathcal{S} \prec \overleftarrow{\mathcal{D}}, \qquad \hat{Q}_{\mathcal{D}}(y \mid x) = \theta_{\mathcal{D}(x)}(y).$$

As a result, if one defines the variables $\bar{w}(s)$ for $s \in \mathcal{C}$ as:

$$\begin{cases} \bar{w}(s) = w(s)\theta_s(y) & \text{if } s \prec x, \\ \bar{w}(s) = w(s) & \text{otherwise ,} \end{cases}$$

then the following holds:

$$\begin{aligned} \hat{Q}(y \mid x) &= \frac{1}{Z} \sum_{\mathcal{D} \prec \overleftarrow{\mathcal{D}}} \pi(\mathcal{D}) \prod_{s \in \mathcal{D}} \bar{w}(s) \\ &= \frac{\bar{\gamma}(\lambda)}{\gamma(\lambda)}, \end{aligned}$$

where $\bar{\gamma}$ is recursively defines on the nodes of $\mathcal{C}$ exactly the same way as γ is, with w being replaced by $\bar{w}$.

5.6 Annex : concentration of the empirical measure of a Markov chain

In this section we prove an elementary deviation inequality for the empirical measure of a mixing Markov chain using the renewal approach. For sake of completeness we give a full proof of a precise statement which is used in Sections 5.2 and 5.3, even though the result as well as the method of proof is very classical in the literature about large deviations of discrete Markov chains.

5.6.1 Main result

Let $(X_n)_{n\geq 1}$ be a stationary irreductible aperiodic homogeneous Markov chain on a finite state space $\mathcal{A}$, with stationary distribution μ. The empirical probability of any state $s \in \mathcal{A}$ is defined for any $N \in \mathbb{N}^*$ as:

$$\hat{\mu}_N(s) = \frac{1}{N}\sum_{i=1}^{N} \delta_s\left(X_i\right).$$

The goal of this section is to compute an upper bound for the distribution of the maximum deviation of the empirical probabilities. Let us introduce some notations in terms of which the main result of this section is stated.

For any $s \in \mathcal{A}$ let $(T_k^s)_{k\geq 1}$ be defined as follows:

$$\begin{aligned} T_0^s &= 0, \\ T_k^s &= \inf\{n > T_{k-1}^s : X_n = s\}, \qquad k \geq 1. \end{aligned}$$

Let also

$$\tau_k^s = T_k^s - T_{k-1}^s, \qquad k \geq 1.$$

It is well known that the $(T_k^s)_{k\geq 0}$ are a.s. finite, that the $(\tau_k^s)_{k\geq 1}$ are independent, that the $(\tau_k^s)_{k\geq 2}$ are identically distributed and that :

$$\mathbb{E}\tau_1^s = \mathbb{E}\tau_2^s = \frac{1}{\mu(s)}. \tag{5.8}$$

Moreover it is known that each τ_s^k has exponential moments, i.e. that there exists two positive constants A and γ such that:

$$\forall (i,k,s) \in \mathbb{N}^2 \times \mathcal{A} \qquad \mathbb{P}\left(\tau_i^s \geq k\right) \leq Ae^{-\gamma k}. \tag{5.9}$$

With these notations we can state the following result:

Lemma 13 *For any $N \in \mathbb{N}$ and $\epsilon > \max_{s\in\mathcal{A}}(1/N\mu(s))$ the following upper bounds hold:*

$$\mathbb{P}\left(\sup_{s\in\mathcal{A}} \frac{\hat{\mu}_N(s)}{\mu(s)} > 1+\epsilon\right) \leq \sum_{s\in\mathcal{A}} \exp\left[-\frac{\gamma^2 N\epsilon^2}{4A\mu(s)}(1-2\epsilon)\right],$$

and

$$\mathbb{P}\left(\inf_{s\in\mathcal{A}} \frac{\hat{\mu}_N(s)}{\mu(s)} < 1-\epsilon\right) \leq \sum_{s\in\mathcal{A}} \exp\left[-\frac{\gamma^2 N\epsilon^2}{4A\mu(s)}(1-\epsilon)\left(1-\frac{2}{\epsilon N\mu(s)}\right)\left(1-\frac{\gamma\epsilon}{2A\mu(s)}\right)\right].$$

A straightforward consequence of Lemma 13 is the following result :

Corollary 1 *If $(\epsilon_N)_{N\in\mathbb{N}}$ is a non-negative sequence such that:*

$$\begin{cases} \lim_{N\to+\infty} \epsilon_N = 0, \\ \lim_{N\to+\infty} N\epsilon_N = +\infty, \end{cases}$$

then there exists an increasing sequence $(C_N)_{N\in\mathbb{N}}$ such that:

$$\lim_{N\to+\infty} C_N = \min_{s\in\mathcal{A}} \frac{\gamma^2}{4A\mu(s)},$$

and such that for any $N \in \mathbb{N}$:

$$\mathbb{P}\left(\sup_{s\in\mathcal{S}} \frac{\hat{\mu}_N(s)}{\mu(s)} > 1+\epsilon_N\right) \leq |\mathcal{A}|e^{-C_N N\epsilon_N^2},$$

and

$$\mathbb{P}\left(\inf_{s\in\mathcal{S}} \frac{\hat{\mu}_N(s)}{\mu(s)} < 1-\epsilon_N\right) \leq |\mathcal{A}|e^{-C_N N\epsilon_N^2}.$$

