Minimax Optimal Bayes Mixtures for Memoryless Sources

Elias Jääsaari

MSc Thesis
UNIVERSITY OF HELSINKI
Department of Computer Science

Helsinki, September 14, 2017

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Laitos — Institution — Department
Faculty of Science	Department of Computer Science

Tekijä — Författare — Author
Elias Jääsaari

Työn nimi — Arbetets titel — Title

Minimax Optimal Bayes Mixtures for Memoryless Sources

Oppiaine — Läroämne — Subject
Computer Science

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
MSc Thesis	September 14, 2017	40

Tiivistelmä — Referat — Abstract

Tasks such as data compression and prediction commonly require choosing a probability distribution over all possible sequences. To achieve an efficient prediction strategy, the chosen distribution should be a good approximation of the true distribution underlying the data. Similarly, an efficient compression strategy should assign shorter codes for more probable sequences. In particular, a compression strategy that minimizes the code-length can be shown to minimize the often-used logarithmic prediction loss. However, the optimal strategy requires knowing the true distribution which is not available in most applications.

In universal compression or prediction we assume that the true probability distribution is not known but belongs to a known class of distributions. A universal code is a code that can compress the data essentially as well as the best distribution in the class in hindsight. Similarly, a universal predictor achieves low prediction loss regardless of the distribution. We call a universal code minimax optimal if it minimizes the worst-case regret, i.e. excess code-length or prediction loss compared to the best distribution in the class.

In this thesis we assume the known class to be discrete memoryless sources. The minimax optimal code for this class is given by the normalized maximum likelihood (NML) distribution. However, in practice computationally more efficient distributions such as Bayes mixtures have to be used. A Bayes mixture is a mixture of the probability distributions in the class weighted by a prior distribution. The conjugate prior to the multinomial distribution is the Dirichlet distribution, using which asymptotically minimax codes have been developed. The Dirichlet distribution requires a hyperparameter that dictates the amount of prior mass given to the outcomes. The distribution given by the symmetric hyperparameter 1/2 has been widely studied and has been shown to minimize the worst-case expected regret asymptotically.

Previous work on minimax optimal Bayes mixtures has mainly been concerned with large sample sizes in comparison to the alphabet size. In this thesis we investigate the minimax optimal Dirichlet prior in the large alphabet setting. In particular, we find that when the alphabet size is large compared to the sample size, the optimal hyperparameter for the Dirichlet distribution is 1/3. The worst-case regret of this mixture turns out to approach the NML regret when the alphabet size grows and the distribution provides an efficient approximation of the NML distribution. Furthermore, we develop an efficient algorithm for finding the optimal hyperparameter for any sample size or alphabet size.

Avainsanat — Nyckelord — Keywords
universal coding, Bayes mixtures, memoryless source, large alphabet

Säilytyspaikka — Förvaringsställe — Where deposited

Muita tietoja — Övriga uppgifter — Additional information

Contents

1 Introduction

In tasks such as prediction and data compression, choosing a probability distribution over all sequences is commonly required. For example, consider the problem of predicting the next observation x_{n+1} given a sequence of past observations $x_1, x_2, \ldots, x_n$. If we have access to a prediction strategy that assigns probabilities to the different possibilities of the next observation, we can use it to predict the most probable one.

The prediction problem was initially studied by Laplace who considered the question "What is the probability that the sun will rise tomorrow?" in the 18th century [19]. Given that the sun has risen k times in the past, he derived the formula $(k+1)/(k+2)$ for the probability of the sun rising tomorrow. The more general Laplace's rule of succession gives probability $(k+1)/(n+2)$ for an event with k occurrences in n trials in the past.

Optimal prediction can be linked to optimal compression. In particular, a good prediction strategy under the logarithmic loss is also a good compression strategy. To compress a sequence $x_1, \ldots, x_n$ of symbols we can assign codes for each of the symbols such that more probable symbols get shorter code-lengths. If we know the true distribution of the data, Shannon showed that expected code-length of essentially entropy can be achieved [30], and that entropy is the lower bound for the expected code-length.

However, in most cases the true distribution is not known. In universal compression we assume that the true distribution is unknown but belongs to a known class of distributions. A universal code is a code that can compress the data efficiently regardless of which distribution in the class was used to generate the data. Similarly, in universal prediction we can achieve low prediction loss no matter which distribution is the true distribution.

In this thesis we focus on the most often studied class of distributions, i.i.d. distributions over sequences of length n drawn from an alphabet of size m, also known as discrete memoryless sources. The Laplace estimator turns out to be a universal predictor for this class. However, it is not optimal in the sense that there are universal predictors that have lower worst-case regret. Here regret means excess logarithmic loss or code-length compared to the best model in the class in hindsight. The minimax regret problem is to find the universal predictor or code that minimizes the regret in the worst case. Minimax optimality is a strong performance guarantee and ensures that the regret is minimal even in the worst cases.

The optimal solution to the minimax regret problem for the class of discrete memoryless sources is known as the normalized maximum likelihood (NML) distribution [31]. The NML distribution has been used in for example code-lengths in model selection with the minimum description length (MDL) principle [12]. The MDL principle advocates choosing the model that results in the shortest total code-length for the model and the data encoded using a universal code, such as the NML distribution, for the model.

Using the NML distribution is often infeasible in practice in tasks such as prediction and data compression since they require calculating conditional probabilities. For the NML distribution, obtaining the conditional probabilities takes exponential time. Therefore, other distributions such as Bayes mixtures that come close to the performance of the NML distribution in the worst case have been studied. In a Bayes mixture a weighted mixture of the probabilities is taken in the assumed class of probability distributions which is parametrized by some parameter set. The weights in the mixture are given by a chosen prior distribution on the parameters.

The conjugate prior for the multinomial model is the Dirichet distribution. Krichevsky and Trofimov suggested [18] using the Bayes mixture with a Dirichlet prior $\mathrm{Dir}(1/2, \ldots, 1/2)$ to minimize the expected regret. The resulting prediction strategy is similar to Laplace's rule of succession and assigns probability $(k + 1/2)/(n + 1)$ for an event with k occurrences in n trials in the past. Laplace's rule can be derived similarly by considering as the prior the $\mathrm{Dir}(1, \ldots, 1)$ distribution, which is the uniform distribution.

Xie and Barron showed [42] that while the Krichevsky-Trofimov estimator achieves lower worst-case regret than Laplace's estimator, it is not asymptotically minimax. That is, its worst-case regret does not necessarily converge to the minimax regret as the sequence length grows. Instead, Xie and Barron showed that the Krichevsky-Trofimov estimator modified by adding mass to the boundaries of the probability simplex is asymptotically minimax. Later Watanabe and Roos proved [38] that a Bayes mixture with a Dirichlet prior dependent on the sequence length achieves asymptotic minimaxity.

The previous results require that the sample size of the data grows faster than the alphabet size. In recent years large alphabet methods have been gaining more attention [43]. The alphabet size can be larger than the sample size or even infinite in application areas such as natural language processing, population estimation, genetics [8] and Bayesian network structure learning [32]. Images can also be considered as data associated with a large alphabet where each pixel can take on 2^{24} different values.

Various strategies for data compression on large alphabets have subsequently been proposed. Universal codes of i.i.d. distributions over infinite alphabets have infinite regret [15] since describing the symbols that appear in the sequence requires an unbounded number of bits. Therefore the work on universal compression of large alphabets has focused on subclasses of i.i.d. distributions such as envelope classes [2, 6] and patterns [23, 26].

However, as codes for these subclasses target a different distribution, their code-lengths are not directly interchangeable with code-lengths for i.i.d. distributions and thus they are not useful in for example model selection. A coding distribution for the i.i.d. class is still needed to calculate a target minimax distribution. Therefore such distributions have recently been proposed for large alphabets [45]. Distributions for i.i.d. classes can also be extended to models that incorporate context such as Markov sources [44].

In this thesis we investigate the optimal Dirichlet prior for the Bayes mixture when the size of the alphabet is large compared to the sample size. We also consider the problem of finding the optimal hyperparameter for any given value of the sample size n and the alphabet size m. More specifically, the contributions of this thesis are as follows:

- We prove that the $\mathrm{Dir}(1/3, \ldots, 1/3)$ prior is the minimax optimal Dirichlet prior for the Bayes mixture when the alphabet size m is large compared to the sample size n. Furthermore, we prove that this property holds not only asymptotically, but also derive a finite bound for it. In particular, we prove that it holds when $m > \frac{5}{2}n + \frac{4}{n-2} + \frac{3}{2}$.
- We prove results on the asymptotic behavior of the 1/3-mixture. These asymptotic results prove that the worst-case regret of the 1/3-mixture converges to that of the NML distribution when m increases. This result also gives a constant-time approximation of the NML regret that is accurate for large values of m.
- We compare the worst-case regret of the 1/3-mixture numerically to the worst-case regrets of other distributions. The numerical experiments suggest that the 1/3-mixture can be preferable to the 1/2-mixture or the Bayes procedure given by the asymptotic formula of Watanabe and Roos already when the alphabet size is large but not larger than the sample size. These comparisons also extend the comparisons of Watanabe and Roos [38] for larger alphabets.
- We present an algorithm for calculating the optimal hyperparameter α with ε precision for any n, m in time $\mathcal{O}(\log(\min\{n, m\}/\varepsilon))$, which is an improvement to the brute-force exponential time algorithm and to an algorithm that works in time $\mathcal{O}(\min\{n, m\}/\varepsilon)$ using previously proven results [38]. This algorithm makes it practical to calculate the optimal hyperparameter efficiently for any feasible values of n and m.

This thesis is structured as follows. In Section 2 we present the mathematical preliminaries needed for understanding the rest of the thesis. We also motivate the minimax regret problem by introducing its connection to data compression, prediction and model selection. In Section 3 we introduce Bayes mixtures and review previous work involving them in the context of universal compression. Finally, we derive an efficient algorithm for finding the minimax optimal Bayes procedure with a Dirichlet prior.

In Section 4 we review previous work on large alphabet methods and then derive the optimal Dirichlet prior for the Bayes mixture for the large alphabet case. Furthermore, we prove properties on the asymptotic behavior of the 1/3-mixture. In Section 5 we compare the worst-case regrets of different distributions numerically. Finally, Section 6 discusses the implications of the results of this thesis and suggests possibilities for future work.

2 Preliminaries

This section describes preliminaries needed for understanding the rest of this thesis. In particular, we introduce the relevant mathematical concepts. We also motivate the problem of finding the minimax optimal distribution by describing its relation to coding, prediction and model selection.

2.1 Mathematical preliminaries

We denote both probability distributions and their corresponding probability mass functions by lower case letters $p, q, \ldots$. The expectation of random variable X where the expectation is taken over the variable X and the relevant distribution is p, is denoted by $\mathbb{E}_{X\sim p}[X]$. The subscript is omitted when it is clear from the context.

We define the entropy [9, p. 14] of a random variable X as

$$H(X) = \mathbb{E}[\log_2 \frac{1}{p(X)}] = \sum_{x\in\mathcal{X}} p(x) \log_2 \frac{1}{p(x)},$$

where the quantity $-\log_2 p(X)$ can be seen as the amount of information content or surprise contained in the random variable X. The less probable an outcome is, the more surprising it is. Here the base 2 of the logarithm means the entropy is measured in bits. When the subscript is omitted, we refer to the natural logarithm and measure the entropy in nats.

The relative entropy or Kullback-Leibler (KL) divergence [9, p. 19] is often used as a measure of difference between probability distributions. The KL divergence between probability distributions p and q is defined as

$$D_{KL}(p \parallel q) = \sum_{x\in\mathcal{X}} p(x) \log_2 \frac{p(x)}{q(x)}.$$

Particularly, $D_{KL}(p \parallel q) = 0$ if and only if $p = q$. However, the KL divergence is not a proper distance measure as it is not symmetric.

The multinomial distribution is a generalization of the binomial distribution for any number of possible outcomes. That is, for each of n trials there are m possible outcomes. Formally, we define a multinomial model with parameters $\theta = (\theta_1, \ldots, \theta_m)$ with

$$p(x_j|\theta) = \theta_j, \qquad \sum_{j=1}^{m} \theta_j = 1.$$

Now the probability of outcomes $x_1, \ldots, x_m$ having counts $n_1, \ldots, n_m$ such that $\sum_{j=1}^{m} n_j = n$, is given by

$$p(n_1, \ldots, n_m) = \frac{n!}{n_1! \cdots n_m!} \theta_1^{n_1} \cdots \theta_m^{n_m}.$$

The conjugate prior for the multinomial distribution is the Dirichlet distribution. Here a prior distribution is called a conjugate prior if the prior and the posterior are from the same distribution. The Dirichlet distribution $\mathrm{Dir}(\alpha_1, \ldots, \alpha_K)$ is characterized by the vector $(\alpha_1, \ldots, \alpha_K)$ of hyperparameters. In the symmetric case we denote $\alpha = \alpha_1 = \cdots = \alpha_K$. Intuitively, these hyperparameters denote how much prior probability is given to each category. The probability mass function of the Dirichlet distribution is given by

$$\frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} x_i^{\alpha_i - 1},$$

where Γ is the gamma function

$$\Gamma(z) = \int_0^\infty x^{z-1} e^{-x}\, dx,$$

which extends the factorial function for all real numbers. That is, the gamma function satisfies the equation $\Gamma(z+1) = z\Gamma(z)$. In particular, for all natural numbers n, we have $\Gamma(n) = (n-1)!$. The logarithm of the gamma function can be approximated using Stirling's approximation [29]

$$\log \Gamma(z) = z \log z - z + \frac{1}{2} \log \frac{2\pi}{z} + \varepsilon(z),$$

where $1/(12z+1) < \varepsilon(z) < 1/(12z)$.

