A Bound for Learning Lossless Source Coding with Online Learning

Anders Høst-Madsen, Mohammad Zaeri Amirani, Narayana Prasad Santhanam
Department of Electrical & Computer Engineering
University of Hawaii, Manoa
Honolulu, HI, 96822, Email: {ahm,zaeri,santhanam}@hawaii.edu

***Abstract*—This paper develops bounds for learning lossless source coding under the PAC (probably approximately correct) framework. The paper considers iid sources with online learning: first the coder learns the data structure from training sequences. When presented with a test sequence for compression, it continues to learn from/adapt to the test sequence. The results show, not unsurprisingly, that there is little gain from online learning when the training sequence length is much longer than the test sequence length. But if the test sequence length is longer than the training sequence, there is a significant gain. Coders for online learning has a somewhat surprising structure: the training sequence is used to estimate a confidence interval for the distribution, and the coding distribution is found through a prior distribution over this interval.**

I. Introduction

We consider lossless coding of sources that are (or assumed to be) in some probability class Λ characterized by an unknown, determinstic parameter vector $\boldsymbol{\theta}$. We consider this in the context of learned coding [1]. We are given a training sequence x^m; based on the training we develop coders $C(x^l;x^m)$ with length function $L(x^l;x^m)$ for encoding *test sequences* x^l. The codelength is $E_\theta[L(X^l;x^m)|x^m]$ (the expectation here is only over x^l), and we measure performance by the redundancy

$$R_l(L,x^m,\boldsymbol{\theta})=E_{\boldsymbol{\theta}}[L(X^l;x^m)|x^m]-lH_{\boldsymbol{\theta}}(X). \tag{1}$$

The redundancy depends on the training sequence x^m. One way to remove this dependency is to average also over x^m,

$$R_l(L,m,\boldsymbol{\theta})=E_{\boldsymbol{\theta}}[L(X^l;X^m)]-lH_{\boldsymbol{\theta}}(X) \tag{2}$$

As usual in source coding (and ML) the performance measure is the worst case over the deterministic θ:

$$R_l^+(m)=\min_L\sup_\theta R_l(L,m,\boldsymbol{\theta}). \tag{3}$$

The paper [2] considers (3), and proves

$$\frac{1}{2m\ln 2}+o\left(\frac{1}{m}\right)\leq\frac{1}{l}R_l^+(m)\leq\frac{\alpha_0}{m\ln 2}+o\left(\frac{1}{m}\right) \tag{4}$$

$$\alpha_0\approx 0.50922. \tag{5}$$

The result was improved in [3] to show that

$$\frac{1}{l}R_l^+(m)=\frac{1}{2m\ln 2}+o\left(\frac{1}{m}\right). \tag{6}$$

However, in machine learning performance usually is not measured by average over test sequences, see [4], [5]. One way performance is measured is in the PAC (probability approximately correct) learning framework [5]. Rather than usual error probability in classification, we use the redundancy (1) as risk measure. We can then say that coding in a class or sources is PAC-learnable if for any $a>0,P_e>0$

$$\inf_{\boldsymbol{\theta}}P\left(R_l(L,X^m,\boldsymbol{\theta})\leq a\right)\geq 1-P_e$$

where the probability is over X^m. Alternatively, we can state this by defining

$$E(m,l,a)=\sup_{\boldsymbol{\theta}}P\left(R_l(L,X^m,\boldsymbol{\theta})>a\right), \tag{7}$$

For some given a and small P_e the goal is then to ensure $E(m,a)\leq P_e$. Thus, we require the redundancy of the learned codelength to be smaller than a, except with a small error probability P_e. Equivalently, we can define

$$a(m,l,P_e)=\inf\{a:E(m,l,a)\leq P_e\} \tag{8}$$

In [1] we analyzed this problem for a binary alphabet and a *frozen* coder, i.e., a coder that does not continue to learn from a test sequence. The result is (Q is the survival function of the normal distribution)

Theorem 1. *For P_e sufficiently small,*

$$\frac{1}{l}a(m,l,P_e)\geq\frac{Q^{-1}(P_e/2)^2}{2m\ln 2}+o\left(\frac{1}{m}\right). \tag{9}$$

for the estimator $\hat{p}=\frac{k+\alpha}{m+2\alpha}$, where k is the number of 1s seen in x^m. The optimum value of α that satisfies $\frac{1}{6}Q^{-1}(P_e/2)^2-1\leq\alpha\leq\frac{1}{6}Q^{-1}(P_e/2)^2+1$ which gives an achievable $a(m,P_e)$;

$$\frac{1}{l}a(m,l,P_e)=b(P_e)\frac{Q^{-1}(P_e/2)^2}{2m\ln 2}+o\left(\frac{1}{m}\right), \tag{10}$$

where $\lim_{P_e\to 0}b(P_e)=1$.