We can also state a concentration result for the empirical probabilities of the suffixes of a context tree. Let $(T_n)_{n\in\mathbb{Z}}$ be a stationary D-th order Markov process whose transition matrix is from a context tree $\overleftarrow{\mathcal{D}}$ type, and such that $\mathbb{P}(t_1 \mid \overleftarrow{\mathcal{D}}(t_{-\infty}^0)) > 0$ for any $t_1 \in \mathcal{A}$ and $\overleftarrow{\mathcal{D}}(t_{-\infty}^0) \in \overleftarrow{\mathcal{D}}$. We recall here the definition of the empirical probability of a string of length not larger than D form the observation of T_{2-D}^N:

$$\forall s \in \bigcup_{i=0}^{D} \mathcal{A}^i, \qquad \hat{\mathbb{P}}_N(s) = \frac{\sum_{n=1}^{N} \delta_s(T_{n+1-l(s)}^n)}{N}.$$

Under these assumptions the following holds (see the proof in section 5.6.4) :

Corollary 2 *There exist three positive constants c_1 , c_2 and C such that for any $N \in \mathbb{N}$ and $\epsilon \in (c_1/N, c_2)$ the following holds:*

$$\mathbb{P}\left(\sup_{s\in\overleftarrow{\mathcal{D}}} \left|\frac{\hat{\mathbb{P}}_N(s)}{\mathbb{P}(s)} - 1\right| > \epsilon\right) \leq 2D|\overleftarrow{\mathcal{D}}|e^{-CN\epsilon^2}.$$

5.6.2 Proof of Lemma 13

Let us first introduce some notations:

$$\begin{aligned}
\forall x \in \mathbb{R}, \qquad & \lceil x \rceil = \inf\{n \in \mathbb{Z} : n > x\}, \\
\forall x \in \mathbb{R}, \qquad & \lceil\lceil x \rceil\rceil = \inf\{n \in \mathbb{Z} : n \geq x\}.
\end{aligned}$$

Note the following inequalities which will be used several times in the sequel and which hold for any $x \in \mathbb{R}$:

$$\begin{aligned}
& x < \lceil x \rceil \leq x + 1, \\
& x \leq \lceil\lceil x \rceil\rceil < x + 1.
\end{aligned}$$

For any $\epsilon > 0$, $N \in \mathbb{N}^*$ and $s \in \mathcal{A}$ the following events are equal:

$$\begin{aligned}
\left\{ \frac{\hat{\mu}_N(s)}{\mu(s)} > 1 + \epsilon \right\} &= \left\{ \sum_{i=1}^{N} \delta_s(X_i) > N\mu(s)(1+\epsilon) \right\} \\
&= \left\{ T^s_{\lceil N\mu(s)(1+\epsilon) \rceil} \leq N \right\} \\
&= \left\{ \sum_{i=1}^{\lceil N\mu(s)(1+\epsilon) \rceil} \tau_i^s \leq N \right\}.
\end{aligned}$$

As a result the following holds for any $(\lambda_s)_{s\in\mathcal{A}} \in (\mathbb{R}^+)^{\mathcal{A}}$:

$$\begin{aligned}
\mathbb{P}\left(\sup_{s\in\mathcal{A}} \frac{\hat{\mu}_N(s)}{\mu(s)} > 1 + \epsilon \right) &= \mathbb{P}\left[\sup_{s\in\mathcal{A}} \left(N - \sum_{i=1}^{\lceil N\mu(s)(1+\epsilon) \rceil} \tau_i^s \right) \geq 0 \right] \\
&= \mathbb{P}\left[\sup_{s\in\mathcal{A}} \exp\left(\lambda_s \left(N - \sum_{i=1}^{\lceil N\mu(s)(1+\epsilon) \rceil} \tau_i^s \right) \right) \geq 1 \right] \\
&\leq \mathbb{E} \sup_{s\in\mathcal{A}} \left[\exp\left(\lambda_s \left(N - \sum_{i=1}^{\lceil N\mu(s)(1+\epsilon) \rceil} \tau_i^s \right) \right) \right] \\
&\leq \sum_{s\in\mathcal{A}} \mathbb{E} \exp\left[\lambda_s \left(N - \sum_{i=1}^{\lceil N\mu(s)(1+\epsilon) \rceil} \tau_i^s \right) \right] \\
&\leq \sum_{s\in\mathcal{A}} \left[e^{\lambda_s N} \prod_{i=1}^{\lceil N\mu(s)(1+\epsilon) \rceil} \mathbb{E} e^{-\lambda_s \tau_i^s} \right].
\end{aligned}$$

In order to upper bound this term we can use the following lemma whose proof is postponed to section 5.6.3:

Lemma 14 *For any* $(i, s) \in \mathbb{N} \times \mathcal{A}$ *and* $\lambda \in (-\infty, \gamma)$ *the following holds:*

$$\mathbb{E}e^{\lambda \tau_i^s} \leq \exp\left[\frac{\lambda}{\mu(s)} + \frac{A\lambda^2}{\gamma(\gamma - \lambda)}\right].$$

This lemma enables us to resume the computation and get the following, after observing that $\gamma + \lambda_s \geq \gamma$ for any $s \in \mathcal{A}$:

$$\mathbb{P}\left(\sup_{s\in\mathcal{A}} \frac{\hat{\mu}_N(s)}{\mu(s)} > 1 + \epsilon\right) \\ \leq \sum_{s\in\mathcal{A}} \exp\left[-\lambda_s\left(\frac{\lceil N\mu(s)(1+\epsilon)\rceil}{\mu(s)} - N\right) + \lambda_s^2 \frac{A\lceil N\mu(s)(1+\epsilon)\rceil}{\gamma^2}\right].$$