The logarithmic derivatives of the gamma function also play an important role in this thesis. We define the mth order polygamma function as the $(m+1)$th derivative of the logarithm of the gamma function:

$$\psi^{(m)}(x) = \frac{d^{m+1}}{dx^{m+1}} \log \Gamma(x)$$

The 0th order polygamma function $\psi^{(0)}$ is known as the digamma function and shortened as ψ. The digamma function satisfies the important property

$$\psi(x+1) = \psi(x) + \frac{1}{x}.$$

The logarithm of the gamma function is an example of a convex function. A function f is convex if its second derivative is non-negative. Intuitively, a line between any two points on a convex function is always above the function. Correspondingly, a function f is called concave if $-f$ is convex. We call a function f quasiconvex if for any two points x, y we have

$$f(\lambda x + (1-\lambda)y) \leq \max\{f(x), f(y)\},$$

where $0 \leq \lambda \leq 1$. That is, the function evaluated between two points does not give a higher value than either of the two points do. If it always gives a lower value, we call the function f strictly quasiconvex. In particular, all monotone functions and functions that decrease up to a point and increase from that point on are strictly quasiconvex.

2.2 Compression and prediction

In data compression we want to find for given data a representation that is as short as possible. Suppose the data is a sequence of symbols, which can be for example characters or words. A symbol code defines a representation for each of the possible symbols. Ideally such a code would assign shorter representations for more probable symbols.

Formally, a symbol code C is a mapping $C : \mathcal{X} \mapsto \{0,1\}^*$ from the alphabet $\mathcal{X}$ to all bitstrings. The extension C^* of a code C is the concatenation $C^*(x_1, \ldots, x_n) = C(x_1) \cdots C(x_n)$ of n symbols. We call a code uniquely decodable if its extension is a one-to-one mapping. The Kraft-McMillan inequality [9, p. 107] states that integers $l_1, l_2, \ldots$ can represent the code-lengths of a uniquely decodable code if and only if they satisfy

$$\sum_{i=1}^{\infty} 2^{-l_i} \leq 1.$$

This result allows the unification of code-lengths and probabilities. Namely, for any code lengths $l_1, l_2, \ldots$ there exists a probability distribution q such that $q(x) = 2^{-\ell(C(x))}$ and for any probability distribution q there exists a uniquely decodable code with code-lengths given by $\ell(C(x)) = -\log_2 q(x)$. If the sum of the probabilities is less than one, it is possible to make the code-lengths shorter. Thus for simplicity we assume that the sum is one.

Let the true probabilities of the symbols be described by the probability distribution p and we specify code-lengths using a distribution q. The expected code-length is now

$$\begin{aligned}
\mathbb{E}[\ell(C(X))] &= \sum_{x \in \mathcal{X}} p(x) \log_2 \frac{1}{q(x)} \\
&= \sum_{x \in \mathcal{X}} p(x) \log_2 \frac{p(x)}{p(x)q(x)} \\
&= \sum_{x \in \mathcal{X}} p(x) \left[\log_2 \frac{1}{p(x)} + \log_2 \frac{p(x)}{q(x)} \right] \\
&= H(X) + D_{KL}(p \parallel q).
\end{aligned}$$

The Gibbs inequality [22, p. 34] states that $D_{KL}(p \parallel q) \geq 0$, where equality stands if and only if $p = q$. Thus the expected per-symbol code-length is lower-bounded by the entropy and the optimal code-lengths are given by

$$\ell(C(x)) = \log_2 \frac{1}{p(x)}.$$

Example 1. *Let the possible symbols be $\mathcal{X} = \{a, b, c, d\}$ with probabilities $p(a) = 1/2, p(b) = 1/8, p(c) = 1/4, p(d) = 1/8$. The optimal codeword lengths are then $\ell(C(a)) = 1, \ell(C(b)) = 3, \ell(C(c)) = 2, \ell(C(d)) = 3$. A possible code with such lengths is $C(a) = 1, C(b) = 000, C(c) = 01, C(d) = 001$.*

In practice code-lengths have to be integers. One possible way to avoid this issue is to simply round the lengths upwards to the next integer. However, this method can in the worst case double the size of the coded sequence as each symbol can incur a maximum of one bit extra length.

An often better solution is to code the data in blocks of length n. We denote a sequence $x_1, x_2, \ldots, x_n$ as x^n where each x_i is a member of an alphabet $\mathcal{X}$ of finite size m. The code-length of each block is then given by

$$\log_2 \frac{1}{p(x_1, \ldots, x_n)} = \log_2 \frac{1}{p(x^n)}$$

and the rounding up has to be done only once for each block, thus incurring in the worst case a maximum of one extra bit per block.

However, block codes still have problems. Namely, they still incur an extra bit per block and cannot be decoded instantaneously. Arithmetic coding [40] represents sequences as intervals $[a, b) \subset [0, 1)$. A narrower interval requires more bits to describe. Each symbol divides the interval into smaller sub-intervals whose lengths are based on the conditional probabilities

$$p(x_{n+1}|x^n).$$

Arithmetic coding allows instantaneous coding and can spread the extra bit from rounding across the whole sequence. For the rest of the thesis, we ignore the integer requirement and allow code-lengths to be non-integer.

The quantity $-\log_2 p(x^n)$ and conditional probabilities also occur in the context of prediction. Assume that we have a sequence of observations x^n and we wish to predict the next observation x_{n+1} based on the past observations x^n. When x_{n+1} occurs we measure the loss by $\log_2 1/p(x_{n+1}|x^n)$ which equals zero if and only if $p(x_{n+1}|x^n) = 1$. A good predictor should now minimize the often-used cumulative logarithmic loss (log-loss)

$$\sum_{k=0}^{n-1} \log_2 \frac{1}{p(x_{k+1}|x^k)} = \log_2 \frac{1}{p(x_1, \ldots, x_n)},$$

where

$$p(x_1, \ldots, x_n) = \prod_{k=0}^{n-1} p(x_{k+1}|x^k).$$

As we want small code-lengths and a small cumulative log-loss, a good compressor is also a good predictor. Namely if we can compress the data well, we have learned something meaningful from it.

The logarithmic loss has been used for example in online learning [10]. Sequential prediction minimizing the log-loss can also be connected to maximizing benefits in the stock market [7]. Finally, we note that there exists a similar connection between gambling and compression [42].

2.3 Universal coding

We can achieve optimal code-lengths only if we know the true distribution for the symbols. We can always estimate the probabilities upon seeing the data. However, in situations such as when the sequence length is very large or the sequence cannot be stored this might not be feasible. Thus it is desirable to have a one-pass algorithm that compresses the data by learning the distribution of the symbols [9, p. 427].

We assume that the true distribution is unknown but belongs to a known class of distributions $\mathcal{P}$. Universal compression methods compress the data well no matter which distribution in $\mathcal{P}$ the data is generated from. If we do not put any restrictions on the class of distributions, there is always a probability distribution that compresses the data to one bit.

For a distribution q, we define its regret relative to the sequence x^n as the excess code-length or log-loss compared to the distribution that maximizes its probability in hindsight:

$$\text{regret}(q, x^n) = \max_{p \in \mathcal{P}} \left[\log_2 \frac{1}{q(x^n)} - \log_2 \frac{1}{p(x^n)} \right] = \max_{p \in \mathcal{P}} \log_2 \frac{p(x^n)}{q(x^n)}$$

The most often studied class is the class of discrete memoryless sources or i.i.d. distributions which we adapt in this thesis. For this class, we have the parameter $\theta = (\theta_1, \ldots, \theta_m)$ such that $p(x|\theta) = \theta_x$. The maximum likelihood parameter $\hat{\theta}$ is defined as the parameter that maximizes the likelihood $p_\theta(x^n) = p(x^n|\theta(x^n))$. Thus the shortest code-length or least log-loss in hindsight for data x^n is achieved by the maximum likelihood model:

$$\min_{\theta \in \Theta} \ell(C_\theta(x^n)) = \min_{\theta \in \Theta} \log_2 \frac{1}{p_\theta(x^n)} = \log_2 \frac{1}{p_{\hat{\theta}}(x^n)}$$

We can now define the regret for the class of i.i.d. distributions as

$$\text{regret}(q, x^n) = \log_2 \frac{1}{q(x^n)} - \log_2 \frac{1}{p_{\hat{\theta}}(x^n)} = \log_2 \frac{p_{\hat{\theta}}(x^n)}{q(x^n)}.$$

Note that the maximum likelihood distribution cannot be used for coding as the sum of its probabilities exceeds 1 in all but trivial cases and thus it does not define a proper probability distribution.

A universal code (or a universal model) is a sequence of distributions $q_1, q_2, \ldots$ such that the per-symbol regret diminishes to zero for all possible sequences x^n as the sequence length n approaches infinity:

$$\lim_{n \to \infty} \frac{1}{n} \text{regret}(q^n, x^n) = 0$$

A weaker condition is that $\frac{1}{n} D_{KL}(p_\theta \parallel q^n) \mapsto 0$ for all $\theta \in \Theta$. That is, the per-symbol Kullback-Leibler divergence to every distribution p_θ in $\mathcal{P}$ shrinks to zero as the sample size increases. Thus, the universal code is in a sense never too far from any of the distributions in the class $\mathcal{P}$ (Figure 1).

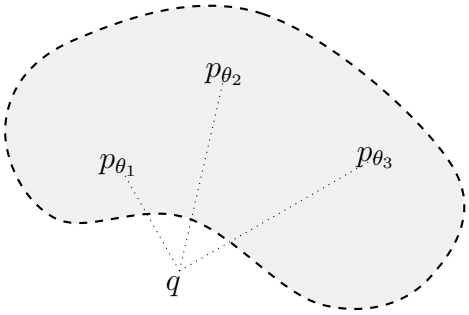

Figure 1: The universal model q is never too far away from any of the distributions p_θ in the class $\mathcal{P}$.

Since there can be many possible universal models we are left with the choice of which one to use. The minimax regret problem is to find the universal code q that minimizes the worst-case regret

$$\min_q \max_{x^n \in \mathcal{X}^n} \text{regret}(q, x^n),$$

where the minimum is taken over all possible probability mass functions. Such minimax methods have been shown to be robust with respect to different data generating mechanisms [11, 21].

Alternatively, we can seek for the universal code q that minimizes the expected worst-case regret or redundancy

$$\min_q \max_{p_n} \mathbb{E}_{X^n \sim p_n} \left[\text{regret}(q, X^n)\right],$$

which is equivalent to minimizing the worst-case KL divergence

$$\min_q \max_{p_n} D_{KL}(p_n \parallel q).$$

It is clear from the definition that the worst-case redundancy is always lower than the worst-case regret. In this thesis we mainly consider the worst-case regret as it is a stronger guarantee of performance — the performance is almost optimal for all possible cases and not just on average.

Finally, we say that the procedure q is asymptotically minimax if

$$\max_{x^n} \text{regret}(q, x^n) = \min_p \max_{x^n} \text{regret}(p, x^n) + o(1),$$

that is, the worst-case regret of q converges to the minimax value when $n \to \infty$. Watanabe and Roos proved [38] that for the class of memoryless sources, no asymptotically minimax strategy can be horizonless. Here horizonless means that the strategy does not depend on the length of the sequence.

2.4 Normalized maximum likelihood

Shtarkov proved [31] that the distribution that achieves minimax regret is the normalized maximum likelihood (NML) distribution

$$p_{\mathrm{NML}}(x^n) = \frac{p_{\hat{\theta}}(x^n)}{C},$$

where

$$C = \sum_{x^n \in \mathcal{X}^n} p_{\hat{\theta}}(x^n)$$

is a normalizing constant known as the Shtarkov sum. The sum is replaced by the corresponding integral in the continuous case. It was later proven [28] by Rissanen that the NML distribution also minimizes the expected regret.

A prohibiting factor in using the NML distribution is that calculating the normalizing constant involves summing over an exponential amount of possible sequences. However, for certain models the calculation can be done efficiently. In particular, Kontkanen and Myllymäki showed [16] that for discrete memoryless sources, calculating the regret can be done in linear time. They showed that if we denote the Shtarkov sum of this class with sample size n and alphabet size m as C_n^m, then C_n^m satisfies the recursive formula

$$C_n^m = C_n^{m-1} + \frac{n}{m-2} C_n^{m-2}.$$

The base cases C_n^1 and C_n^2 can be calculated in $\mathcal{O}(n)$ time and thus the Shtarkov sum C_n^m can be calculated in time $\mathcal{O}(n+m)$.

The asymptotic growth of the quantity C_n^m was studied by Orlitsky and Santhanam [24] in different asymptotic settings:

i) $m = o(n)$:

$$\log C_n^m \sim \frac{m-1}{2} \log \frac{n}{m}$$

ii) $m = \Theta(n)$:

$$\log C_n^m = \Theta(n)$$

iii) $n = o(m)$:

$$\log C_n^m \sim n \log \frac{m}{n}.$$

More precise asymptotics were studied later by Szpankowski and Weinberger [36]. From the definition of NML it follows that

$$\mathrm{regret}(p_{\mathrm{NML}}, x^n) = \log_2 C_n^m.$$

In particular, the value of the regret does not depend on the sequence x^n and as the regret grows at a logarithmic rate, the per-symbol regret diminishes to zero as n approaches infinity making NML a universal code.

Even though the NML distribution can be calculated in linear time, its use with arithmetic coding and other applications where sequential predictions are needed is problematic as computing the conditional probabilities takes exponential amount of time. In particular, computing all the conditional probabilities $p_{\mathrm{NML}}(x_{n+1}|x^n)$ up to n takes $\mathcal{O}(m^n)$ time.

2.5 Model selection

The NML distribution has been successfully used for model selection with the minimum description length (MDL) principle [12], a modern formalization of Occam's razor. It asserts that given a choice of different models, one should choose the model that yields the shortest description of the data while also taking into account the complexity of the model.

In model selection, given a choice of models, that is, a set of probability distributions, we have to find the model that best fits the data. This has to be done by balancing goodness-to-fit such as to prevent overfitting. For example, given a choice of different order polynomials we have to choose the order of the polynomial that fits the data well. The higher degree polynomials naturally fit the data better but do not generalize as well.