We generalized this to finite state machines and general alphabets in [6]. The assumption in these papers is that there is a distinct learning phase, where the coder learns the distribution of data, and then a coding phase where the coding distribution is fixed. This makes the result not quite fundamental. The coder could continue to learn from the test sequence, already suggested in [7], which is a type of online

The research was funded in part by the NSF grants CCF-1908957, CCF-2324396, and NRT-AI 2244574.

learning. In this paper we find bounds for this case. Since this problem is much more complex, we limit ourselves to the binary iid case, which still can provide significant insight.

II. Online Learning with Average Performance

For online learning, the limit when both $l, m \to \infty$, where $\frac{l}{m}$ is kept fixed makes the most sense from an operational point of view.

We consider first the average case, i.e., averaging over both training and test sequences. Although this is not of main interest, the results are nicer and do give some insight. In this case, for a frozen coder, we get from (6)

$$R_l^+(m) = \frac{l}{2m\ln 2} + \epsilon\left(\frac{1}{m}\right) \tag{11}$$

where $\epsilon(x)$ denotes any function converging to zero as $x \to 0$.

For the achievable rate for the online coder, a natural approach is to just continue updating the coding distribution as in [2], which results in

$$\begin{aligned} R_l^+(m) &\leq \frac{1}{2\ln 2}\sum_{k=m}^{l+m-1}\frac{1}{k} + \epsilon\left(\frac{1}{m}\right) \\ &= \frac{1}{2}\log\left(1+\frac{l}{m}\right) + \epsilon\left(\frac{1}{m}\right) \end{aligned} \tag{12}$$

We can lower bound the redundancy by

Proposition 1.

$$R_l^+(m) \geq \max_{\pi(\theta)} I(\theta; X^l | X^m)$$

The proof can be done as in [8]. Now

$$\begin{aligned} \max_{\pi(\theta)} I(\theta; X^l|X^m) &= \max_{\pi(\theta)}\{I(\theta; X^{l+m}) - I(\theta; X^m)\} \\ &\geq \max_{\pi(\theta)} I(\theta; X^{l+m}) - \max_{\pi(\theta)} I(\theta; X^m) \end{aligned}$$

From [9] we have

$$\max_{\pi(\theta)} I(\theta; X^n) = \frac{1}{2}\log\frac{n}{2\pi e} + \frac{1}{n}\log\frac{\pi^{1/2}}{\Gamma(1/2)}$$

and thus

$$R_l^+(m) \geq \frac{1}{2}\log\left(1+\frac{l}{m}\right) + \epsilon\left(\frac{1}{m}\right) \tag{13}$$

For $\frac{l}{m}$ small, series expansion of (13) gives (11). This is the case when there is more training data than test data, which is the common scenario in machine learning. We therefore get the common sense conclusion that there is nothing gained from continuing to update with the test data. However, if $l \gg m$ is there a significant gain. This is not a usual learning scenario, but one situation where this could happen is when the length of test sequences are unknown, and the training data therefore could be too little. Notice that training still helps: the redundancy is reduced from $\frac{1}{2}\log l$ to $\frac{1}{2}\log\frac{l}{m}$.

III. Online Learning with the PAC Criterion

As in Theorem 1 the aim is to find the asymptotic limit $\lim_{m\to\infty}\sup_p ma(m,l,P_e)$. Let $p_{\max}(m)$ be a value of p where $a(m,l,P_e)$ is achieved. As argued in [1] instead of finding the maximum of $ma(m,l,P_e)$ for each m we can consider convergent sequences $mp(m)$ and the take the maximum in the end. We can divide such sequences into three regimes: the *CLT regime*: $\lim_{m\to\infty} mp(m) = \infty$, where the central limit theorem (CLT) can be applied and the *Poisson regime* $0 < \lim_{m\to\infty} mp(m) < \infty$, where a Poisson approximation can be used

A. Converse

For the PAC learning criterion, we first derive a converse. For online learning a in (8) depends on both l and m.