Observe that for any $s \in \mathcal{A}$:

$$\begin{aligned}\frac{\lceil N\mu(s)(1+\epsilon)\rceil}{\mu(s)} - N &> \frac{N\mu(s)(1+\epsilon)}{\mu(s)} - N \\ &\geq N\epsilon \\ &\geq 0.\end{aligned}$$

As a result the term in the exponential corresponding to a particular s is minimized for λ_s^* defined by:

$$\lambda_s^* = \frac{\lceil N\mu(s)(1+\epsilon)\rceil\mu(s)^{-1} - N}{2A\gamma^{-2}\lceil N\mu(s)(1+\epsilon)\rceil},$$

and choosing $\lambda_s = \lambda_s^*$ for all $s \in \mathcal{A}$ leads to the following:

$$\begin{aligned}\mathbb{P}\left(\sup_{s\in\mathcal{A}} \frac{\hat{\mu}_N(s)}{\mu(s)} > 1 + \epsilon\right) &\leq \sum_{s\in\mathcal{A}} \exp\left[-\frac{\left(\mu(s)^{-1}\lceil N\mu(s)(1+\epsilon)\rceil - N\right)^2}{4A\gamma^{-2}\lceil N\mu(s)(1+\epsilon)\rceil}\right] \\ &\leq \sum_{s\in\mathcal{A}} \exp\left[-\frac{N^2\epsilon^2}{4A\gamma^{-2}\left(N\mu(s)(1+\epsilon)+1\right)}\right] \\ &\leq \sum_{s\in\mathcal{A}} \exp\left[-\frac{N\epsilon^2\gamma^2}{4A\mu(s)} \times \left(1 - \epsilon - \frac{1}{N\mu(s)}\right)\right] \\ &\leq \sum_{s\in\mathcal{A}} \exp\left[-\frac{N\epsilon^2\gamma^2}{4A\mu(s)} \times (1 - 2\epsilon)\right],\end{aligned}$$

because $\epsilon > (N\mu(s))^{-1}$ by hypothesis. This proves the first inequality stated in Lemma 13.

In order to prove the second inequality we proceed the same way and write:

$$\begin{aligned}
\left\{\frac{\hat{\mu}_N(s)}{\mu(s)} < 1-\epsilon\right\} &= \left\{\sum_{i=1}^{N} \delta_s(X_i) < N\mu(s)(1-\epsilon)\right\} \\
&= \left\{T^s_{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil} > N\right\} \\
&= \left\{\sum_{i=1}^{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil} \tau_i^s > N\right\}.
\end{aligned}$$

As a result the following holds for any $(\lambda_s)_{s\in\mathcal{A}} \in [0,\gamma)^{\mathcal{A}}$:

$$\begin{aligned}
&\mathbb{P}\left(\inf_{s\in\mathcal{A}} \frac{\hat{\mu}_N(s)}{\mu(s)} < 1-\epsilon\right) \\
&\quad = \mathbb{P}\left[\sup_{s\in\mathcal{A}}\left(\sum_{i=1}^{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil} \tau_i^s - N\right) > 0\right] \\
&\quad = \mathbb{P}\left[\sup_{s\in\mathcal{A}} \exp\left(\lambda_s\left(\sum_{i=1}^{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil} \tau_i^s - N\right)\right) > 1\right] \\
&\quad \le \mathbb{E}\sup_{s\in\mathcal{A}}\left[\exp\left(\lambda_s\left(\sum_{i=1}^{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil} \tau_i^s - N\right)\right)\right] \\
&\quad \le \sum_{s\in\mathcal{A}} \mathbb{E}\exp\left[\lambda_s\left(\sum_{i=1}^{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil} \tau_i^s - N\right)\right] \\
&\quad \le \sum_{s\in\mathcal{A}}\left[e^{-\lambda_s N}\prod_{i=1}^{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil} \mathbb{E}e^{\lambda\tau_i^s}\right] \\
&\quad \le \sum_{s\in\mathcal{A}} \exp\left[-\lambda\left(N - \frac{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil}{\mu(s)}\right) + \frac{A\lambda_s^2}{\gamma(\gamma-\lambda_s)}\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil\right].
\end{aligned}$$

By hypothesis $\epsilon > (N\mu(s))^{-1}$ for any $s \in \mathcal{A}$ and therefore :

$$\begin{aligned}
N - \frac{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil}{\mu(s)} &> N\epsilon - \mu(s)^{-1} \\
&> 0.
\end{aligned}$$

The term in the exponential corresponding to the state $s \in \mathcal{A}$ has the form:

$$f(\lambda_s) = -a\lambda_s + b\frac{\lambda_s^2}{\gamma - \lambda_s},$$

with

$$\begin{cases} a &= N - \dfrac{\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil}{\mu(s)} > 0, \\ b &= \dfrac{A}{\gamma}\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil. \end{cases}$$