The old-style MDL principle chooses the model and its parameter such that they minimize the combined code-length of the model, the parameter θ and the data encoded using the parameter θ:

$$\hat{M}_{\mathrm{MDL}} = \arg\min_{\theta,M} \ell(M) + \ell(\theta) + \log_2 \frac{1}{p_\theta(x^n)}$$

This criterion naturally balances goodness-to-fit, as more complex models often yield shorter descriptions of the data but an increase in the number parameters requires longer description length for the parameters.

Modern versions of the MDL principle state that given a choice of different model classes, one should choose the model class M that yields the shortest combined description length of the model class and the data x^n encoded using a universal code for the model class:

$$\hat{M}_{\mathrm{MDL}} = \arg\min_{M} \ell(M) + \ell(x^n; M).$$

For example, using the NML distribution gives the code-length

$$\ell_{\mathrm{NML}}(x^n; M) = \log_2 \frac{1}{p(x^n|\hat{\theta}(x^n), M)} + \log_2 \sum_{x^n} p(x^n|\hat{\theta}(x^n), M).$$

The MDL principle has been applied to a wide variety of problems including linear regression [35] and image denoising [27]. The NML distribution has found use with MDL in for example histogram density estimation [17] and Bayesian network structure learning [34, 32].

3 Mixture codes

In this section we define the Bayes mixture for the multinomial model and review previous work on examining minimax optimal hyperparameters for the mixture. Furthermore, we develop an algorithm for computing the optimal hyperparameter for the multinomial Bayes mixture with given precision in logarithmic time. This makes calculating the optimal hyperparameter feasible for any reasonable parameter values.

3.1 Mixture codes for memoryless sources

Given a class $\mathcal{P}$ of distributions parameterized by some parameter set Θ, if W is a distribution on Θ, we can construct a new distribution p_{Bayes} by taking a weighted mixture over the distributions in $\mathcal{P}$:

$$p_{\text{Bayes}}(x^n) = \sum_{\theta \in \Theta} p(x^n|\theta) W(\theta),$$

where the sum is replaced by an integral in the continuous case. This is called a Bayes mixture or the Bayesian marginal likelihood [5].

Such Bayes mixtures can be shown to be universal codes [12, p. 176]. In certain exponential families Bayes mixtures are asymptotically minimax for both the worst-case and the expected regret [37]. Even exact representation of the NML distribution is possible with Bayes mixtures using signed mixtures [3], but requires high computational complexity.

The corresponding conjugate prior for the class of discrete memoryless sources is the Dirichlet distribution. In the symmetric case it takes the form

$$q(\theta|\alpha) = \frac{\Gamma(m\alpha)}{\Gamma(\alpha)^m} \prod_{j=1}^{m} \theta_j^{\alpha-1},$$

where $\alpha > 0$ is a hyperparameter and m is the alphabet size. We now get the probabilities for the sequences x^n by taking the weighted mixture with respect to the Dirichlet prior by integrating over the parameter space:

$$p_{B,\alpha}(x^n) = \int_\Theta \prod_{i=1}^{n} p(x_i|\theta) q(\theta|\alpha)\, d\theta = \frac{\Gamma(m\alpha)}{\Gamma(\alpha)^m} \frac{\prod_{j=1}^{m} \Gamma(n_j + \alpha)}{\Gamma(n + m\alpha)},$$

where n_j is the number of occurrences of symbol j in the sequence x^n. The sequential predictions can now easily be calculated as

$$p_{B,\alpha}(x_{n+1}|x^n) = \frac{p_{B,\alpha}(x^{n+1})}{p_{B,\alpha}(x^n)} = \frac{k + \alpha}{n + m\alpha},$$

where k is the number of times the symbol x_{n+1} occurs in x^n. This means that all predictive probabilities up to n can be calculated in time $\mathcal{O}(nm)$.

The choice of α dictates the amount of prior mass given to each of the symbols. For example, assuming that $\alpha = 1$, the prior is the $\mathrm{Dir}(1, \ldots, 1)$ distribution, which is the uniform distribution. Using this as the prior gives the Laplace estimator, for which the sequential predictions take the form $(k+1)/(n+m)$. Plugging in $m = 2$ gives Laplace's rule of succession.

The regret of the Laplace estimator is largest for sequences containing very few ones or zeros. Using the $\mathrm{Dir}(1/2, \ldots, 1/2)$ prior puts more mass to the boundaries of the probability simplex. This modifies the mixture by giving larger weight to the distributions that achieve short code-lengths on the critical sequences. The choice $\alpha = 1/2$ is called the Krichevsky-Trofimov estimator [18] and corresponds to the Jeffreys prior used in Bayesian statistics, where it is used as an uninformative prior. The Krichevsky-Trofimov estimator minimizes the expected regret asymptotically [41].

However, the Krichevsky-Trofimov estimator is not asymptotically minimax. In particular, Xie and Barron showed [42] that the regret of the Krichevsky-Trofimov estimator is higher than the minimax regret by a non-vanishing amount on the boundaries of the probability simplex. Xie and Barron modified the Krichevsky-Trofimov estimator to be asymptotically minimax by adding extra mass to the boundaries of the probability simplex:

$$q_{\mathrm{MJ}}^{(n)}(\theta) = \frac{\varepsilon_n}{2}\left\{\delta\left(\theta - \frac{1}{n}\right) + \delta\left(\theta - 1 + \frac{1}{n}\right)\right\} + (1 - \varepsilon_n) b_{1/2}(\theta),$$

where δ is the Dirac delta function, $b_{1/2}$ is the density function of the $\mathrm{Beta}(1/2, 1/2)$ distribution and $\varepsilon_n = n^{-1/8}$ as recommended by Xie and Barron. Notably this procedure depends on the sequence length n as no horizonless procedure can be asymptotically minimax.

Roos and Watanabe proved [38] that a simpler Bayes procedure with the sequence length-dependent hyperparameter

$$\alpha_n = \frac{1}{2} - \frac{\log 2}{2}\frac{1}{\log n}$$

achieves asymptotic minimaxity. This strategy has lower computational complexity and it achieves smaller worst-case regret than the method of Xie and Barron [38]. Figure 1 shows the optimal α as a function of n when $m = 2$ along with the asymptotic formula α_n given by Watanabe and Roos. Notably the asymptotic formula converges to the optimal α as the sequence length n increases. Thus both the optimal and the asymptotic prior converge to the asymptotic value $1/2$ at a logarithmic rate.

Example 2 (The sunrise problem)**.** *Consider a situation where the sun has risen ten times this year. The Krichevsky-Trofimov estimator gives probability* 95.5% *for the sun rising tomorrow, while Laplace's rule of succession gives* 91.7%*. For a horizon of 11 days, the asymptotic formula of Watanabe and Roos gives the probability* 96.7%*, and* 96.0% *for a horizon of 365 days.*

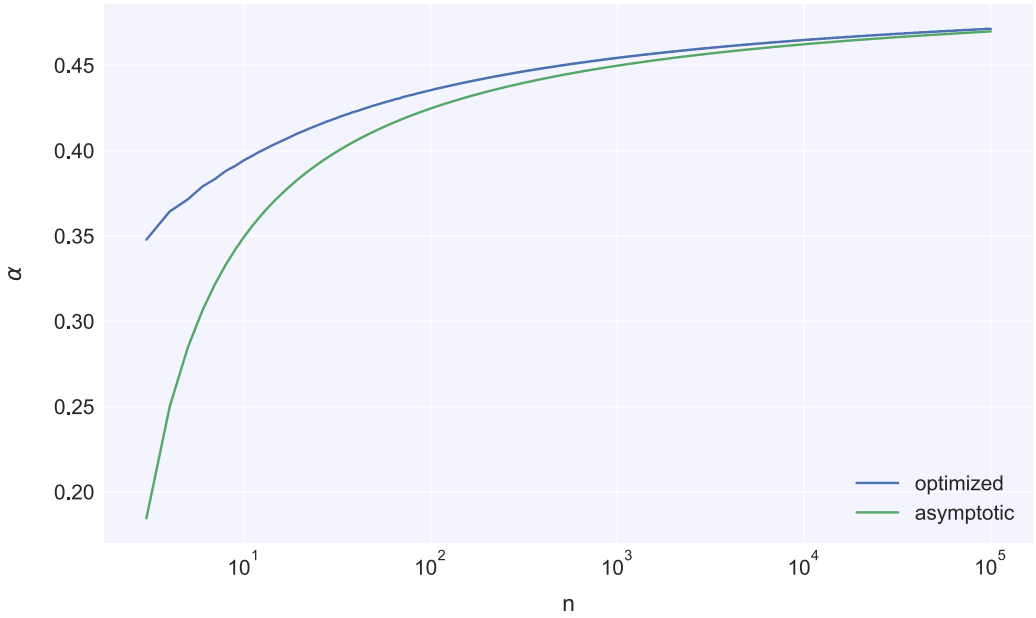

Figure 2: Optimal α and α_n given by the asymptotic formula of Watanabe and Roos as a function of n when $m = 2$.

3.2 Computation of the optimal hyperparameter

The formula given by Watanabe and Roos is asymptotic and thus it differs from the optimal α especially for small sample sizes as can be seen in Figure 2. Furthermore, the differences become greater as m increases as will be shown in Section 5. Even small differences in the hyperparameter α can be crucial in for example Bayesian network structure learning [33].

More specifically, if we consider the Bayes mixtures $p_{B,\alpha}$ defined previously, we would like to find $\alpha > 0$ such that

$$\max_{x^n} \operatorname{regret}(p_{B,\alpha}, x^n) = \max_{x^n} \log \frac{p(x^n|\hat{\theta}(x^n))}{p_{B,\alpha}(x^n)}$$

is minimized. Since there are an exponential number of possible sequences x^n, computing the optimal α at ε precision (e.g. $\varepsilon = 10^{-3}$) by considering all possible sequences takes exponential time.

In this section we present an algorithm that can compute the optimal α with ε precision in $\mathcal{O}(\log(\min\{n, m\}/\varepsilon))$ time where n is the sample size and m is the alphabet size. This algorithm makes it practical to calculate the optimal α fast for any typical values of n and m. The first step is the following lemma proved by Watanabe and Roos [38] which narrows down the number of possible worst-case sequences considerably:

Lemma 3. *The possible worst-case sequences x^n in*

$$\max_{x^n} \log \frac{p(x^n|\hat{\theta}(x^n))}{p_{B,\alpha}(x^n)}$$

have l non-zero counts ($l = 1, 2, \ldots, m$), each of which is $\lfloor \frac{n}{l} \rfloor$ or $\lfloor \frac{n}{l} \rfloor + 1$ and all the other counts are zeros.

Thus there are only $\mathcal{O}(\min\{n, m\})$ possible worst-case sequences. Since the count vector of each possible worst-case sequence contains at most two different elements, we can evaluate the regret in constant time. Thus we can find the optimal α with ε precision in time $\mathcal{O}(\min\{n, m\}/\varepsilon)$ by considering all α on a grid with length ε intervals. This can be further reduced to $\mathcal{O}(\min\{n, m\} \log(1/\varepsilon))$ by using the following lemma:

Lemma 4. *The function*

$$\max_{x^n} \log \frac{p(x^n|\hat{\theta}(x^n))}{p_{B,\alpha}(x^n)}$$

is unimodal as a function of α on the interval $(0, \infty)$.

Proof. We first consider the regret for a fixed x^n. Taking the derivative with respect to α, we obtain

$$\frac{\partial}{\partial \alpha} \log \frac{p(x^n|\hat{\theta}(x^n))}{p_{B,\alpha}(x^n)} = -\frac{\partial}{\partial \alpha} \log p_{B,\alpha}(x^n).$$

Levin and Reeds proved [20] that the derivative of $\log p_{B,\alpha}$ with respect to α has at most one zero on the interval $(0, \infty)$ and if this happens at a finite α, the corresponding zero has to be a local maximum. Therefore the regret for any x^n as a function of α is either decreasing, increasing or decreases up to a point and increases from that point on.

All monotone functions and functions that decrease up to a point and increase from that point on are strictly quasiconvex. Furthermore, the maximum of strictly quasiconvex functions is strictly quasiconvex. Therefore as a function of α the worst-case regret is strictly quasiconvex. Since a strictly quasiconvex function is strictly unimodal, the claim follows. □

Since the regret function is unimodal, we can optimize it with an algorithm such as golden section search [14] in time $\mathcal{O}(\log(1/\varepsilon))$ on the fixed-length interval $(0, 1/2)$. However, as there are $\mathcal{O}(\min\{n, m\})$ possible worst-case sequences and the regret for each of them can be evaluated in constant time, each evaluation of the function to be optimized by golden section search takes time $\mathcal{O}(\min\{n, m\})$. Therefore the optimal hyperparameter α can be found in time $\mathcal{O}(\min\{n, m\} \log(1/\varepsilon))$ with ε precision.

We now describe a way to evaluate the search for the worst-case sequence in time $\mathcal{O}(\log(\min\{n,m\}))$, yielding an $\mathcal{O}(\log(\min\{n,m\}/\varepsilon))$ algorithm. Here we assume that memory and variables can be accessed in constant time, and arithmetic operations, logarithms and the gamma function can also be evaluated in constant time. We first note that the regret of a sequence x^n can be written as

$$\mathrm{regret}(p_{B,\alpha}, x^n) = \sum_{j=1}^{m} \{n_j \log n_j - \log\Gamma(n_j+\alpha)\} + \kappa,$$

where κ is a quantity that does not depend on the count vector $(n_1,\ldots,n_m)$. Here n_j is the number of times the symbol x_j appears in the sequence x^n. For convenience, we allow the counts to be any non-negative real numbers as long as $\sum_{j=1}^m n_j = n$. We also denote any sequence with a count vector consisting of a symbols with count x, b symbols with count $x+1$ and the remaining $m-b-a$ symbols with count zero as $x^n_{a,b}$.