Theorem 2. *Consider the asymptotic regime where $l, m \to \infty$ while $\frac{l}{m}$ is fixed. Then*

$$a(m,l,P_e) \geq C_a(A)$$

where $C_a(A)$ is capacity of the additive white noise Gaussian (AWGN) channel with amplitude *constraint A and noise power 1, and where*

$$A = \sqrt{\frac{l}{m}}Q^{-1}(P_e/2)$$

Proof: A general coder based on the training data x^m assigns a probability $\hat{p}^l(m)(x^l)$ to the test sequence x^l. The redundancy of the coder is (within 1 bit) is $D(p^l\|\hat{p}^l(m))$, where p^l is the IID distribution on x^l. With this

$$a(m,l,P_e) = \inf_a \max_p P(D(p^l\|\hat{p}^l(m)) \geq a) \leq P_e$$

Here $\hat{p}^l(m)$ depends only on the sufficient statistic $\check{p} = \frac{k}{m}$, so we will write $\hat{p}^l(\check{p})$ and the probability is with respect to $\check{p}$. We can therefore also write

$$\begin{aligned} a(m,l,P_e) &= \inf_{S:\forall p:P(S(p))\geq 1-P_e} \max_p \sup_{\check{p}\in S_p} D(p^l\|\hat{p}^l(\check{p})) \\ &\geq \inf_{S::\forall p:P(S(p))\geq 1-P_e} \sup_{\check{p}} \sup_{p\in\check{S}_{\check{p}}} D(p^l\|\hat{p}^l(\check{p})) \end{aligned}$$

Here $S : [0,1] \to 2^{[0,1]}$ is a set function of p: for every p it gives a (measurable) subset $S_p \subset (\delta, 1-\delta)$, and explicitly $P(S(p)) = P(\check{p} \in S(p))$, while

$$\check{S}_{\check{p}} = \{p : \check{p} \in S(p)\}.$$

We can think of $\check{S}_{\check{p}}$ as a kind of confidence interval. We therefore have the lower bound

$$a(m,l,P_e) \geq \inf_{\hat{p}^l} \inf_{S:\forall p:P(S(p))\geq 1-P_e} \sup_{\check{p}} \sup_{p\in\check{S}_{\check{p}}} D(p^l\|\hat{p}^l(\check{p}))$$

As in the proof of the lower bound in Theorem 1 in [1], we consider a lower bound in the CLT regime only. Thus, we let $p \in [\delta, 1-\delta]$ be fixed as $m \to \infty$; we can consider a slightly smaller interval than $[0,1]$ to avoid endpoint effect, and let $\delta \to 0$ in the end. Now

$$\begin{aligned}
&\inf_{S:\forall p:P(S(p))\geq 1-P_e} \inf_{\hat{p}^l} \sup_{\check{p}} \sup_{p\in \check{S}_{\check{p}}} D(p^l \| \hat{p}^l(\check{p})) \\
&\overset{(a)}{=} \inf_{S:\forall p:P(S(p))\geq 1-P_e} \inf_{\hat{p}^l} \max_{\check{p}} \max_{\pi} E_\pi[D(p^l\|\hat{p}^l(\check{p}))] \\
&\overset{(b)}{\geq} \inf_{S:\forall p:P(S(p))\geq 1-P_e} \max_{\check{p}} \inf_{\hat{p}^l} \max_{\pi} E_\pi[D(p^l\|\hat{p}^l(\check{p}))] \\
&\overset{(c)}{=} \inf_{S:\forall p:P(S(p))\geq 1-P_e} \max_{\check{p}} \max_{\pi} \min_{\hat{p}^l} E_\pi[D(p^l\|\hat{p}^l(\check{p}))] \\
&\overset{(d)}{\geq} \inf_{S:\forall p:P(S(p))\geq 1-P_e} \max_{\check{p}} \max_{\pi} I(\theta;x^l) \qquad (14)
\end{aligned}$$

In step (a), π is an arbitrary distribution for p over $\check{S}_{\check{p}}$; (a) is true because we can use point distribution as special case. Step (b) is true because $\hat{p}^l$ is in fact a direct function of $\check{p}$. Step (c) is true because $E_{\pi,\pi_\pm}[D(p^l\|\hat{p}^l(\check{p}))]$ is concave in $\hat{p}^l$ and linear in π (see [10]). Step (c) is true because we can see $\max_\pi \min_{\hat{p}^l} E_{\pi_\pm} D(p^l\|\hat{p}^l(\check{p}^l))$ as a standard universal source coding problem where the unknown parameter is in $\check{S}_{\check{p}}$, and we can therefore apply Gallagher's lower bound [10], [9]; θ here is a random variable distributed according to π over $\check{S}_{\check{p}}$, and x^l is IID Bernoulli according to θ. Notice that $I(\theta;x^l) = I(\theta;\bar{p}) = I(\theta-p;\bar{p}-p)$, where $\bar{p}=\frac{\bar{k}}{l}$ with $\bar{k}$ the number of ones in the test sequence.