The function f is differentiable on $[0,\gamma)$ and:

$$f'(\lambda) = -a + b\lambda\frac{2\gamma-\lambda}{(\gamma-\lambda)^2}.$$

As a result a minimum is reached at

$$\lambda_s^* = \gamma\left(1 - \sqrt{\frac{b}{a+b}}\right),$$

and the minimum is equal to:

$$\begin{aligned} f(\lambda_s^*) &= -\gamma\left(\sqrt{b+a} - \sqrt{b}\right)^2 \\ &= -A\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil\left(\sqrt{1 + \frac{\gamma N}{A\lceil\lceil N\mu(s)(1-\epsilon)\rceil\rceil} - \frac{\gamma}{A\mu(s)}} - 1\right)^2 \\ &\leq -AN\mu(s)(1-\epsilon)\left(\sqrt{1 + \frac{\gamma N}{AN\mu(s)(1-\epsilon)+A} - \frac{\gamma}{A\mu(s)}} - 1\right)^2 \\ &\leq -AN\mu(s)(1-\epsilon)\left(\sqrt{1 + \frac{\gamma}{A\mu(s)}\left(\epsilon - \frac{1}{N\mu(s)}\right)} - 1\right)^2 \\ &\leq \frac{N\gamma^2\epsilon^2}{A\mu(s)}\left(1 - \frac{1}{N\mu(s)\epsilon}\right)^2(1-\epsilon)\left(\sqrt{1 + \frac{\gamma}{A\mu(s)}\left(\epsilon - \frac{1}{N\mu(s)}\right)} + 1\right)^{-2} \\ &\leq -\frac{N\gamma^2\epsilon^2}{A\mu(s)}\left(1 - \frac{2}{N\mu(s)\epsilon}\right)(1-\epsilon)\left(\sqrt{1 + \frac{\gamma\epsilon}{A\mu(s)}} + 1\right)^{-2} \\ &\leq -\frac{N\gamma^2\epsilon^2}{4A\mu(s)}\left(1 - \frac{2}{N\mu(s)\epsilon}\right)(1-\epsilon)\left(1 + \frac{\gamma\epsilon}{4A\mu(s)}\right)^{-2} \\ &\leq -\frac{N\gamma^2\epsilon^2}{4A\mu(s)}\left(1 - \frac{2}{N\mu(s)\epsilon}\right)(1-\epsilon)\left(1 - \frac{\gamma\epsilon}{2A\mu(s)}\right). \end{aligned}$$

As a result we obtain the following upper bound:

$$\mathbb{P}\left(\inf_{s\in\mathcal{A}}\frac{\hat{\mu}_N(s)}{\mu(s)} < 1-\epsilon\right) \leq \sum_{s\in\mathcal{A}}\exp\left[-\frac{N\epsilon^2\gamma^2}{4A\mu(s)} \times (1-\epsilon)\left(1 - \frac{2}{N\mu(s)\epsilon}\right)\left(1 - \frac{\gamma\epsilon}{2A\mu(s)}\right)\right],$$

which is the second inequality stated in Lemma 13. □

5.6.3 Proof of Lemma 14

For any $(i,s) \in \mathbb{N} \times \mathcal{A}$ let us use the notation $\tau = \tau_i^s$. For any $\lambda \in (-\infty, \gamma)$ the random variable $\exp(\lambda\tau) - 1 - \lambda\tau$ is non-negative and therefore:

$$\mathbb{E}e^{\lambda\tau} - 1 - \lambda\mathbb{E}\tau = \int_0^\infty \mathbb{P}\left(e^{\lambda\tau} - 1 - \lambda\tau \geq u\right) du.$$

After the change of variable $u = \exp(\lambda v) - 1 - \lambda v$, we can use (5.9) and the fact that $\lambda(\exp(\lambda v - 1)) \geq 0$ in order to get:

$$\begin{aligned}
\mathbb{E}e^{\lambda\tau} - 1 - \lambda\mathbb{E}\tau &= \lambda \int_0^\infty \mathbb{P}(\tau \geq v)\left(e^{\lambda v} - 1\right) dv \\
&\leq A\lambda \int_0^\infty e^{-\gamma v}\left(e^{\lambda v} - 1\right) dv \\
&\leq \frac{A\lambda^2}{\gamma(\gamma - \lambda)}.
\end{aligned}$$

Using (5.8) it follows that :

$$\begin{aligned}
\mathbb{E}e^{\lambda\tau} &\leq 1 + \lambda\mathbb{E}\tau + \frac{A\lambda^2}{\gamma(\gamma-\lambda)} \\
&\leq \exp\left(\frac{\lambda}{\mu(s)} + \frac{A\lambda^2}{\gamma(\gamma-\lambda)}\right),
\end{aligned}$$

which concludes the proof of Lemma 14. □

5.6.4 Proof of Corollary 2

For any finite string $s \in \mathcal{A}^*$ let $g(s)$ denote the string obtained after removing the last letter of s, i.e. $g(s_1^l) = s_1^{l-1}$ and $g(\lambda) = \lambda$. For any set $\overleftarrow{\mathcal{B}} \subset \mathcal{A}^*$ let $\mathcal{T}(\overleftarrow{\mathcal{B}})$ be the smallest complete suffix tree such that for any $s \in \overleftarrow{\mathcal{B}}$ there exists a $s' \in \mathcal{T}(\overleftarrow{\mathcal{B}})$ which satisfies $s \prec s'$.

Let now :

$$\overleftarrow{\mathcal{D}}_g = \mathcal{T}\left(\bigcup_{i=0}^{D-1} g^i(\overleftarrow{\mathcal{D}})\right),$$

where $\overleftarrow{\mathcal{D}}$ is the complete suffix tree model used to define the stationary process $(T_n)_{n\in\mathbb{Z}}$.