Consider now the following function, which represents the difference in regret by replacing x/y y counts in a count vector with a single x count:

$$h_\alpha(x,y) = x\log\frac{x}{y} - \log\Gamma(x+\alpha) + \frac{x}{y}\log\Gamma(y+\alpha) + (1-\frac{x}{y})\log\Gamma(\alpha).$$

This function has the following properties which can be verified by straightforward calculations:

- $h_\alpha(x,x) = 0$
- $h_\alpha(x,y) = -\frac{x}{y}h_\alpha(y,x)$
- $ah_\alpha(x,y) + bh_\alpha(x+1,y) = \mathrm{regret}(p_{B,\alpha}, x^n_{a,b}) - \mathrm{regret}(p_{B,\alpha}, y^n_{c,0})$,

where $c = n/y$. A key observation is described in the following lemma:

Lemma 5. *The second derivative*

$$\frac{\partial^2}{\partial x^2} h_\alpha(x,y)$$

has at most one zero.

Proof. Taking the second derivative, we have

$$\frac{\partial^2}{\partial x^2} h_\alpha(x,y) = \frac{1}{x} - \psi^{(1)}(x+\alpha).$$

Using the inequality $(\psi^{(1)}(x))^2 + \psi^{(2)}(x) > 0$ given by Batir [4], we can prove that the second derivative has at most one zero since the derivative

$$\frac{d}{dx}\left(x - \frac{1}{\psi^{(1)}(x+\alpha)}\right) = \frac{\psi^{(2)}(x+\alpha)}{\psi^{(1)}(x+\alpha)^2} + 1$$

is positive, meaning that the function $x - 1/\psi^{(1)}(x+\alpha)$ is increasing from $-1/\psi^{(1)}(\alpha) < 0$ and thus has at most one zero coinciding with the zero of the second derivative $\frac{\partial^2}{\partial x^2} h_\alpha(x,y)$. □

Since $\lim_{x\to 0^+} \frac{\partial^2}{\partial x^2} h_\alpha(x,y) = \infty$, we have that the function $x \mapsto h_\alpha(x,y)$ is either convex on the whole interval or convex up to an inflection point and concave from that point on. Suppose now that for a given α, we would like to find an integer y such that $h_\alpha(x,y) \leq 0$ for all other integers x. Intuitively this means that swapping a count $n_j = y$ in the count vector to any other counts such that their sum is y would lower the regret.

Since the point at which the function $x \mapsto h_\alpha(x,y)$ switches from convex to concave (if such a point exists) is not dependent on y, we can separately find y_1 such that $h_\alpha(x,y_1) \leq 0$ for all integers x on the convex interval and y_2 such that $h_\alpha(x,y_2) \leq 0$ for all integers x on the concave interval. We consider first finding such integer y_2. The following lemma states that if y_2 is the smallest integer on the concave interval for which $h_\alpha(y_2+1,y_2) < 0$, then $h_\alpha(x,y_2) < 0$ for all other integers x on the concave interval:

Lemma 6. *Let c be such that $x \mapsto h_\alpha(x,y)$ is concave for all $x \geq c$.*

- *If there exists a smallest integer $c \leq z < n$ such that $h_\alpha(z+1,z) < 0$, then $h_\alpha(x+1,x) < 0$ for all $x \geq z+1$. Furthermore, $h_\alpha(x,z) < 0$ for all $c \leq x \leq z-1$ and $z+1 \leq x \leq n$.*
- *If such integer z does not exist, $h_\alpha(x,n) < 0$ for all $c \leq x \leq n-1$.*

Proof. Assume first that there exists a smallest integer z such that $z \geq c$ and $h_\alpha(z+1,z) < 0$. Now let $x \geq z+1$. Since $x \mapsto h_\alpha(x,y)$ is concave and $h_\alpha(z,z) = 0$ as well as $h_\alpha(z+1,z) < 0$, we must have $h_\alpha(x,z) < 0$ and therefore $h_\alpha(z,x) > 0$. Since also $h_\alpha(x,x) = 0$, by concavity of the function $x \mapsto h_\alpha(x,y)$ we must have $h_\alpha(x+1,x) < 0$.

Since $h_\alpha(z,z) = 0$ and $h_\alpha(z+1,z) < 0$, by concavity $h_\alpha(x,z) < 0$ for all $x \geq z+1$. Now assume $h_\alpha(z-1,z) > 0$. Thus $h_\alpha(z,z-1) < 0$ which is a contradiction since z is the smallest integer such that $h_\alpha(z+1,z) < 0$. Thus by concavity $h_\alpha(x,z) < 0$ for all $x \leq z-1$.

For the final part, we have that $h_\alpha(n,n-1) > 0$ and thus $h_\alpha(n-1,n) < 0$. By the fact that $h_\alpha(n,n) = 0$ and the concavity of the function $t \mapsto h_\alpha(t,n)$, we have $h_\alpha(x,n) < 0$ for all $c \leq x \leq n-1$. □

Note that from the above lemma we also get that if such y_2 does not exist, then $h_\alpha(x,n) < 0$ for all integers x on the concave interval. Furthermore, as Lemma 6 states that if y_2 is the smallest integer for which $h_\alpha(y_2+1,y_2) < 0$, then $h_\alpha(x+1,x) < 0$ for all $x \geq y_2+1$, we can find y_2 in time $\mathcal{O}(\log n)$ by a binary search like routine (Algorithm 1). This algorithm returns the first integer z on the range `[start, end]` such that $f(z)$ is true, assuming that $f($`end`$)$ is true and $f(y)$ is true also for all $y > z$.

The following lemma verifies that if y_2 is such that $h_\alpha(x,y_2) < 0$ for all other integers on the concave interval, then the regret of sequences with counts vector consisting of only integers that are in a sense near to y_2 achieve higher regret compared to all other sequences:

Algorithm 1

```
function BIN(f, start, end)
    z, b ← start − 1, end
    while b ≥ 1 do
        while not f(z + b) and z + b ≤ end do
            z ← z + b
        end while
        b ← ⌊b/2⌋
    end while
    return z + 1
end function
```

Lemma 7. *Let k and c be such that $t \mapsto h_\alpha(t,k)$ is concave for all $t \geq c$ and $x, z \in \mathbb{N}, y \in \mathbb{R}$ be such that $c \leq x < x+1 \leq y < z < k$ and $h_\alpha(t,k) < 0$ for all $c \leq t \leq k-1$ and there exist sequences $x^n_{a,b}, y^n_{c,0}, z^n_{d,e}$. Then*

$$\text{regret}(p_{B,\alpha}, x^n_{a,b}) < \text{regret}(p_{B,\alpha}, z^n_{d,e}).$$

Now let $c \leq k \leq z < z+1 \leq y < x$ and $h_\alpha(t,k) < 0$ for all $t \geq k+1$. Then

$$\text{regret}(p_{B,\alpha}, x^n_{a,b}) < \text{regret}(p_{B,\alpha}, z^n_{d,e}).$$

Proof. Since $h_\alpha(z,k) < 0$, we have $h_\alpha(k,z) > 0$. Using the fact that $h_\alpha(z,z) = 0$, we have $h_\alpha(y,z) < 0$ by concavity of the function $t \mapsto h_\alpha(t,k)$. Thus we have $h_\alpha(z,y) > 0$. Now also $h_\alpha(z+1,y) > 0$ as otherwise $h_\alpha(k,y) < 0$ and thus $h(y,k) > 0$, which is a contradiction. Thus

$$\text{regret}(p_{B,\alpha}, z^n_{d,e}) - \text{regret}(p_{B,\alpha}, y^n_{c,0}) = dh_\alpha(z,y) + eh_\alpha(z+1,y) > 0.$$

Again, by concavity we have $h_\alpha(x,y) < 0$ and $h_\alpha(x+1,y) \leq 0$. Therefore

$$\begin{aligned}\text{regret}(p_{B,\alpha}, x^n_{a,b}) - \text{regret}(p_{B,\alpha}, z^n_{d,e}) &< \text{regret}(p_{B,\alpha}, x^n_{a,b}) - \text{regret}(p_{B,\alpha}, y^n_{c,0}) \\ &= ah_\alpha(x,y) + bh_\alpha(x+1,y) < 0\end{aligned}$$

The other part follows from a similar argument. □

In particular, let there be the sequences $\ell_1{}^n_{a_1,b_1}, \ell_2{}^n_{a_2,b_2}, \ldots, \ell_t{}^n_{a_t,b_t}$, where $\ell_1 < \ell_2 < \cdots < \ell_t$. Furthermore, let ℓ_c be the largest ℓ_i such that $x \mapsto h_\alpha(x,t)$ is convex for all $x \leq \ell_c$ and ℓ_d be the smallest ℓ_i such that $\ell_d \geq y_2 \geq c$. Then the above lemma states that for the sequences with $\ell_i > \ell_c$, the highest regret is achieved by a sequence corresponding to $\ell_{d-2}, \ell_{d-1}, \ell_d$ or ℓ_{d+1}.

The following lemmas verify that in the convex region the highest regret is at the boundaries of the interval. That is, the highest regret is achieved by a sequence corresponding to $\ell_1, \ell_2, \ell_{c-1}$ or ℓ_c. The proofs of the lemmas are the same as for Lemma 6 and Lemma 7, except using the fact that if $x \mapsto h_\alpha(x,k)$ is convex, then $x \mapsto -h_\alpha(x,k)$ is concave:

Lemma 8. *Let c be such that $x \mapsto h_\alpha(x, y)$ is convex for all $x \leq c$.*

- *If there exists a smallest integer $1 \leq z \leq c$ such that $h_\alpha(z+1, z) > 0$, then $h_\alpha(x+1, x) > 0$ for all $x \geq z+1$. Furthermore, $h_\alpha(y, z) > 0$ for all $z+1 \leq y \leq c$ and $1 \leq y \leq z-1$.*
- *If such integer does not exist, $h_\alpha(x, c) > 0$ for all $1 \leq x+1 \leq c$.*

Lemma 9. *Let c be such that $t \mapsto h_\alpha(t, k)$ is convex for all $t \leq c$ and $x, z \in \mathbb{N}, y \in \mathbb{R}$ be such that $x+1 \leq y < z < k \leq c$ and $h_\alpha(t, k) > 0$ for all $t+1 \leq k \leq c$ and there exist sequences $x^n_{a,b}, y^n_{c,0}, z^n_{d,e}$. Then*

$$\operatorname{regret}(p_{B,\alpha}, x^n_{a,b}) > \operatorname{regret}(p_{B,\alpha}, z^n_{d,e}).$$

Now let $k \leq z < z+1 \leq y < x \leq c$ and $h_\alpha(t, k) > 0$ for all $k+1 \leq t \leq c$. Then

$$\operatorname{regret}(p_{B,\alpha}, x^n_{a,b}) > \operatorname{regret}(p_{B,\alpha}, z^n_{d,e}).$$

These lemmas lead to Algorithm 3. Given n, m and α, the function `F` finds the maximum possible regret. On line 16, the algorithm uses the `BIN` routine to find the smallest integer x such that the function $x \mapsto h_\alpha(x, y)$ is concave. On the next line, the algorithm uses `BIN` to find the smallest integer x in the concave region such that $h(x+1, x)$ is negative. The result is saved into the variable y, matching y_2 in the previous discussion. On line 18, the variable q is set as the smallest ℓ such that $\ell \geq y$ and there exists a sequence $\ell^n_{a,b}$ for some $a, b \in \mathbb{N}_0$, matching ℓ_d in the previous discussion.

The functions `PREV` and `NEXT` find for a given x the previous (next) y such that there exists a sequence $y^n_{a,b}$ for some $a, b \in \mathbb{N}_0$. Using these functions, the maximum regret can be found on lines 19–24 by considering all the possible cases mentioned previously. The regret for each case is calculated in constant time by Algorithm 2. Since there can be multiple count vectors that consist of the integers k and $k+1$, the sign of $h_\alpha(k, k+1)$ is calculated on line 2. It is easy to verify that if $h_\alpha(k, k+1) < 0$, then the count vector with the maximum amount of k counts should be preferred, and the count vector with the maximum amount of $k+1$ counts otherwise.

The `MINIMAX` function on line 27 uses golden section search to minimize the maximum regret $\texttt{F}(n, m, \alpha)$ on the fixed-length interval $(0, 1/2]$. Since the `BIN` routine works in time $\mathcal{O}(\log n)$ and all other operations are constant time operations, this yields a $\mathcal{O}(\log(n/\varepsilon))$ time algorithm. However, both of the binary searches can also be performed by considering only the numbers $\lfloor n/m \rfloor, \lfloor n/(m-1) \rfloor, \ldots, n$ as possible inputs for the function f, which takes $\mathcal{O}(\log m)$ time. Thus the total time complexity is $\mathcal{O}(\log(\min\{n, m\}/\varepsilon))$. For example, on a computer with a 2.3 GHz Intel Core i5-7360U processor, a C implementation of the algorithm can return the optimal α for $n = m = 2^{60}$ in 4 ms at $\varepsilon = 10^{-3}$ precision and in 7 ms at $\varepsilon = 10^{-9}$ precision.