Explicitly,

$$\begin{aligned}
I(\theta;\bar{p}) &= E\left[-\log\left(\int \frac{P(\bar{p}|\tilde{\theta})}{P(\bar{p}|\theta)} dF(\tilde{\theta})\right)\right] \\
&= \int \sum_{\bar{p}} -\log\left(\int \frac{P(\bar{p}|\tilde{\theta})}{P(\bar{p}|\theta)} dF(\tilde{\theta})\right) P(\bar{p}|\theta) dF(\theta)
\end{aligned} \qquad (15)$$

Let $s_l = l\bar{p}$. Then the local CLT for integer-valued random variables [11] states

$$\begin{aligned}
P(s_l|\theta) &= \frac{1}{\sqrt{2\pi l\theta(1-\theta)}} \exp\left(-\frac{(s_l-l\theta)^2}{2l\theta(1-\theta)}\right) + o\left(\frac{1}{\sqrt{l}}\right) \\
&= f_{\mathcal{N}(l\theta,l\theta(1-\theta))}(s_l) + o\left(\frac{1}{\sqrt{l}}\right)
\end{aligned}$$

where $o\left(\frac{1}{\sqrt{l}}\right)$ is uniform in s_l.

The natural choice for S is the centered interval around the mean based on the CLT

$$S(p) = \left[p - \frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}(P_e/2), p + \frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}(P_e/2)\right] \qquad (16)$$

which asymptotically has probability $1-P_e$. We will later see that this is optimum. The corresponding $\check{S}_{\check{p}}$ is

$$\begin{aligned}
\check{S}_{\check{p}} = \Bigg[&\check{p} - \frac{\sqrt{\check{p}\check{q}}}{\sqrt{m}} Q^{-1}(P_e/2) + o\left(\frac{1}{\sqrt{m}}\right), \\
&\check{p} + \frac{\sqrt{\check{p}\check{q}}}{\sqrt{m}} Q^{-1}(P_e/2) + o\left(\frac{1}{\sqrt{m}}\right)\Bigg]
\end{aligned}$$

Then $\sup_{\theta\in\check{S}_{\check{p}}} |\theta-\check{p}| = O\left(\frac{1}{\sqrt{l}}\right)$ since the ratio l/m is fixed. Further,

$$\begin{aligned}
&f_{\mathcal{N}(l\theta,l\theta(1-\theta))}(s_l) \\
&\quad = f_{\mathcal{N}(l\theta,l\check{p}(1-\check{p}))}(s_l) \\
&\quad + f_{\mathcal{N}(l\theta,l\check{p}(1-\check{p}))}(s_l) \left(a(\check{p}) + b(\check{p}) \frac{(s_l-l\theta)^2}{l}\right)(\theta-\check{p}) \\
&\quad + o(\theta-\check{p})
\end{aligned}$$

where a and b are functions of $\check{p}$ alone. Here it can be seen that

$$\lim_{l\to\infty} \sup_{s_l} f_{\mathcal{N}(l\theta,l\check{p}(1-\check{p}))}(s_l) \frac{(s_l-l\theta)^2}{l} = 0 \qquad (17)$$

and therefore

$$\begin{aligned}
P(s_l|\theta) &= f_{\mathcal{N}(l\theta,l\check{p}(1-\check{p}))}(s_l) + o\left(\frac{1}{\sqrt{l}}\right) \\
&= \frac{1}{\sqrt{l\check{p}(1-\check{p})}} f_{\mathcal{N}(\mu,1)}(x_l) + o\left(\frac{1}{\sqrt{l}}\right)
\end{aligned} \qquad (18)$$

where the $o(\cdot)$-term is still uniform in s_l, and

$$x_l = \frac{s_l - l\check{p}}{\sqrt{l\check{p}(1-\check{p})}} \qquad \mu = \frac{\sqrt{l}(\theta-\check{p})}{\sqrt{\check{p}(1-\check{p})}}$$