Corollary 2 is now a direct consequence of the following lemma and of Lemma 13 applied to the Markov chain $(\overleftarrow{\mathcal{D}}_g(T_{-\infty}^n))_{n\in\mathbb{Z}}$:

Lemma 15 *Let* $S_n = \overleftarrow{\mathcal{D}}_g(T_{-\infty}^n)$. *The process* $(S_n)_{n\in\mathbb{Z}}$ *is a stationary* 1*-order Markov chain on the space* $\overleftarrow{\mathcal{D}}_g$*, and:*

$$|\overleftarrow{\mathcal{D}}_g| \leq D \times |\overleftarrow{\mathcal{D}}|.$$

Proof of Lemma 15:

The following properties are easily checked:

- If $\overleftarrow{\mathcal{B}}$ is a complete suffix tree then $\mathcal{T}(g(\overleftarrow{\mathcal{B}})) \subset g(\overleftarrow{\mathcal{B}})$.
- If $(\overleftarrow{\mathcal{B}}_i)_{i\in I}$ is a finite collection of complete suffix trees then $\mathcal{T}(\bigcup_{i\in I}\overleftarrow{\mathcal{B}}_i) \subset \bigcup_{i\in I}\overleftarrow{\mathcal{B}}_i$.
- If $(\overleftarrow{\mathcal{B}}_i)_{i\in I}$ is a finite collection of sets then $\mathcal{T}(\bigcup_{i\in I}\overleftarrow{\mathcal{B}}_i) = \mathcal{T}(\bigcup_{i\in I}\mathcal{T}(\overleftarrow{\mathcal{B}}_i))$.

As a result the following holds:

$$\begin{aligned}
\overleftarrow{\mathcal{D}}_g &= \mathcal{T}\left(\bigcup_{i=0}^{D-1} g^i(\overleftarrow{\mathcal{D}})\right) \\
&= \mathcal{T}\left(\bigcup_{i=0}^{D-1} \mathcal{T}\left(g^i(\overleftarrow{\mathcal{D}})\right)\right) \\
&\subset \bigcup_{i=0}^{D-1} \mathcal{T}\left(g^i(\overleftarrow{\mathcal{D}})\right) \\
&\subset \bigcup_{i=0}^{D-1} g^i(\overleftarrow{\mathcal{D}}),
\end{aligned}$$

and therefore

$$|\overleftarrow{\mathcal{D}}_g| \leq \sum_{i=0}^{D-1} |g^i(\overleftarrow{\mathcal{D}})| \leq D \times |\overleftarrow{\mathcal{D}}|.$$

Let us now show that $(S_n)_{n\in\mathbb{Z}}$ is a 1-order Markov chain. For any $s_{n+1} \in \overleftarrow{\mathcal{D}}_g$ there exists $i \in [0, D-1]$ such that $s_{n+1} \in g^i(\overleftarrow{\mathcal{D}})$. As a result $g(s_{n+1}) \in g^{i+1}(\overleftarrow{\mathcal{D}})$ and there exists $s \in \overleftarrow{\mathcal{D}}_g$ such that $g(s_{n+1}) \prec s$. Therefore the following events are equal:

$$\begin{aligned}
\left\{g(S_{n+1}) \prec T_{-\infty}^n\right\} &= \left\{g(S_{n+1}) \prec \overleftarrow{\mathcal{D}}_g\left(X_{-\infty}^n\right)\right\} \\
&= \left\{g(S_{n+1}) \prec S_n\right\}.
\end{aligned}$$

By definition, $S_{n+1} \prec T_{-\infty}^{n+1}$ and therefore $g(S_{n+1}) \prec T_{-\infty}^n$ almost surely. Hence $g(S_{n+1}) \prec S_n$ a.s. and therefore:

$$\mathbb{P}(S_{n+1} = s_{n+1} \,|\, S_1^n = s_1^n) = 0 \text{ if } g(s_{n+1}) \not\prec s_n.$$

On the other hand if $g(s_{n+1}) \prec s_n$ then there exists $a \in \mathcal{A}^*$ such that $s_n = ag(s_{n+1})$, and therefore there exists $b \in \mathcal{A}$ such that $as_{n+1} = s_n b$. Applying $\overleftarrow{\mathcal{D}}_g(.)$ on both sides we get $s_{n+1} = \overleftarrow{\mathcal{D}}_g(s_n b)$. We can now write for any $b \in \mathcal{A}$:

$$\begin{aligned}
&\mathbb{P}(S_{n+1} = \overleftarrow{\mathcal{D}}_g(s_n b) \,|\, S_{-\infty}^n = s_{-\infty}^n) \\
&\qquad = \mathbb{P}\left(T_{n+1} = b \,|\, T_{-\infty}^n = t_{-\infty}^n \;:\; \forall i \leq n \,,\, \overleftarrow{\mathcal{D}}_g(t_{-\infty}^i) = s_i\right) \\
&\qquad = \mathbb{P}\left(T_{n+1} = b \,|\, \overleftarrow{\mathcal{D}}(T_{-\infty}^n) = \overleftarrow{\mathcal{D}}(t_{-\infty}^n) \;:\; \forall i \leq n \,,\, \overleftarrow{\mathcal{D}}_g(t_{-\infty}^i) = s_i\right) \\
&\qquad = \mathbb{P}(T_{n+1} = b \,|\, \overleftarrow{\mathcal{D}}(T_{-\infty}^n) = s_n).
\end{aligned}$$

This shows that in all cases $\mathbb{P}(S_{n+1} = s_{n+1} \,|\, S_{-\infty}^n = s_{-\infty}^n) = \mathbb{P}(S_{n+1} = s_{n+1} \,|\, S_n = s_n)$ and therefore that S_n is a first-order Markov chain. It is obviously stationary because $(T_n)_{n\in\mathbb{Z}}$ is stationary itself. $\square$

Bibliography

[1] H. Akaike. Information theory and an extension of the maximum likelihood principle. In P.N. Petrov and F. Csaki (Eds.), editors, *Proceedings 2nd International Symposium on Information Theory*, pages 267–281, 1974.