Algorithm 2

function $\textsc{Regret}(n, m, k, \alpha)$
  if $h_\alpha(k, k+1) > 0$ **then**
    $y, x \leftarrow n - k\lfloor n/k \rfloor, (n - y(k+1))/k$
  else
    $x, y \leftarrow (k+1)\lceil n/(k+1) \rceil - n, (n - xk)/(k+1)$
  end if
  $l \leftarrow xk \log k + y(k+1) \log (k+1) - n \log n$
  $b_1 \leftarrow \log \Gamma(m\alpha) - m \log \Gamma(\alpha) - \log \Gamma(n + m\alpha)$
  $b_2 = x \log \Gamma(k + \alpha) + y \log \Gamma(k + 1 + \alpha) + (m - x - y) \log \Gamma(\alpha)$
  return $l - b_1 - b_2$
end function

Algorithm 3

function $\textsc{f}(n, m, \alpha)$
  function $\textsc{prev}(k)$
    if $k \leq 1$ **then**
      return 0
    end if
    $r \leftarrow \lfloor n/k \rfloor + 1$
    return $\lfloor n/r \rfloor$
  end function
  function $\textsc{next}(k)$
    if $k \geq n$ **then**
      return n
    end if
    $r \leftarrow \lceil (n - k)/(k+1) \rceil$
    return $\lfloor n/r \rfloor$
  end function
  $p \leftarrow \textsc{bin}(f(x) := 1/x - \psi^{(1)}(x + \alpha) < 0, 1, n)$
  $y \leftarrow \min(n, \max(\lfloor n/m \rfloor, \textsc{bin}(f(x) := h_\alpha(x+1, x) < 0, p, n)))$
  $r, q, u \leftarrow \lfloor n/y \rfloor, \lfloor n/r \rfloor, -1$
  for $t \in \{q, \textsc{next}(q), \textsc{prev}(q), \textsc{prev}(\textsc{prev}(q)),$
      $\textsc{prev}(p), \textsc{prev}(\textsc{prev}(p)), \lfloor n/m \rfloor, \textsc{next}(\lfloor n/m \rfloor)\}$ **do**
    if $\max(1, \lfloor n/m \rfloor) \leq t \leq n$ **then**
      $u \leftarrow \max(u, \textsc{regret}(n, m, t, \alpha))$
    end if
  end for
  return u
end function
function $\textsc{minimax}(n, m, \varepsilon)$
  return $\textsc{gss}(f(\alpha) := \textsc{f}(n, m, \alpha), 0, \frac{1}{2}, \varepsilon)$
end function

4 Large alphabet coding

In this section we first review previous work on universal compression on large alphabets. For large alphabets in comparison to the sample size, we prove that the minimax optimal Bayes mixture is given by the hyperparameter $\alpha = 1/3$. We also prove asymptotic properties of the 1/3-mixture which show that the worst-case regret of the 1/3-mixture approaches that of the NML distribution as the alphabet size increases.

4.1 Related work

For data compression on very large or even infinite alphabets, universal compression of i.i.d. distributions has traditionally been avoided. This is due to a result proved originally by Kieffer [15], which states that as the alphabet size grows unbounded, so does the regret. This holds even when the length of the sequence or the block-length is allowed to grow with the alphabet size. Thus progressively larger sample sizes are needed to achieve a given target level of regret as the alphabet size increases.

Therefore, recent work on universal compression on large alphabets has mainly focused on algorithms such as Lempel-Ziv that avoid the problems of large alphabets by converting the alphabet into a smaller one [9], or universal compression on subclasses of i.i.d. distributions. Examples of such distributions include envelope classes [2, 6] and patterns [23, 26].

The pattern of a sequence represents the relative order in which its symbols appear. For example, the pattern of abracadabra is 12314151231. The probability of a pattern ψ is the sum of the probabilities of all sequences whose pattern is ψ. Acharya et al. prove [1] that for the class of all distributions over patterns of length n induced by all i.i.d. distributions, the worst-case regret is bounded by $n^{1/3}(\log n)^4$. Thus the average regret diminishes to zero as the sequence length increases regardless of the alphabet size.

However, as patterns target a different distribution, codes based on patterns are not directly interchangeable with codes for i.i.d. distributions. Thus for example in model selection it is still needed to calculate a target minimax distribution for i.i.d. sources. Furthermore, in situations where both the sequence length and alphabet size are known, asymptotic results hold less weight. Distributions for coding i.i.d. distributions can also be extended to models such as Markov sources that incorporate context [44].

Yang and Barron propose [45] a distribution for universal coding of i.i.d. distributions for all different setups of the sample size and the alphabet size. The distribution is based on Poisson sampling and tilting. Using Poisson sampling makes the counts independent. In addition, the length n is allowed to be variable, which considers a larger class of distributions. However, the distribution can also be used for fixed sample size coding and prediction by conditioning on the sample size.

The coding distribution proposed by Yang and Barron, called the tilted Stirling ratio distribution, is given by

$$p_{\text{Stirling}}(x^n) = p_{\text{uniform}}(x^n|n_1, \ldots, n_m)Q_a(n_1, \ldots, n_m),$$

where

$$Q_a(n_1, \ldots, n_m) = \prod_{i=1}^{m} P_a(n_m) = \prod_{i=1}^{m} \frac{n_i^{n_i} e^{-n_i}}{n_i!} \frac{e^{-an_i}}{C_a}.$$

Here C_a is a normalizer $\sum_{k=0}^{\infty} k^k e^{-(1+a)k}/k!$ and $a > 0$ is a tilting parameter whose optimal value depends only on the ratio m/n. The normalizer converges exponentially to zero and can in practice be computed by cutting the sum at k large compared to $1/a$. In practice the tilting parameter has to be optimized by considering a grid of possible parameter values.

This distribution is slightly suboptimal for coding sequences of fixed length n as the coding distribution assumes the sample size is not known. Thus the above distribution assigns a probability to all finite-length sequences. However, Yang and Barron prove that the worst-case regret is asymptotically near the minimax regret in both cases $n = o(m)$ and $m = o(n)$, although the worst-case regret does not converge to the minimax regret. Additionally, these results assume that the tilting parameter a has been optimized.

When conditioned on the sequence length n, the tilted Stirling ratio distribution allows exact computation of the NML distribution by

$$p_{\text{NML}}(x^n) = \frac{Q_a(n_1, \ldots, n_m)}{P_a^m(n)},$$

where

$$P_a^m(k) = \sum_{k'=0}^{k} P_a(k')P_a^{m-1}(k - k')$$

is the m-fold convolution of $P_a(n)$. This convolution can be calculated in time $\mathcal{O}(mn^2)$. Although this is inferior to the linear-time algorithm by Kontkanen and Myllymäki, the conditional distributions of the NML distribution can be calculated faster than in exponential time through this method. However, in practice even this time complexity is often infeasible.

Yang and Barron extend this distribution for coding Markov sources [44]. In a Markov source each symbol is generated according to a probability that depends on the symbol's context (a sequence of symbols preceding it). This allows taking dependencies between the symbols into account. These contexts can be modeled by a context tree that determines the set $\mathcal{S}$ of possible contexts. Yang and Barron describe a greedy algorithm for choosing the context set such as to minimize the worst-case coding cost

$$\log 1/Q(X|\mathcal{S}) + D(\mathcal{S}),$$

where $Q(X|\mathcal{S}) = \prod_{s \in \mathcal{S}} Q_{a_s}(n_{s1}, \ldots, n_{sm})$ is the probability of the data given the tree and $D(\mathcal{S})$ is the cost of coding the tree itself. Here n_{si} is the number of times the symbol i occurs such that its context is s.

4.2 Mixture codes for large alphabets

We now consider finding the minimax optimal Dirichlet prior for the Bayes mixture defined in Section 3 in a large alphabet setting. As the sample size n grows, we know that the optimal hyperparameter α converges to the asymptotic value $\alpha = 1/2$ when the alphabet size is fixed. However, in this section we prove that the minimax optimal hyperparameter is $\alpha = 1/3$ when the alphabet size is large compared to the sample size. The worst-case regret of $p_{B,1/3}$ turns out to approach the regret of the NML distribution when m increases, as we will see in Section 4.3.

We prove this result by showing that there is always a sequence whose regret is decreasing as a function of α and a sequence whose regret is increasing as a function of α when $m > \frac{5}{2} + \frac{4}{n-2} + \frac{3}{2}$. Furthermore, the regrets of these sequences intersect at the point $\alpha = 1/3$. Finally, as no other sequence has higher regret at this point, the point $\alpha = 1/3$ has to be the minimax point. This is illustrated in Figure 3 which shows the regrets of the possible worst-case sequences when $n = 10, m = 30$.

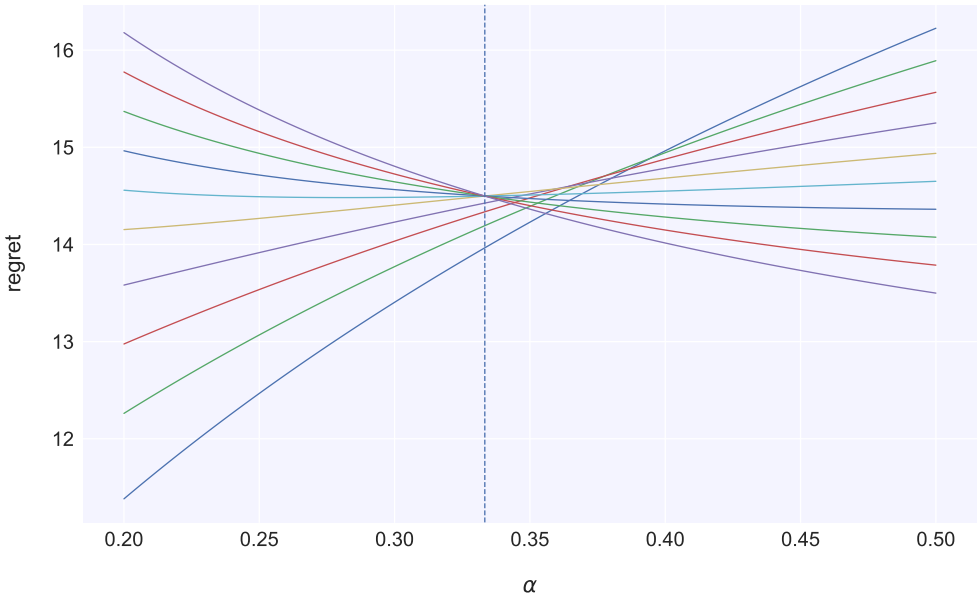

Figure 3: Regrets of the possible worst-case sequences as a function of α when $n = 10, m = 30$. A vertical line is shown at the minimax point $\alpha = \frac{1}{3}$.

We start by proving that when $m \geq n$ and $\alpha = 1/3$, all sequences where each symbol can occur at most twice have the same regret. This requires first proving the following lemma:

Lemma 10. *For $k \geq 2$, the function*

$$f(k) = k \log k - k \log 3 - \log \Gamma(k + \frac{1}{3}) + \log \Gamma(\frac{1}{3})$$

satisfies $f(k) \leq 0$, where equality holds if and only if $k = 2$.

Proof. The derivative of f is

$$f'(k) = \log \frac{k}{3} - \psi(k + \frac{1}{3}) + 1.$$

Using the inequality $\psi^{(1)}(k) > 1/k + 1/(2k^2)$ (e.g. [13]), we can show that if $k > 2/3$, we have

$$f''(k) = \frac{1}{k} - \psi^{(1)}(k + \frac{1}{3}) < 0.$$

Furthermore, since $f'(2) < 0$, the derivative f' must also be negative for all $k \geq 2$ and therefore f is decreasing. The claim holds since $f(2) = 0$. □

Lemma 11. *Let m, n be integers such that $m \geq n$ and x^n be a sequence where all the symbols are different. Then for all sequences y^n we have*

$$\text{regret}(p_{B,\frac{1}{3}}, x^n) \geq \text{regret}(p_{B,\frac{1}{3}}, y^n),$$

where equality holds if and only if no symbol in y^n occurs more than twice.

Proof. Let y^n be a sequence with at least one symbol occurring more than once. In particular, there is a symbol j that occurs $n_j > 1$ times. Let z^n be any sequence corresponding to the count vector of y^n except that n_j and $n_j - 1$ zeros have been replaced by n_j ones. Now, using Lemma 10 we have

$$\begin{aligned}
\text{regret}(p_{B,\frac{1}{3}}, y^n) &= \text{regret}(p_{B,\frac{1}{3}}, z^n) - n_j(1 \log 1 - \log \Gamma(1 + \frac{1}{3})) \\
&\quad - (n_j - 1) \log \Gamma(\frac{1}{3}) + n_j \log n_j - \log \Gamma(n_j + \frac{1}{3}) \\
&= \text{regret}(p_{B,\frac{1}{3}}, z^n) + n_j \log \frac{n_j}{3} + \log \frac{\Gamma(\frac{1}{3})}{\Gamma(n_j + \frac{1}{3})} \\
&\leq \text{regret}(p_{B,\frac{1}{3}}, z^n),
\end{aligned}$$

where equality holds if and only if $n_j = 2$. Since any count vector can be transformed into $(1, 1, \ldots, 0, 0, \ldots)$ by repeatedly replacing elements in the count vector by ones, the claim holds. □

Lemma 12. *Let m, n be integers such that $m \geq n$ and x^n be a sequence with all symbols occurring at most once. Then*

$$\text{regret}(p_{B,\alpha}, x^n)$$

is decreasing as a function of α.