At first we will limit $-B \leq x_l \leq B$, which also means $B_- = -B\sqrt{l\check{p}(1-\check{p})} + l\check{p} \leq s_l \leq B\sqrt{l\check{p}(1-\check{p})} + l\check{p} = B_+$. We can then find the limit of (15) by calculating equation (19) at the top of the next page. Then letting $B\to\infty$,

$$\max_\pi I(\theta;x^l) \to \max_{\tilde{\pi}} I(X;Y)$$

where $Y = X+N$, $N\sim\mathcal{N}(0,1)$, and $\tilde{\pi}$ is a distribution over $I_A = [-\sqrt{\frac{l}{m}}Q^{-1}(P_e/2), \sqrt{\frac{l}{m}}Q^{-1}(P_e/2)]$. Notice that this is independent of $\check{p}$, thus it is a minimax solution conditioned on (16) being optimum. We can argue for this as follows. If we remove some $\check{p}_1$ from S_p given by (16), we will have to add at least one other point $\check{p}_2$ outside the interval in (16) (recall that the distribution is discrete). Thus, p is removed from $\check{S}_{\check{p}_1}$ but added to $\check{S}_{\check{p}_2}$. These added points will result in points x outside the interval I_A. This will strictly increase the capacity of the corresponding channel. Namely, the capacity achieving distribution for the amplitude constrained Gaussian channel is discrete [12], and we can then move one of the discrete modulation points to the new value of $x\notin I_A$. Since this has a larger amplitude, it will increase capacity. Thus, if we move points outside (16) it will increase (14) showing that (16) minimizes (14).

■

There is no closed form expression for the capacity of the amplitude constrained AWGN. However, Smith [12] showed that the capacity is achieved by a discrete input distribution, and provided a numerical optimization method for finding the optimum input distribution and capacity. While there is no closed form expression, the papers [13], [14] show that the

$$\begin{aligned}
I(\theta;\bar{p})^B &\equiv \int_{\check{S}_{\check{p}}} \sum_{s_l=B_-}^{B_+} -\log\left(\int_{\check{S}_{\check{p}}} \frac{P(s_l|\tilde{\theta})}{P(s_l|\theta)} dF(\tilde{\theta})\right) P(s_l|\theta) dF(\theta) \\
&= \int_{-\sqrt{l/m}Q^{-1}(P_e/2)+\epsilon(m)}^{\sqrt{l/m}Q^{-1}(P_e/2)+\epsilon(m)} \sum_{s_l=B_-}^{B_+} -\log\left(\int_{-\sqrt{l/m}Q^{-1}(P_e/2)+\epsilon(m)}^{\sqrt{l/m}Q^{-1}(P_e/2)+\epsilon(m)} \frac{f_{\mathcal{N}(\tilde{\mu},1)}(x_l)+\epsilon(l)}{f_{\mathcal{N}(\mu,1)}(x_l)+\epsilon(l)} dF(\tilde{\mu})\right) \\
&\times \left(f_{\mathcal{N}(\mu,1)}(x_l)+\epsilon(l)\right) \frac{1}{\sqrt{l\check{p}(1-\check{p})}} dF(\mu) \\
&\to \int_{-\sqrt{l/m}Q^{-1}(P_e/2)}^{\sqrt{l/m}Q^{-1}(P_e/2)} \int_{-B}^{B} -\log\left(\int_{-\sqrt{l/m}Q^{-1}(P_e/2)}^{\sqrt{l/m}Q^{-1}(P_e/2)} \frac{f_{\mathcal{N}(\tilde{\mu},1)}(x)}{f_{\mathcal{N}(\mu,1)}(x)} dF(\tilde{\mu})\right) f_{\mathcal{N}(\mu,1)}(x) dx dF(\mu) \qquad (19)
\end{aligned}$$

capacity approximately is

$$C_a(A,\sigma) \approx \min\left\{\log\left(1+A\sqrt{\frac{2}{\pi e}}\right), \frac{1}{2}\log\left(1+A^2\right)\right\}$$

Thus,

$$\begin{aligned}
a(m,l,P_e) \gtrapprox \min\Bigg\{&\log\left(1+\sqrt{\frac{l}{m}}Q^{-1}(P_e/2)\sqrt{\frac{2}{\pi e}}\right), \\
&\frac{1}{2}\log\left(1+\frac{l}{m}Q^{-1}(P_e/2)^2\right)\Bigg\}
\end{aligned}$$

We see that we reach the same conclusion as for average performance in Section II. For pure training, in the regime where $\frac{l}{m}$ is fixed, we get from (9) that

$$R_l^+(m) \geq \frac{lQ^{-1}(P_e/2)^2}{2m\ln 2} + \epsilon\left(\frac{1}{m}\right)$$

Thus, essentially, $\frac{lQ^{-1}(P_e/2)^2}{m}$ moves inside the logarithm. As for average performance, for $\frac{l}{m}$ small there is little gain, but for $l \gg m$, there could be a gain. We will show that by developing a coder that can realize some of that gain in this regime.

B. Coding

The naive coder would be to simply update $\hat{p}$ with new data, as was optimum in the average case, eq. (12). However, it can be shown that this does not move $\frac{lQ^{-1}(P_e/2)^2}{m}$ inside the log as in the converse. Rather, the proof of the converse hints at how a coder should be designed: From $\hat{p}$ from the training, a P_e confidence interval for p is found, and the test sequence is then coded according to some prior distribution over this confidence interval.