[2] H. Alzer. On some inequalities for the gamma and psi functions. *Mathematics of Computation*, 66(217):373–389, January 1997.

[3] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, 1999.

[4] Lalit R. Bahl, Peter F. Brown, Peter V. de Souza, and Robert L. Mercer. A tree-based statistical language model for natural language speech recognition. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 37:1001–1008, July 1989.

[5] Andrew Barron. Are bayes rules consistent in information ? In *Open Problems in Communication and Computation, T.M. Cover and B. Gopinath Ed.* Springer Verlag, 1987.

[6] Andrew Barron, Lucien Birgé, and Pascal Massart. Risk bounds for model selection via penalization. *Probability Theory and Related Fields*, 113(3):301–413, 1999.

[7] Andrew Barron, Jorma Rissanen, and Bin Yu. The minimum description length principle in coding and modeling. *IEEE Trans. Inform. Theory*, 44(6):2743–2760, October 1998.

[8] Andrew Barron and Yuhong Yang. Information-theoretic determination of minimax rates of convergence. *The Annals of Statistics*, 27(5):1564–1599, 1999.

[9] Andrew R. Barron and Thomas M. Cover. Minimum complexity density estimation. *IEEE Trans. Inform. Theory*, 37(4):1034–1054, July 1991.

[10] T.C. Bell, J.G. Cleary, and I.H. Witten. *Text compression.* Prentice Hall, Englewood Cliffs, NJ, 1990.

[11] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proceedings of the Twenty-Second Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-99)*, 1999.

[12] Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1), 1996.

[13] Lucien Birgé. Approximation dans les espaces métriques et théorie de l'estimation. *Z. Wahrscheinlichkeitstheor. Verw. Geb.*, 65:181–237, 1983.

[14] Lucien Birgé and Pascal Massart. Rates of convergence for minimum contrast estimators. *Probability Theory and Related Fields*, 97:113–150, 1993.

[15] Lucien Birgé and Pascal Massart. From model selection to adaptive estimation. In *Festschrift for Lucien Le Cam*, pages 55–87. Springer, New-York, 1997.

[16] Gilles Blanchard. The "progressive mixture" estimator for regression trees. *Ann. Inst. Henri Poincaré, Probabilités et Statistiques*, 35(6):793–820, 1999.

[17] Leo Breiman. Bagging predictors. Technical Report 421, Statistics Department, University of California at Berkeley, September 1994.

[18] P. Brown, J. Cocke, S. Della Pietra, V. Della Pietra, F. Jelinek, J. Lafferty, R. Mercer, and P. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, 1990.

[19] Peter F. Brown, Stephen A. Della Pietra, vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, June 1993.

[20] Peter Bülmann and Abraham J. Wyner. Variable length markov chains. *The Annals of Statistics*, 27(2), April 1999.

[21] Olivier Catoni. Data compression and adaptive histograms. University Paris 6, Preprint PMA-609, Sep. 2000. Available at `http://www.proba.jussieu.fr/mathdoc/preprints`.

[22] Olivier Catoni. Gibbs estimators. Ecole normale supérieure, Preprint LMENS-98-21, May 1998, pp. 1-23. Available at `http://www.dma.ens.fr/edition`.

[23] Olivier Catoni. A mixture approach to universal model selection. Ecole normale supérieure de Paris, Preprint LMENS - 97 - 30, Oct. 1997, pp. 1-19. Available at `http://www.dma.ens.fr/edition.`, 1997.

[24] Olivier Catoni. "Universal" aggregation rules with exact bias bounds. *to appear in the Annals of Statistics*, 1999. Preprint PMA-510 available at `http://www.proba.jussieu.fr/mathdoc/preprints`.

[25] Stanley F. Chen, Douglas Beeferman, and Ronald Rosenfeld. Evaluation metrics for language models. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pages 275–280, 1998.

[26] Bertrand S. Clarke and Andrew R. Barron. Information-theoretic asymptotics of bayes methods. *IEEE Trans. Inform. Theory*, 36(3):453–471, May 1990.

[27] Bertrand S. Clarke and Andrew R. Barron. Jeffrey's prior is asymptotically least favorable under entropy risk. *Journal of Statistical Planning and Inference*, 41:37–60, 1994.

[28] Ronald Cole, Joseph Mariani, Hans Uszkoreit, Giovanni Battista Varile, Annie Zaenen, and Antonio Zampolli, editors. *Survey of the State of the Art in Human Language Technology.* Cambridge University Press, 1998.

[29] Thomas M. Cover and Joy A. Thomas. *Elements of information theory.* Wiley, 1991.

[30] Lee D. Davisson. Universal noiseless coding. *IEEE Trans. Inform. Theory*, 19(6):783–795, November 1973.