Proof. The count vector corresponding to x^n is $(1, \ldots, 1, 0, \ldots, 0)$, where we have n ones and $m - n$ zeros. The regret of $p_{B,\alpha}$ is given by

$$\text{regret}(p_{B,\alpha}, x^n) = \log \frac{p(x^n|\hat{\theta}(x^n))}{p_{B,\alpha}(x^n)} = n \log (1/n) - \log p_{B,\alpha}(x^n).$$

Taking the derivative with respect to α, we obtain

$$\begin{aligned}\frac{\partial}{\partial \alpha}\text{regret}(p_{B,\alpha}, x^n) &= -\frac{\partial}{\partial \alpha} \log p_{B,\alpha}(x^n) \\ &= m(\psi(n + m\alpha) - \psi(m\alpha)) - \frac{n}{\alpha}.\end{aligned}$$

Repeated application of the identity $\psi(n+1) = \psi(n) + \frac{1}{n}$ gives

$$\begin{aligned}m(\psi(n + m\alpha) - \psi(m\alpha)) - \frac{n}{\alpha} &= m\left(\sum_{k=0}^{n-1} \frac{1}{m\alpha + k}\right) - \frac{n}{\alpha} \\ &= \sum_{k=0}^{n-1} \left(\frac{1}{\alpha + \frac{k}{m}} - \frac{1}{\alpha}\right) \leq 0\end{aligned}$$

for all positive α. Thus the regret corresponding to the sequence x^n is always a decreasing function of α. □

Lemma 13. *Let m, n be natural numbers such that $m \geq n$ and $0 < \alpha < 1$. Then*

$$m\psi(n + m\alpha) - m\psi(m\alpha) > \frac{1}{2\alpha} + \frac{1}{2(\alpha + 1)} + \frac{2m(n-1)}{2m\alpha + n - 1}.$$

Proof. We can first write

$$m\psi(n + m\alpha) - m\psi(m\alpha) = \sum_{k=0}^{n-1} \frac{1}{\alpha + \frac{k}{m}}.$$

Now applying the trapezoidal rule

$$\int_1^N f(x)\, dx < \sum_{k=1}^{N} f(k) - \frac{1}{2}(f(1) + f(N)),$$

where f is a convex function, we have

$$\begin{aligned}\sum_{k=0}^{n-1} \frac{1}{\alpha + \frac{k}{m}} &> \frac{1}{2}\left(\frac{1}{\alpha} + \frac{1}{\alpha + \frac{n-1}{m}}\right) + \int_0^{n-1} \frac{dx}{\alpha + \frac{x}{m}} \\ &> \frac{1}{2}\left(\frac{1}{\alpha} + \frac{1}{\alpha + 1}\right) + \int_0^{n-1} \frac{dx}{\alpha + \frac{x}{m}} \\ &= \frac{1}{2\alpha} + \frac{1}{2(\alpha + 1)} + m \log\left(1 + \frac{n-1}{m\alpha}\right).\end{aligned}$$

Using the inequality

$$\log(1+x) \geq \frac{2x}{2+x}$$

valid for $x \geq 0$, we get

$$\begin{aligned}\frac{1}{2\alpha} + \frac{1}{2(\alpha+1)} + m\log\left(1+\frac{n-1}{m\alpha}\right) &\geq \frac{1}{2\alpha} + \frac{1}{2(\alpha+1)} + \frac{2m\frac{n-1}{m\alpha}}{2+\frac{n-1}{m\alpha}} \\ &= \frac{1}{2\alpha} + \frac{1}{2(\alpha+1)} + \frac{2m(n-1)}{2m\alpha+n-1}.\end{aligned}$$

□

Lemma 14. *Let m, n be natural numbers such that $n > 2$ and*

$$m > \frac{5}{2}n + \frac{4}{n-2} + \frac{3}{2}.$$

Then the regret of a sequence where each symbol occurs twice (and one symbol occurs once if n is odd) is an increasing function of α when $\alpha \geq 1/3$.

Proof. Assume first that n is even and let x^n be a sequence with counts being $(2, \ldots, 2, 0, \ldots, 0)$, that is, there are $n/2$ twos and the rest $m - n/2$ counts are zeros. Taking the derivative we obtain

$$\begin{aligned}\frac{\partial}{\partial\alpha}\mathrm{regret}(p_{B,\alpha}, x^n) &= -\frac{\partial}{\partial\alpha}\log p_{B,\alpha}(x^n) \\ &= m\psi(n+m\alpha) - m\psi(m\alpha) + \frac{n}{2}(\psi(\alpha) - \psi(2+\alpha)).\end{aligned}$$

For this to be positive, from Lemma 13 we have the inequality

$$\frac{1}{2\alpha} + \frac{1}{2(\alpha+1)} + \frac{2m(n-1)}{2m\alpha+n-1} - n\left(\frac{1}{2\alpha} + \frac{1}{2(\alpha+1)}\right) > 0,$$

from which we can solve

$$m > \frac{(2\alpha+1)(n-1)}{2\alpha}$$

for $0 < \alpha < 1, n > 2$.

Now assume that n is odd and take x^n to be a sequence with counts $(2, \ldots, 2, 1, 0, \ldots, 0)$, that is, $(n-1)/2$ symbols occur twice and one symbol occurs once. Taking the derivative again gives

$$\begin{aligned}&\frac{\partial}{\partial\alpha}\mathrm{regret}(p_{B,\alpha}, x^n) \\ &\quad = m\psi(n+m\alpha) - m\psi(m\alpha) + \frac{n-1}{2}(\psi(\alpha) - \psi(2+\alpha)) - \frac{1}{\alpha}.\end{aligned}$$

Again, from Lemma 13 we get the inequality

$$\frac{1}{2\alpha} + \frac{1}{2(\alpha+1)} + \frac{2m(n-1)}{2m\alpha+n-1} - (n-1)\left(\frac{1}{2\alpha} + \frac{1}{2(\alpha+1)}\right) - \frac{1}{\alpha} > 0,$$

from which we can solve

$$m > \frac{(n-1)(2\alpha(n-1)+n)}{2\alpha(n-2)}$$

for $0 < a < 1, n > 2$. Since

$$\frac{(n-1)(2\alpha(n-1)+n)}{2\alpha(n-2)} > \frac{2\alpha(n-1)+n}{2\alpha} > \frac{(2\alpha+1)(n-1)}{2\alpha},$$

if m satisfies the bound for the odd case, the bound for the even case is also satisfied. Clearly this bound is decreasing as a function of α. Plugging $\alpha = 1/3$ into the bound, we have

$$m > \frac{5}{2}n + \frac{4}{n-2} + \frac{3}{2}.$$

Therefore if m satisfies the above bound, the derivative is positive for all $\alpha \geq 1/3$ and the regret is growing for all $\alpha \geq 1/3$. □

We are now ready to prove the main result which states that the Dirichlet prior which minimizes the worst-case regret is the $\mathrm{Dir}(1/3, \ldots, 1/3)$ distribution when $n > 2$ and $m > 5/2n + 4/(n-2) + 3/2$:

Theorem 15. *Let m, n be natural numbers such that $n > 2$ and*

$$m > \frac{5}{2}n + \frac{4}{n-2} + \frac{3}{2}.$$

Then

$$\arg\min_{\alpha} \max_{x^n} \mathrm{regret}(p_{B,\alpha}, x^n) = \frac{1}{3}.$$

Proof. We denote

$$f(\alpha) = \max_{x^n} \mathrm{regret}(p_{B,\alpha}, x^n).$$

From Lemma 11, we have $f(1/3) = \mathrm{regret}(p_{B,1/3}, x^n) = \mathrm{regret}(p_{B,1/3}, y^n)$, where x^n is a sequence where each symbol is different and y^n is a sequence where each symbol in the sequence occurs twice (and one symbol occurs once if n is odd). Using Lemma 12, we have $f(\alpha) \geq \mathrm{regret}(p_{B,\alpha}, x^n) > f(1/3)$ for all $\alpha < 1/3$. From Lemma 14, we know that $f(\alpha) \geq \mathrm{regret}(p_{B,\alpha}, y^n) > f(1/3)$ for all $\alpha > 1/3$ when $m > 5/2n + 4/(n-2) + 3/2$. Therefore the function f is minimized at the point $\alpha = 1/3$. □

We note that the above bound is quite tight, as can be seen in Figure 4. In particular, the bound given for m will converge to $\frac{5}{2}n + \frac{3}{2}$, while numerical results show that the optimal ratio of m/n converges to between 2.1 and 2.2.

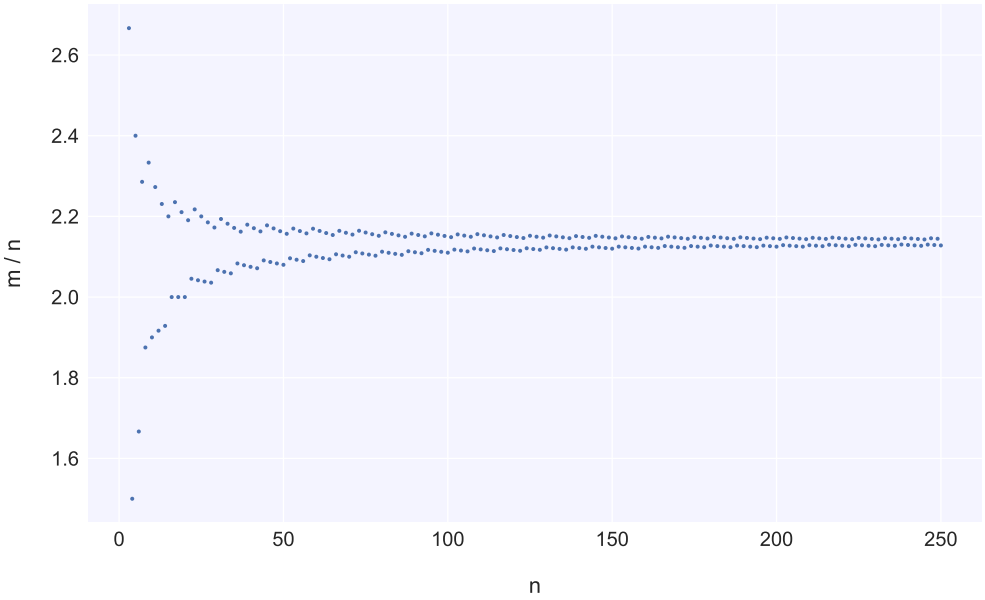

Figure 4: The ratio m^*/n as a function of n where m^* is the smallest m for which $\alpha = 1/3$ is optimal.

4.3 Asymptotic properties of the 1/3-mixture

The following theorem states that the worst-case regret of $p_{B,1/3}$ grows asymptotically at the same rate as the regret of the NML distribution when n grows asymptotically slower than m:

Theorem 16. *If $n = o(m)$, the worst-case regret of $p_{B,1/3}$ grows as*

$$n \log \frac{m}{n} + \frac{3}{2} \frac{n(n-1)}{m} + \mathcal{O}\left(\frac{n^3}{m^2}\right).$$

Proof. When $n = o(m)$, by definition there is a m_0 such that $m \geq n$ for all $m \geq m_0$. Then for $p_{B,1/3}$ the worst-case regret occurs when x^n is a sequence where all the symbols are different. Thus

$$\begin{aligned}
\max_{x^n} \operatorname{regret}(p_{B,\frac{1}{3}}, x^n) &= -\log \Gamma(\frac{m}{3}) - n \log n - n \log \Gamma(1 + \frac{1}{3}) \\
&\quad - (m-n) \log \Gamma(\frac{1}{3}) + m \log \Gamma(\frac{1}{3}) + \log \Gamma(n + \frac{m}{3}) \\
&= -\log \Gamma(\frac{m}{3}) - n \log n - n \log \frac{1}{3} + \log \Gamma(n + \frac{m}{3}).
\end{aligned}$$

Applying Stirling's approximation and using the identity

$$\log\left(\frac{m}{3}+n\right) = \log\frac{m}{3} + \log\left(1+\frac{3n}{m}\right),$$

we have

$$\max_{x^n} \operatorname{regret}(p_{B,\frac{1}{3}}, x^n) = \left(\frac{m}{3}+n-\frac{1}{2}\right)\log\left(1+\frac{3n}{m}\right) + n\log\frac{m}{n} - n + o(1).$$

Using the Taylor expansion $\log(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \cdots$ results in

$$\max_{x^n} \operatorname{regret}(p_{B,\frac{1}{3}}, x^n) = n\log\frac{m}{n} + \frac{3}{2}\frac{n(n-1)}{m} + \mathcal{O}\left(\frac{n^3}{m^2}\right).$$

□

We note that this result matches exactly the growth-rate of the NML regret in the case $n = o(m)$ as given by Szpankowski and Weinberger [36]. In particular, if n is fixed and m grows, the worst-case regret of the 1/3-mixture will converge to the regret of the NML distribution.

It also directly follows from Theorem 16 that the NML regret can be approximated in constant time by the worst-case regret

$$n\log\frac{3}{n} + \log\Gamma(n+\frac{m}{3}) - \log\Gamma(\frac{m}{3})$$

of the 1/3-mixture when m is large compared to n. This approximation is compared to the regret of the NML distribution in Table 1 for different values of n and m. In particular, the approximation converges to the exact value of the NML regret as m increases towards infinity.

Table 1: Regret values for various values of n and m.

n	m	approx	$\log C_n^m$
50	100	60.555	60.004
	1000	153.292	153.276
	10000	265.282	265.281
500	1000	609.691	603.928
	10000	1533.550	1533.379
	100000	2652.883	2652.881
5000	10000	6101.034	6043.158
	100000	15336.133	15334.406
	1000000	26528.893	26528.873

We can also prove that when m is fixed and n grows, the difference between the worst-case regret of the 1/3-mixture and the NML regret will converge to a constant. We use the following lemma which characterizes the worst-case sequences of $p_{B,1/3}$ when $n \geq 2m$:

Lemma 17. *If $n \geq 2m$, the worst-case sequences of $p_{B,1/3}$ have each element occurring n/m times (ignoring integer constraints).*

Proof. We first note that when ignoring the constraint of the counts being integers, the proof of Lemma 1 by Watanabe and Roos [38] shows that the worst-case sequences of $p_{B,\alpha}$ are maximally uniform, i.e. have l non-zero counts ($l = 1, 2, \ldots, m$), each of which is n/l.