Let $[\check{p}_-, \check{p}_+]$ be a confidence interval for p based on the training data. This should be a proper confidence interval, meaning that $\forall p : P(p \notin [\check{p}_-, \check{p}_+]) \leq P_e$. We use a uniform distribution over $[\check{p}_-, \check{p}_+]$. The probability of a test sequence x^l with $\bar{k}$ ones can then be calculated as

$$\begin{aligned}
&\frac{1}{\check{p}_+ - \check{p}_-}\int_{\check{p}_-}^{\check{p}_+} \theta^{\bar{k}}(1-\theta)^{l-\bar{k}} d\theta \\
&= \frac{1}{\check{p}_+ - \check{p}_-}\left(I_{\check{p}_+}(\bar{k}+1, l-\bar{k}+1) - I_{\check{p}_+}(\bar{k}+1, l-\bar{k}+1)\right) \\
&\times B(\bar{k}+1, l-\bar{k}+1) \\
&= \frac{1}{\check{p}_+ - \check{p}_-}\left(F(\bar{k}; l+1, \check{p}_-)F(\bar{k}; l+1, \check{p}_+)\right) \\
&\times B(\bar{k}+1, l-\bar{k}+1)
\end{aligned}$$

where $F(\bar{k}; l+1, \check{p})$ is the CDF for the binomial distribution, and $B(\bar{k}+1, l-\bar{k}+1)$ is the Beta function. The codelength is $-\log$ of this probability. To bound $-\log B(\bar{k}+1, l-\bar{k}+1)$ we can use the bound in [10, 13.2], and we then get the following codelength bound

$$\begin{aligned}
L \leq\, & lH\left(\frac{\bar{k}}{l}\right) + \frac{1}{2}\log(l) - \frac{1}{2}\log\left(\pi\frac{\bar{k}}{l}\frac{l-\bar{k}}{l}\right) + 2 \\
& + \log(\check{p}_+ - \check{p}_-) - \log(F(\bar{k}; l+1, \check{p}_-) - F(\bar{k}; l+1, \check{p}_+)) \qquad (20)
\end{aligned}$$

The coder is based on finding a proper confidence interval. Now, from [1], [2] we know that the critical performance is in the Poisson regime. We therefore focus on performance in the Poisson regime. By calling $k_- = mp_-$ and $k_+ = mp_+$, according to [15] we can set

$$\begin{aligned}
k_- &= \frac{1}{2}\chi^2(P_e/2; 2k) \\
k_+ &= \frac{1}{2}\chi^2(1-P_e/2; 2k+2) \qquad (21)
\end{aligned}$$

We now rewrite (20) as

$$\begin{aligned}
L \leq\, & lH\left(\frac{\bar{k}}{l}\right) + \frac{1}{2}\log\left(\frac{l}{m}\right) - \frac{1}{2}\log\left(\bar{k}\frac{m}{l}\right) \\
& - \frac{1}{2}\log\left(\frac{l-\bar{k}}{l}\right) + 2 - \frac{1}{2}\log(\pi) \\
& + \log(\check{k}_+ - \check{k}_-) - \log(F(\bar{k}; l+1, \check{p}_-) - F(\bar{k}; l+1, \check{p}_+))
\end{aligned}$$

The expected codelength with respect to $\bar{k}$ is[1]

$$
\begin{aligned}
L \leq & lE\left[H\left(\frac{\bar{k}}{l}\right)\right]+\frac{1}{2}\log\left(\frac{l}{m}\right)-\frac{1}{2}E\left[\log\left(\bar{k}\frac{m}{l}\right)\right]\\
& -\frac{1}{2}E\left[\log\left(\frac{l-\bar{k}}{l}\right)\right]+2-\frac{1}{2}\log(\pi)\\
& +\log(\check{k}_+-\check{k}_-)-E[\log(F(\bar{k};l+1,\check{p}_-)-F(\bar{k};l+1,\check{p}_+))]\\
& \leq lH\left(p\right)+\frac{1}{2}\log\left(\frac{l}{m}\right)-\frac{1}{2}E\left[\log\left(\bar{k}\frac{m}{l}\right)\right]\\
& -\frac{1}{2}E\left[\log\left(\frac{l-\bar{k}}{l}\right)\right]+2-\frac{1}{2}\log(\pi)\\
& +\log(\check{k}_+-\check{k}_-)-E[\log(F(\bar{k};l+1,\check{p}_-)-F(\bar{k};l+1,\check{p}_+))]
\end{aligned}
$$