[31] Lee D. Davisson. Minimax noiseless universal coding for markov sources. *IEEE Trans. Inform. Theory*, 29(2):211–215, March 1983.

[32] Lee D. Davisson and Alberto Leon-Garcia. A source matching approach to finding minimax codes. *IEEE Trans. Inform. Theory*, 26(2):166–174, March 1980.

[33] A.L. Delcher, D. Harmon, S. Kasif, O. White, and S.L. Salzberg. Improved microbial gene identification with glimmer. *Nucleic Acids Research*, 27(23):4636–4641, 1999.

[34] L. Devroye, L. Gyorfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition.* Springer-Verlag, New-York, 1996.

[35] D. Donoho and I. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

[36] David L. Donoho, Iain M. Johnstone, Gérard Kerkyacharian, and Dominique Picard. Density estimation by wavelet thresholding. *The Annals of Statistics*, 24(2):508–539, 1996.

[37] David L. Donoho, Iain M. Johnstone, Gérard Kerkyacharian, and Dominique Picard. Universal near minimaxity of wavelet shrinkage. In *Festschrift for Lucien Le Cam*, pages 183–218. Springer, New-York, 1997.

[38] Susan T. Dumais, John Platt, David Heckerman, and Mehran Sahami. Inductive learning algorithms and representations for text categorization. In Georges Gardarin, James C. French, Niki Pissinou, Kia Makki, and Luc Bouganim, editors, *Proceedings of CIKM-98, 7th ACM International Conference on Information and Knowledge Management*, pages 148–155, Bethesda, US, 1998. ACM Press, New York, US.

[39] S. Yu. Efroimovich. Nonparametric estimation of a density of unknown smoothness. *Theory Probab. Appl.*, 30:557–568, 1995.

[40] S. Yu. Efroimovich and M. S. Pinsker. A self-educating nonparametric filtration algorithm. *Automat. Remote Control*, pages 58–65, 1984.

[41] Meir Feder and Neri Merhav. Hierarchical universal coding. *IEEE Trans. Inform. Theory*, 42(5):1354–1364, September 1996.

[42] R. G. Gallager. Source coding with side information and universal coding. Unpublished manuscript; also presented at the *Int. Symp. Information Theory*, Oct. 1974.

[43] D. Haussler and M. Opper. Mutual information, metric entropy and cumulative relative entropy risk. *The Annals of Statistics*, 25:2451–2492, 1997.

[44] David Haussler. A general minimax result for relative entropy. *IEEE Trans. Inform. Theory*, 43:1276–1280, July 1997.

[45] W. Härdle and J. S. Marron. Optimal bandwidth selection in nonparametric regression function estimation. *The Annals of Statistics*, 13:1465–1481, 1985.

[46] Frederick Jelinek. *Statistical Methods for Speech Recognition.* MIT Press: Cambridge, 1998.

[47] Frederick Jelinek, John D. Lafferty, and Robert L. Mercer. Basic methods of probabilistic context-free grammars. In P. Laface and R. De Mori, editors, *Speech recognition and understanding: recent advances, trends and applications*, volume 75 of *Computer and Systems Science*, pages 354–360. Springer Verlag, 1992.

[48] Frederick Jelinek and Kenneth S. Schneider. On variable-length-to-block coding. *IEEE Trans. Inform. Theory*, 18(6):765–774, May 1972.

[49] Thorsten Joachims. A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization. In Douglas H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 143–151, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US.

[50] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, number 1398 in Lecture Notes in Computer Science, pages 137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE.

[51] Thorsten Joachims. Transductive inference for text classification using support vector machines. In Ivan Bratko and Saso Dzeroski, editors, *Proceedings of ICML-99, 16th International Conference on Machine Learning*, pages 200–209, Bled, SL, 1999. Morgan Kaufmann Publishers, San Francisco, US.

[52] Raphail E. Krichevsky and Victor K. Trofimov. The performance of universal encoding. *IEEE Trans. Inform. Theory*, 27(2):199–207, March 1981.

[53] Rafail E. Krichevskyi. Laplace's law of succession and universal encoding. *IEEE Trans. Inform. Theory*, 44(1):296–303, January 1998.

[54] J.C. Lawrence. A new universal coding scheme for the binary memoryless source. *IEEE Trans. Inform. Theory*, 23(4):466–472, 1977.

[55] David D. Lewis. Naive (Bayes) at forty: The independence assumption in information retrieval. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, number 1398 in Lecture Notes in Computer Science, pages 4–15, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE.

[56] C.L. Mallows. Some comments on c_p. *Technometrics*, 15:661–675, 1973.

[57] Andrew McCallum and Kamal Nigam. A comparison of event models for naive bayes text classification. In *Proceedings of the AAAI/ICML-98 Workshop on Learning for Text Categorization*, pages 41–48, 1998.

[58] Neri Merhav and Meir Feder. Universal prediction. *IEEE Trans. Inform. Theory*, 44(6):2124–2147, October 1998.

[59] Isabelle Moulinier, Gailius Raškinis, and Jean-Gabriel Ganascia. Text categorization: a symbolic approach. In *Proceedings of SDAIR-96, 5th Annual Symposium on Document Analysis and Information Retrieval*, pages 87–99, Las Vegas, US, 1996.

[60] Arkadi Nemirovski. Topics in non-parametric statistics. In P. Bernard, editor, *Lectures on Probability Theory and Statistics, Ecole d'Eté de Probabilités de Saint-Flour XXVIII - 1998*, volume 1738 of *Lecture Notes in Mathematics*, pages 85–277. Springer, 2000.