Consider now the function $h_{\frac{1}{3}}(x, y)$ from Section 3. Taking the first and second order derivatives, we have

$$\frac{\partial}{\partial x} h_{\frac{1}{3}}(x, y) = \log \frac{x}{y} - \psi(x + \frac{1}{3}) + \frac{1}{y} \log \frac{\Gamma(y + \frac{1}{3})}{\Gamma(\frac{1}{3})} + 1 =: g(x, y)$$

$$\frac{\partial^2}{\partial x^2} h_{\frac{1}{3}}(x, y) = \frac{1}{x} - \psi^{(1)}(x + \frac{1}{3}).$$

As in the proof of Lemma 10, we have $1/x - \psi^{(1)}(x + \frac{1}{3}) < 0$ for $x > 2/3$ and thus $x \mapsto g(x, y)$ is decreasing for $x > 2/3$. Since we can numerically verify that $g(2, 2) < 0$, we have $g(x, 2) < 0$ for all $x > 2$ and thus also $x \mapsto h_{\frac{1}{3}}(x, y)$ is decreasing for $x > 2$. Since $h_{\frac{1}{3}}(2, 2) = 0$, we have $h_{\frac{1}{3}}(x, 2) < 0$ and $h_{\frac{1}{3}}(2, x) > 0$ for all $x > 2$. Now by concavity $h_{\frac{1}{3}}(y, x) < 0$ for all $y > x \geq 2$. Thus for all $y > x \geq n/m \geq 2$,

$$\mathrm{regret}(p_{B,\frac{1}{3}}, y^n_{a,0}) - \mathrm{regret}(p_{B,\frac{1}{3}}, x^n_{b,0}) = a h_{\frac{1}{3}}(y, x) < 0,$$

and the regret is maximized when x is minimized, that is, $x = n/m$. □

Theorem 18. *If $m = o(n)$, the worst-case regret of $p_{B,1/3}$ grows as*

$$\frac{m - 1}{2} \log \frac{n}{2\pi} + \log \frac{\Gamma(\frac{1}{3})^m}{\Gamma(\frac{m}{3}) m^{\frac{m}{6}}} + \mathcal{O}\left(\frac{m^2}{n}\right).$$

Proof. When $m = o(n)$, by definition there is a n_0 such that $n \geq 2m$ for all $n \geq n_0$. From Lemma 17, we know that the maximum regret is then achieved by a sequence where each element has count n/m. Thus we now have the maximum regret

$$-n \log m - \log \Gamma(\frac{m}{3}) + m \log \Gamma(\frac{1}{3}) + \log \Gamma(n + \frac{m}{3}) - m \log \Gamma(\frac{n}{m} + \frac{1}{3}).$$

Applying Stirling's approximation gives

$$\left(\frac{m}{6}-n\right)\log\left(\frac{1}{3}+\frac{n}{m}\right)+\left(\frac{m}{3}+n-\frac{1}{2}\right)\log\left(\frac{m}{3}+n\right)-n\log m \\ +\log\frac{\Gamma(\frac{1}{3})^m}{\Gamma(\frac{m}{3})}-\frac{m-1}{2}\log(2\pi)+o(1).$$

Using the identity

$$\log\left(\frac{1}{3}+\frac{n}{m}\right)=\log\frac{n}{m}+\log\left(1+\frac{m}{3n}\right)$$

and applying the Taylor expansion $\log(1+x)=x-\frac{x^2}{2}+\frac{x^3}{3}-\cdots$ we have

$$\left(\frac{m}{6}-n\right)\log\left(\frac{1}{3}+\frac{n}{m}\right)=-n\log\frac{n}{m}+\frac{m}{6}\left(\log\frac{n}{m}-2\right)+\mathcal{O}\left(\frac{m^2}{n}\right).$$

Similarly, using the identity

$$\log\left(\frac{m}{3}+n\right)=\log n+\log\left(1+\frac{m}{3n}\right)$$

and applying the Taylor expansion for $\log(1+x)$, we have

$$\left(\frac{m}{3}+n-\frac{1}{2}\right)\log\left(\frac{m}{3}+n\right)=n\log n+\frac{1}{6}((2m-3)\log n+2m)+\mathcal{O}\left(\frac{m^2}{n}\right).$$

Putting these together, the worst-case regret is

$$\frac{m-1}{2}\log\frac{n}{2\pi}+\log\frac{\Gamma(\frac{1}{3})^m}{\Gamma(\frac{m}{3})}-\frac{m}{6}\log m+\mathcal{O}\left(\frac{m^2}{n}\right).$$

□

Theorem 18 proves that when m is fixed and n increases towards infinity, the difference of the NML regret and the worst-case regret of $p_{B,1/3}$ will converge to within a constant that depends only on m. This can be seen from the asymptotic form of the NML regret as given by Xie and Barron [42]:

$$\frac{m-1}{2}\log\frac{n}{2\pi}+\log\frac{\Gamma(\frac{1}{2})^m}{\Gamma(\frac{m}{2})}+o(1).$$

In comparison, the worst-case regret of the Krichevsky-Trofimov estimator is asymptotically given by

$$\frac{m-1}{2}\log\frac{n}{\pi}+\log\frac{\Gamma(\frac{1}{2})^m}{\Gamma(\frac{m}{2})}+o(1).$$

5 Experiments

In this section we present numerical results evaluating the regret of the 1/3-mixture for different settings of the sample size and the alphabet size. We also evaluate the regrets of other distributions for comparison.

5.1 Regret as a function of the sample size

We compare the worst-case regrets of the tilted Stirling ratio distribution and the Bayes mixture with different Dirichlet hyperparameters: 1/2, 1/3, optimized and the asymptotic formula given by Watanabe and Roos. We first calculated the worst-case regrets when m is fixed and n increases. In Figure 5 the alphabet size is fixed as $m = 20$. The regret of the NML distribution is subtracted to make the comparison clearer. It can be seen that the worst-case regret of the tilted Stirling ratio distribution is higher than that of any of the Bayes mixtures. Naturally the optimized Bayes mixture always has the lowest worst-case regret amongst the Bayes mixtures.

It can be seen that the 1/3-mixture achieves lower worst-case regret than the Krichevsky-Trofimov ($\alpha = 1/2$) mixture. Neither is asymptotically minimax but both will converge to within a constant of the NML regret. As both the asymptotic formula and thus also the optimized Bayes mixture are asymptotically minimax, they will eventually converge to the NML regret. However, this convergence is quite slow as can be seen in Figure 5.

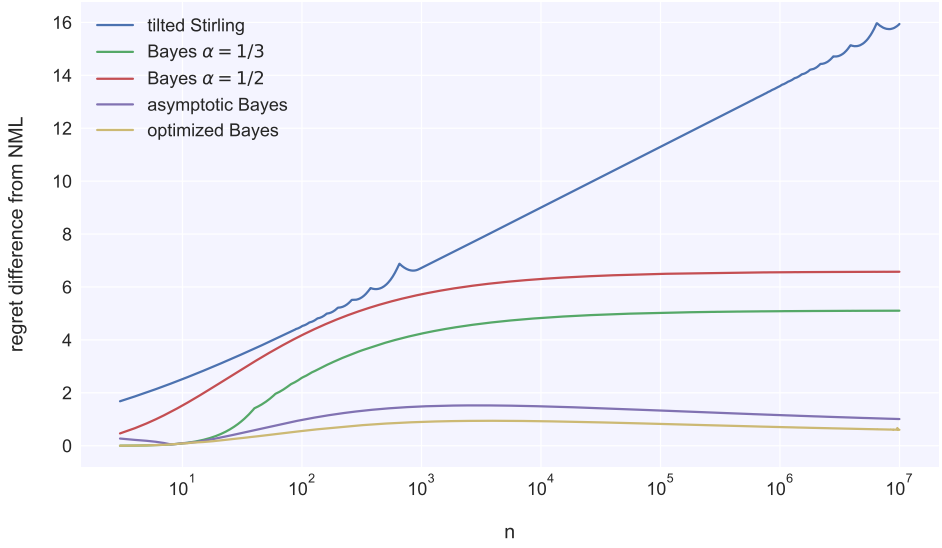

Figure 5: Worst-case regret differences from NML regret when $m = 20$.

When the alphabet size grows, the worst-case regret of the tilted Stirling ratio distribution gets closer to the NML regret. In particular, Figure 6 shows that when $m = 200$, the worst-case regrets of the tilted Stirling ratio distribution and the optimized Bayes mixture are comparable. The 1/3-mixture achieves lower worst-case regret when m is large compared to n.

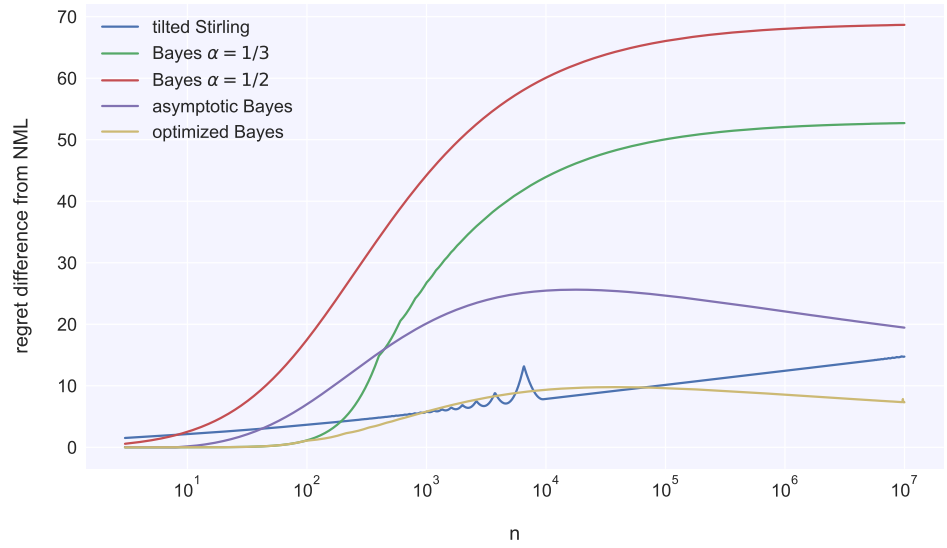

Figure 6: Worst-case regret differences from NML regret when $m = 200$.

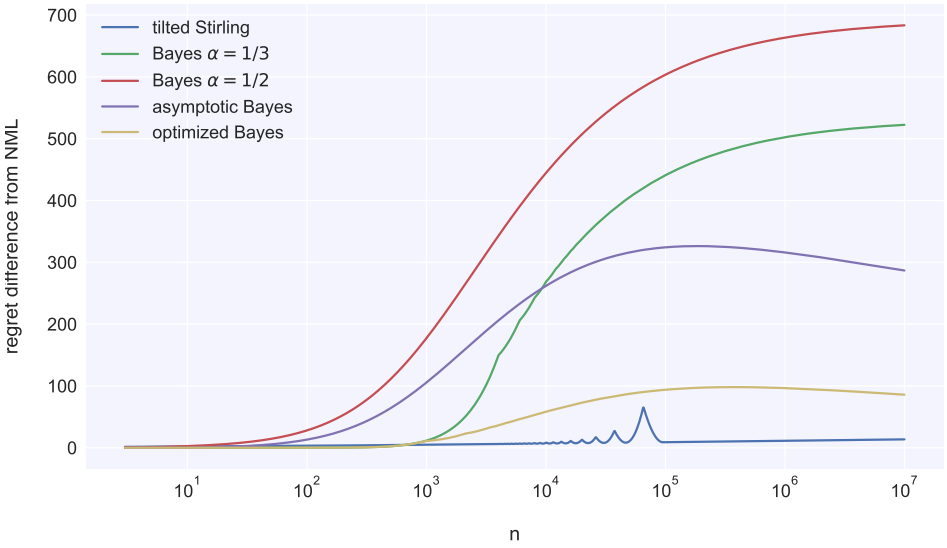

Figure 7: Worst-case regret differences from NML regret when $m = 2000$.

When $m = 2000$ (Figure 7), the tilted Stirling ratio distribution is close to NML. However, it will never actually converge to it, while the asymptotic Bayes mixture will. It can be seen that the 1/3-mixture achieves lower worst-case regret than the asymptotic formula until roughly $n = 9000$.

5.2 Regret as a function of the alphabet size

Next, we calculated the worst-case regrets when n is fixed and m grows. In Figure 8, n is fixed as $n = 100$. The worst-case regret of the 1/3-mixture is always lower than the worst-case regret of the tilted Stirling ratio distribution and converges to the NML regret as m grows. The 1/3-mixture also achieves lower worst-case regret than the mixture given by the asymptotic formula already when $m > 60$ and is optimal for $m > 200$.

We also calculated the worst-case regrets for $n = 1000$ (Figure 9), for which the tilted Stirling ratio distribution achieves the lowest worst-case regret except for small m. However, the 1/3-mixture will converge to zero eventually, unlike the tilted Stirling ratio distribution. The 1/3-mixture again achieves lower worst-case regret than the asymptotic formula when $m > 400$ and is optimal for roughly $m > 2000$.

For the worst-case regret of the 1/3-mixture to achieve lower worst-case regret than the tilted Stirling ratio distribution requires large m. This is seen in Figure 10, which shows the largest m such that the worst-case regret of the tilted Stirling ratio distribution is lower than that of the 1/3-mixture as a function of n. For example, m has to be around 150000 for $n = 10000$.

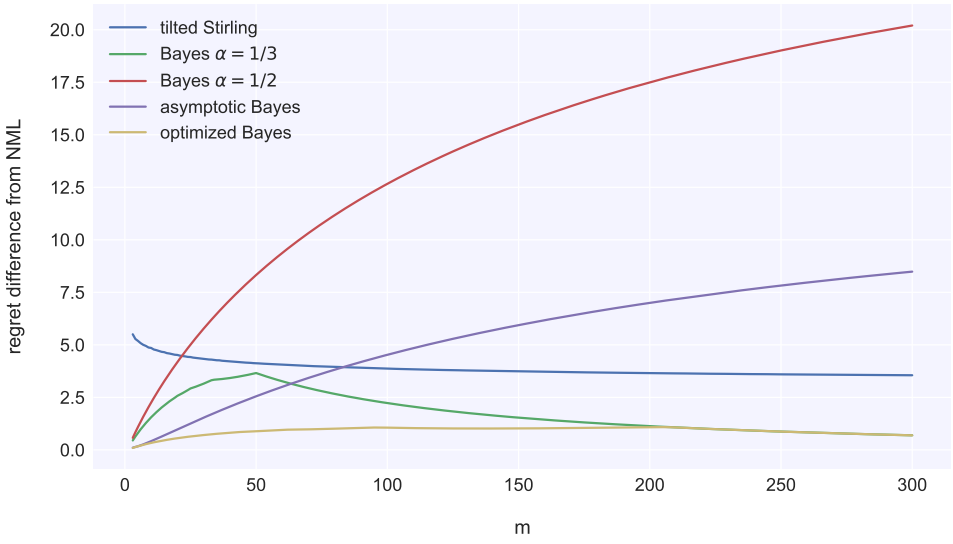

Figure 8: Worst-case regret differences from NML regret when $n = 100$.