We use

Lemma 3. *Assume $p \leq \frac{1}{2}$ and $\bar{k} \leq \frac{1}{2}l$. Then*

$$\lim_{l\to\infty} -E\left[\log\left(\frac{l-\bar{k}}{l}\right)\right] \leq 1$$

Proof: It can be shown that by setting $\kappa = 4\ln 2 - 2$ the function $f(x) = \kappa x^2 + x + \ln(1-x) \geq 0$ for all real $0 \leq x \leq \frac{1}{2}$. In fact, the function is zero on the boundaries and has positive value at its only extremum point $x^* = 1 - \frac{1}{2\kappa}$ where $f'(x^*) = 0$. Therefore, by taking expectation in the Poisson regime, we have:

$$
\begin{aligned}
-E\left[\log(\frac{l-\bar{k}}{l})\right]\ln 2 \leq \kappa\frac{E[\bar{k}^2]}{l^2}+\frac{E[\bar{k}]}{l} = \kappa p^2 + p + \kappa\frac{p}{l}\\
\leq \ln 2 + o\left(\frac{1}{l}\right)
\end{aligned}
$$

for $p \leq \frac{1}{2}$ as it is claimed. ■

The redundancy in Poisson limit is therefore bounded by

$$
\begin{aligned}
a(m,l,P_e) \leq \frac{1}{2}\log\left(\frac{l}{m}\right)-\frac{1}{2}E\left[\log\left(\bar{k}\frac{m}{l}\right)\right]+2.5-\frac{1}{2}\log(\pi)\\
+\log(\check{k}_+-\check{k}_-)-E\left[\log(\mathbb{P}_{\frac{l}{m}k_-}(\bar{k})-\mathbb{P}_{\frac{l}{m}k_+}(\bar{k}))\right]
\end{aligned} \tag{22}
$$

The expectation in (22) is with respect to $\bar{k}$. Suppose the Poisson limit of k has mean γ; then $\bar{k}$ converges to a Poisson distribution with mean $\frac{l}{m}\gamma$. Notice that $k_\pm$ is based on k. In order to get the different terms to the same scale, we therefore calculate expectation with respect to a Poisson distribution with mean $\frac{l}{m}\gamma$, explicitly

$$
\begin{aligned}
a(m,l,P_e) \leq & \frac{1}{2}\log\left(\frac{l}{m}\right)+1-\frac{1}{2}\log(\pi)\\
& -\frac{1}{2}E_{\frac{l}{m}\gamma}\left[\log\left(\bar{k}\frac{m}{l}\right)\right]+\log(\check{k}_+-\check{k}_-)\\
& -E_{\frac{l}{m}\gamma}\left[\log(\mathbb{P}_{\frac{l}{m}k_-}(\bar{k})-\mathbb{P}_{\frac{l}{m}k_+}(\bar{k}))\right]
\end{aligned} \tag{23}
$$

The first line of (23) gives the dependency of $a(m,l,P_e)$ on $\frac{l}{m}$ explicitly, and is independent of P_e and γ and k. The second line is dependent on k and γ. The bound should be maximized over k and γ, but conditioned on successful training, that is $p \in [\check{p}_-,\check{p}_+]$, that is the maximum should be calculated over $k,\gamma : k_- \leq \gamma \leq k_+$. We define

$$
\begin{aligned}
c(P_e) = \sup_{\gamma,k:k_-\leq\gamma\leq k_+} & -\frac{1}{2}E_{\frac{l}{m}\gamma}\left[\log\left(\bar{k}\frac{m}{l}\right)\right]+\log(\check{k}_+-\check{k}_-)\\
& -E_{\frac{l}{m}\gamma}\left[\log(\mathbb{P}_{\frac{l}{m}k_-}(\bar{k})-\mathbb{P}_{\frac{l}{m}k_+}(\bar{k}))\right]
\end{aligned} \tag{24}
$$

The function $c(P_e)$ is not quite independent of $\frac{l}{m}$ due to the discrete nature of the Poisson distribution, but there is only a weak dependency. The function $c(P_e)$ can be calculated numerically. In Fig. 1 we have plotted a result of the numerical calculation. Of course online learning is much better than pure training for $\frac{l}{m}$ large (the logarithmic plot disguises this somewhat); the advantage disappears for $\frac{l}{m} = 0.3$. The main reason the online learning achievable rate is worse than pure learning is mainly due to the bound from [10, Section 13.2] used in (20) is not that tight.