[61] A.S. Nemirovskii, B.T. Polyak, and A.B. Tsybakov. Rate of convergence of nonparametric estimates of maximum-likelihood method. *Problems of Information Transmission*, 21:258–272, 1985.

[62] Kamal Nigam, John Lafferty, and Andrew McCallum. Using maximum entropy for text classification. In *Proceedings of the IJCAI-99 Workshop on Machine Learning for Information Filtering*, pages 61–67, 1999.

[63] Kamal Nigam, Andrew K. McCallum, Sebastian Thrun, and Tom M. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134, 2000.

[64] A. Nikiforov and V. Ouvarov. *Fonctions spéciales de la physique mathématique*. Mir, 1983.

[65] Jay A. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the Twenty-First Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-98)*, pages 275–281, 1998.

[66] Jorma Rissanen. Minimax codes for finite alphabets. *IEEE Trans. Inform. Theory*, 24(3):389–392, May 1978.

[67] Jorma Rissanen. A universal data compression system. *IEEE Trans. Inform. Theory*, 29:656–664, September 1983.

[68] Jorma Rissanen. Fast universal coding with context models. *IEEE Trans. Inform. Theory*, 45(4), May 1999.

[69] Jorma Rissanen and Glen G. Langdon, Jr. Universal modeling and coding. *IEEE Trans. Inform. Theory*, 27(1):12–23, January 1981.

[70] Ronald Rosenfeld. Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE*, 88(8), 2000.

[71] B. Ya. Ryabko. Twice-universal coding. *Problems of Information Transmission*, 20(3):24–28, July 1984.

[72] B. Ya. Ryabko. Prediction of random sequences and universal coding. *Problems of Information Transmission*, 24(2):87–96, 1988.

[73] B. Ya. Ryabko. A fast adaptive coding algorithm. *Problems of Information Transmission*, 26(4):305–317, 1990.

[74] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, November 1975.

[75] Robert E. Schapire and Yoram Singer. BoosTexter: a boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.

[76] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6:461–464, 1978.

[77] C.E. Shannon. Prediction and entropy of printed english. *Bell System Technical Journal*, 30:50–64, 1951.

[78] CE Shannon and W Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, Urbana, Illinois, 1949.

[79] P. C. Shields. *The ergodic theory of discrete sample paths.* AMS Graduate Series in Mathematics, volume GS-13. American Mathematical Society, 1996.

[80] Seán Slattery and Mark Craven. Combining statistical and relational methods for learning in hypertext domains. In David Page, editor, *Proceedings of ILP-98, 8th International Conference on Inductive Logic Programming*, number 1446 in Lecture Notes in Computer Science, pages 38–52, Madison, US, 1998. Springer Verlag, Heidelberg, DE.

[81] C. J. Stone. Optimal global rates of convergence for nonparametric regression. *The Annals of Statistics*, 10(4):1040–1053, 1982.

[82] Tjalling J. Tjalkens and Frans M.J. Willems. A universal variable-to-fixed lenght source code based on lawrence's algorithm. *IEEE Trans. Inform. Theory*, 38(2):247–253, 1992.

[83] Flemming Topsoe. Instances of exact prediction and a new type of inequalities obtained by anchoring. In *Proceedings 1999 IEEE Information Theory and Communication Workshop*, page 99. Kruger National Park, 1999.

[84] B.P. Tunstall. Synthesis of noiseless compression codes. Ph.D. dissertation, September 1967.

[85] Sara Van De Geer. Estimating a regression function. *The Annals of Statistics*, 18:907–924, 1990.

[86] Vladimir N. Vapnik. *Statistical learning theory.* Wiley, 1998.

[87] Jean-Philippe Vert. Adaptive context trees and text clustering. To appear in IEEE Trans. Inform. Theory, 2001.

[88] Jean-Philippe Vert. Text categorization using adaptive context trees. In A. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing, Proceedings of the Second International Conference, CICLing 2001*, volume 2004 of *LNCS*, pages 423–436, Mexiko City, Mexico, February 2001. Springer-Verlag.

[89] Marcelo J. Weinberger, Abraham Lempel, and Jacob Ziv. A sequential algorithm for the universal coding of finite memory sources. *IEEE Trans. Inform. Theory*, 38(3):1002–1014, May 1992.

[90] Marcelo J. Weinberger, Jorma J. Rissanen, and Meir Feder. A universal finite memory source. *IEEE Trans. Inform. Theory*, 41(3):643–652, May 1995.

[91] Frans M.J. Willems, Yuri M. Shtarkov, and Tjalling J. Tjalkens. The context tree weighting method : Basic properties. *IEEE Trans. Inform. Theory*, 41(3):653–664, May 1995.

[92] Frans M.J. Willems, Yuri M. Shtarkov, and Tjalling J. Tjalkens. Context weighting for general finite-context sources. *IEEE Trans. Inform. Theory*, 42:1514–1520, September 1996.

[93] Qun Xie and Andrew R. Barron. Asymptotic minimax regret for data compression, gambling, and prediction. *IEEE Trans. Inform. Theory*, 46(2):431–445, March 2000.

[94] Yiming Yang. An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1(1-2):69–90, 1999.

[95] Yuhong Yang. Mixing strategies for density estimation. *The Annals of Statistics*, 28(1):75–87, 2000.