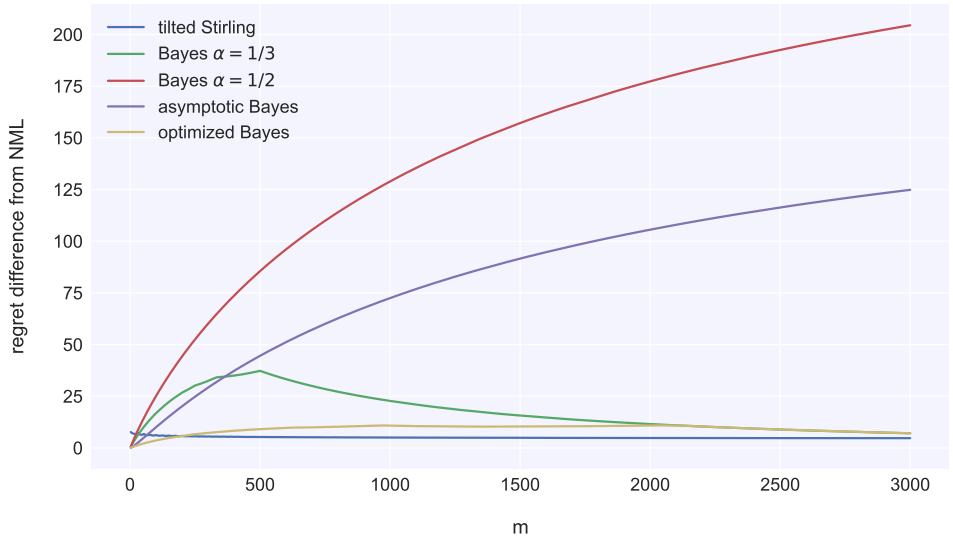

Figure 9: Worst-case regret differences from NML regret when $n = 1000$.

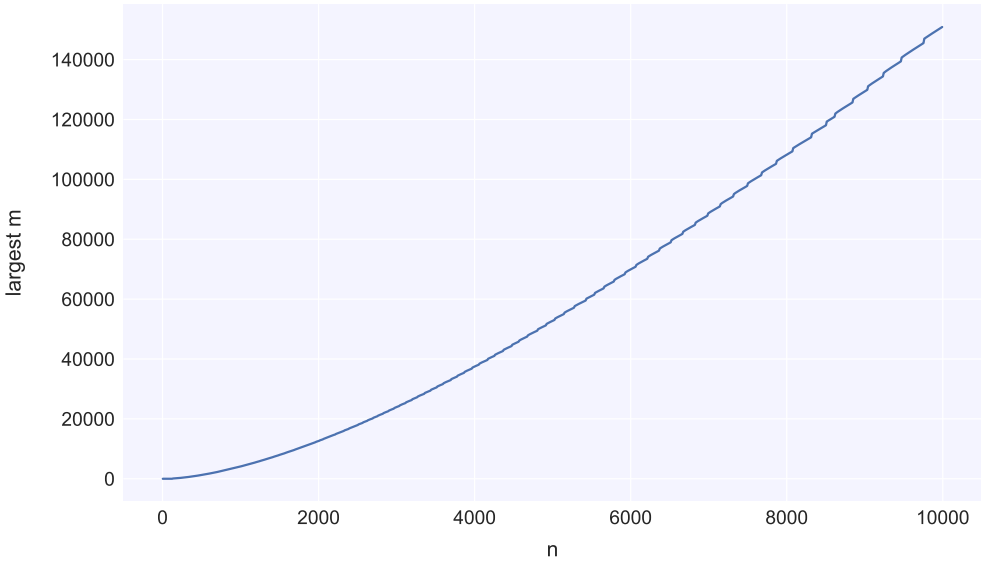

Figure 10: Largest m as a function of n such that the worst-case regret of the tilted Stirling ratio distribution is lower than that of the 1/3-mixture.

6 Discussion

In this thesis, we showed that the Bayes mixture can be an efficient universal code for the class of i.i.d. distributions and has the advantage of being fast to compute, even when optimizing the hyperparameter α. The minimax optimal hyperparameter for the Bayes mixture was shown to be 1/3 when the alphabet size is large compared to the sample size. The asymptotic properties of the 1/3-mixture show that as n is fixed and m approaches infinity, the worst-case regret of the 1/3-mixture approaches that of the NML distribution. This also results in a constant-time approximation of the NML regret for large m. In numerical experiments the 1/3-mixture achieves lower worst-case regret than an earlier proposed tilted Stirling ratio distribution when n is small or m is very large compared to n.

We also devised an algorithm that can compute the optimal hyperparameter efficiently for any sample size and alphabet size. This algorithm can be useful in all applications for universal coding on i.i.d. distributions, such as compression, prediction and gambling [42]. The choice of α is important if we wish to achieve as low regret as possible even in the worst case, and can be important in for example Bayesian network structure learning [33]. Certain choices of α achieve asymptotic minimaxity and thus approach the optimal worst-case regret when the sequence length increases.

Possible application areas for the 1/3-mixture include natural language processing and Bayesian networks as both can involve large alphabets. The 1/3-mixture provides a universal coding distribution whose worst-case performance is almost optimal when the size of the alphabet is large. Even though compression and probability estimation by add-constant rules on large alphabets have traditionally been avoided [25], the 1/3-mixture can be useful in for example model selection where its regret can be calculated in time not dependent m, or by allowing approximation of the NML regret in constant time. Furthermore, we note that the optimality of the 1/3-mixture holds not only in the limit, but for all values of n as long as m exceeds the derived bound. The minimax optimality of the $\mathrm{Dir}(1/3, \ldots, 1/3)$ prior can also serve as a theoretical justification for choosing the hyperparameters in a model with Dirichlet priors when the alphabet size is large.

There are several questions remaining for future work. One possibility is to study whether $\alpha = 1/3$ also minimizes the worst-case redundancy when the size of the alphabet is large compared to the sample size. Furthermore, it could also be useful to derive an asymptotic formula similar to the one proposed by Watanabe and Roos, but including a dependency on m yielding improved finite-sample performance for larger values of m. Finally, the behavior of Bayes mixtures should be studied in the large alphabet setting as building blocks in models that incorporate context. In particular, an algorithm similar to the context-tree weighting algorithm [39] which uses the Krichevsky-Trofimov estimator could be developed for large alphabets.

References

[1] Acharya, J., Das, H., Jafarpour, A., Orlitsky, A., and Suresh, A. T.: *Tight bounds for universal compression of large alphabets.* In *Proceedings of the IEEE International Symposium on Information Theory (ISIT 2013)*, pages 2875–2879. IEEE, May 2013.

[2] Acharya, J., Jafarpour, A., Orlitsky, A., and Suresh, A. T.: *Poissonization and universal compression of envelope classes.* In *Proceedings of the IEEE International Symposium on Information Theory (ISIT 2014)*, pages 1872–1876. IEEE, June 2014.

[3] Barron, A. R., Roos, T., and Watanabe, K.: *Bayesian properties of normalized maximum likelihood and its fast computation.* In *Proceedings of the IEEE International Symposium on Information Theory (ISIT 2014)*, pages 1667–1671. IEEE, June 2014.

[4] Batir, N.: *On some properties of digamma and polygamma functions.* Journal of Mathematical Analysis and Applications, 328(1):452–465, April 2007.

[5] Bernardo, J. M. and Smith, A. F. M.: *Bayesian Theory.* John Wiley & Sons, New York, NY, USA, 2001.

[6] Bontemps, D.: *Universal coding on infinite alphabets: exponentially decreasing envelopes.* IEEE Transactions on Information Theory, 57(3):1466–1478, March 2011.

[7] Cesa-Bianchi, N. and Lugosi, G.: *Prediction, Learning, and Games.* Cambridge University Press, Cambridge, UK, 2006.

[8] Chen, S. F. and Goodman, J.: *An empirical study of smoothing techniques for language modeling.* In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318. Association for Computational Linguistics, June 1996.

[9] Cover, T. M. and Thomas, J. A.: *Elements of Information Theory.* John Wiley & Sons, New York, NY, USA, 2012.

[10] DeSantis, A., Markowsky, G., and Wegman, M. N.: *Learning probabilistic prediction functions.* In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science (FOCS 1988)*, pages 110–119. IEEE, October 1988.

[11] Eggeling, R., Roos, T., Myllymäki, P., and Grosse, I.: *Robust learning of inhomogeneous PMMs.* In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics (AISTATS 2014)*, pages 229–237, April 2014.

[12] Grünwald, P.: *The Minimum Description Length Principle.* The MIT Press, Cambridge, MA, USA, 2007.

[13] Guo, B. and Qi, F.: *Refinements of lower bounds for polygamma functions.* Proceedings of the American Mathematical Society, 141(3):1007–1015, June 2013.

[14] Kiefer, J.: *Sequential minimax search for a maximum.* Proceedings of the American mathematical society, 4(3):502–506, February 1953.

[15] Kieffer, J.: *A unified approach to weak universal source coding.* IEEE Transactions on Information Theory, 24(6):674–682, November 1978.

[16] Kontkanen, P. and Myllymäki, P.: *A linear-time algorithm for computing the multinomial stochastic complexity.* Information Processing Letters, 103(6):227–233, September 2007.

[17] Kontkanen, P. and Myllymäki, P.: *MDL histogram density estimation.* In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS 2007)*, pages 219–226, March 2007.

[18] Krichevsky, R. and Trofimov, V.: *The performance of universal encoding.* IEEE Transactions on Information Theory, 27(2):199–207, March 1981.

[19] Laplace, P. S.: *A Philosophical Essay on Probabilities.* Dover, 1795/1951.

[20] Levin, B. and Reeds, J.: *Compound multinomial likelihood functions are unimodal: Proof of a conjecture of I.J. Good.* The Annals of Statistics, 5(1):79–87, January 1977.

[21] Määttä, J., Schmidt, D. F., and Roos, T.: *Subset selection in linear regression using sequentially normalized least squares: Asymptotic theory.* Scandinavian Journal of Statistics, 43(2):382–395, June 2016.

[22] MacKay, D. J. C.: *Information Theory, Inference and Learning Algorithms.* Cambridge university press, Cambridge, UK, 2003.

[23] Orlitsky, A. and Santhanam, N. P.: *Performance of universal codes over infinite alphabets.* In *Proceedings of the Data Compression Conference (DCC 2003)*, pages 402–410. IEEE, March 2003.

[24] Orlitsky, A. and Santhanam, N. P.: *Speaking of infinity [i.i.d. strings].* IEEE Transactions on Information Theory, 50(10):2215–2230, October 2004.

[25] Orlitsky, A., Santhanam, N. P., and Zhang, J.: *Always Good Turing: Asymptotically optimal probability estimation.* Science, 302(5644):427–431, October 2003.

[26] Orlitsky, A., Santhanam, N. P., and Zhang, J.: *Universal compression of memoryless sources over unknown alphabets.* IEEE Transactions on Information Theory, 50(7):1469–1481, July 2004.

[27] Rissanen, J.: *MDL denoising.* IEEE Transactions on Information Theory, 46(7):2537–2543, November 2000.

[28] Rissanen, J.: *Strong optimality of the normalized ML models as universal codes and information in data.* IEEE Transactions on Information Theory, 47(5):1712–1717, July 2001.

[29] Robbins, H.: *A remark on Stirling's formula.* The American Mathematical Monthly, 62(1):26–29, January 1955.

[30] Shannon, C. E.: *A mathematical theory of communication.* The Bell System Technical Journal, 27(3):379–423, July 1948.

[31] Shtarkov, Y. M.: *Universal sequential coding of single messages.* Problemy Peredachi Informatsii, 23(3):3–17, 1987.

[32] Silander, T.: *Bayesian network structure learning with a quotient normalized maximum likelihood criterion.* In *Proceedings of the Ninth Workshop on Information Theoretic Methods in Science and Engineering (WITMSE 2016)*, pages 32–25, September 2016.

[33] Silander, T., Kontkanen, P., and Myllymäki, P.: *On sensitivity of the MAP Bayesian network structure to the equivalent sample size parameter.* In *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence (UAI 2007)*, pages 360–367. AUAI Press, July 2007.

[34] Silander, T., Roos, T., Kontkanen, P., and Myllymäki, P.: *Factorized normalized maximum likelihood criterion for learning Bayesian network structures.* In *Proceedings of the Fourth European Workshop on Probabilistic Graphical Models (PGM 2008)*, pages 257–264, September 2008.

[35] Speed, T. P. and Yu, B.: *Model selection and prediction: normal regression.* Annals of the institute of statistical mathematics, 45(1):35–54, March 1993.

[36] Szpankowski, W. and Weinberger, M. J.: *Minimax pointwise redundancy for memoryless models over large alphabets.* IEEE Transactions on Information Theory, 58(7):4094–4104, July 2012.

[37] Takeuchi, J. and Barron, A. R.: *Asymptotically minimax regret by Bayes mixtures.* In *Proceedings of the IEEE International Symposium on Information Theory (ISIT 1998)*, page 318. IEEE, August 1998.

[38] Watanabe, K. and Roos, T.: *Achievability of asymptotic minimax regret by horizon-dependent and horizon-independent strategies.* Journal of Machine Learning Research, 16(1):2357–2375, January 2015.

[39] Willems, F. M. J., Shtarkov, Y. M., and Tjalkens, T. J.: *The context-tree weighting method: basic properties.* IEEE Transactions on Information Theory, 41(3):653–664, May 1995.

[40] Witten, I. H., Neal, R. M., and Cleary, J. G.: *Arithmetic coding for data compression.* Communications of the ACM, 30(6):520–540, June 1987.

[41] Xie, Q. and Barron, A. R.: *Minimax redundancy for the class of memoryless sources.* IEEE Transactions on Information Theory, 43(2):646–657, March 1997.

[42] Xie, Q. and Barron, A. R.: *Asymptotic minimax regret for data compression, gambling, and prediction.* IEEE Transactions on Information Theory, 46(2):431–445, March 2000.

[43] Yang, X.: *Compression and Predictive Distributions for Large Alphabets.* PhD thesis, Yale University, 2015.

[44] Yang, X. and Barron, A. R.: *Compression and predictive distributions for large alphabet i.i.d. and Markov models.* In *Proceedings of the IEEE International Symposium on Information Theory (ISIT 2014)*, pages 2504–2508. IEEE, July 2014.

[45] Yang, X. and Barron, A. R.: *Minimax compression and large alphabet approximation through Poissonization and tilting.* IEEE Transactions on Information Theory, 63(5):2866–2884, May 2017.