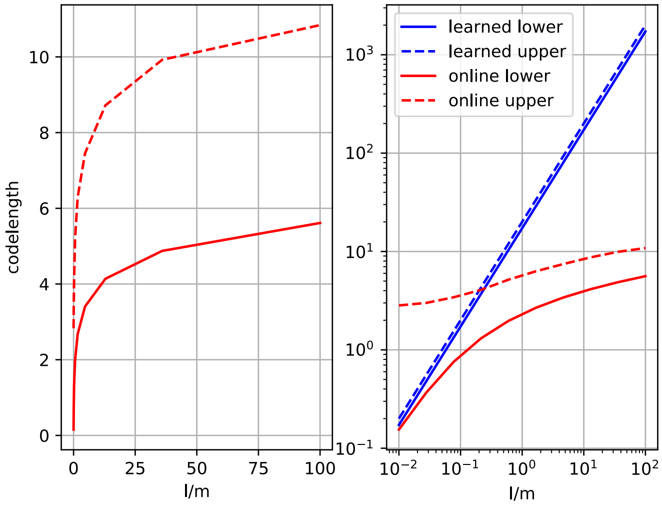

Figure 1. Plot of codelength for online learning for $P_e = 10^{-6}$.

IV. Conclusion

As would be expected from common sense, if $l \ll m$, not much is gained from online learning; this is of course the usual learning scenario. But if l is of the same order as m there is a significant gain. One could think of this scenario more as universal coding prepped by training. One significant result of the paper is that Theorem 2 provides a lower bound in *any* kind of learned coder, without restricting it to be for example a frozen coder. One could imagine a coding scheme which decides between the learned coder and a universal coder on a sequence by sequence basis. But Theorem 2 shows that this cannot beat a pure learned coder.

Another significant insight is that online learning is quite different than both pure learning and universal coding. In either of these case, coding is done by estimating the probability distribution. But online learning is totally different: the training data is used to calculate a confidence interval, and the coding distribution is found by using a prior over this confidence interval.

[1] The expectation as written is clearly infinite. This is because the expression (20 is not valid for $\bar{k} = 0, l$. These cases therefore have to be handled separately in numerical evaluation.

References

[1] A. Høst-Madsen, “Bounds for learning lossless source coding,” in *ISIT’2021, Melbourne, Australia, July 12-20, 2021*, 2021.

[2] R. E. Krichevskiy, “Laplace’s law of succession and universal encoding,” *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 296–303, Jan 1998.

[3] D. Braess and T. Sauer, “Bernstein polynomials and learning theory,” *Journal of Approximation Theory*, vol. 128, no. 2, pp. 187–206, 2004.

[4] V. N. Vapnik, *Statistical Learning Theory*. John Wiley, 1998.

[5] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. MIT Press, 2018.

[6] M. Z. Amirani and A. Høst-Madsen, “Learning source coding for general alphabets and finite state machines,” in *59th Annual Allerton Conference on Communications, Control, and Computing, Urbana-Champaign, Illinois, September 26-29, 2023*, 2023.

[7] J. Schmidhuber and S. Heil, “Sequential neural text compression,” *IEEE Transactions on Neural Networks*, vol. 7, no. 1, pp. 142–146, 1996.

[8] Y. Fogel and M. Feder, “Universal batch learning with log-loss,” in *IEEE International Symposium on Information Theory: ISIT’18 (Vail, Colorado)*, 2018.

[9] G. Shamir, “On the mdl principle for i.i.d. sources with large alphabets,” *Information Theory, IEEE Transactions on*, vol. 52, no. 5, pp. 1939–1955, May 2006.

[10] T. Cover and J. Thomas, *Information Theory, 2nd Edition*. John Wiley, 2006.

[11] G. R. Grimmett and D. R. Stirzaker, *Probability and Random Processes, Third Edition*. Oxford University Press, 2001.

[12] J. G. Smith, “The information capacity of amplitude- and variance-constrained scalar gaussian channels,” *Information and Control*, vol. 18, no. 3, pp. 203–219, 1971. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0019995871903469

[13] A. L. McKellips, “Simple tight bounds on capacity for the peak-limited discrete-time channel,” in *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.*, 2004, pp. 348–348.

[14] A. Thangaraj, G. Kramer, and G. BÃűcherer, “Capacity bounds for discrete-time, amplitude-constrained, additive white gaussian noise channels,” *IEEE Transactions on Information Theory*, vol. 63, no. 7, pp. 4172–4182, 2017.

[15] K. Ulm, “Simple method to calculate the confidence interval of a standardized mortality ratio (smr),” *American Journal of Epidemiology*, vol. 131, no. 2, pp. 373–375, 1990.