Prediction with a Short Memory*

Vatsal Sharan
Stanford University, USA
vsharan@stanford.edu

Sham Kakade
University of Washington, USA
sham@cs.washington.edu

Percy Liang
Stanford University, USA
pliang@cs.stanford.edu

Gregory Valiant
Stanford University, USA
valiant@stanford.edu

ABSTRACT

We consider the problem of predicting the next observation given a sequence of past observations, and consider the extent to which accurate prediction requires complex algorithms that explicitly leverage long-range dependencies. Perhaps surprisingly, our positive results show that for a broad class of sequences, there is an algorithm that predicts well on average, and bases its predictions only on the most recent few observation together with a set of simple summary statistics of the past observations. Specifically, we show that for any distribution over observations, if the mutual information between past observations and future observations is upper bounded by I, then a simple Markov model over the most recent I/ϵ observations obtains expected KL error ϵ—and hence ℓ_1 error $\sqrt{\epsilon}$—with respect to the optimal predictor that has access to the entire past and knows the data generating distribution. For a Hidden Markov Model with n hidden states, I is bounded by $\log n$, a quantity that does not depend on the mixing time, and we show that the trivial prediction algorithm based on the empirical frequencies of length $O(\log n/\epsilon)$ windows of observations achieves this error, provided the length of the sequence is $d^{\Omega(\log n/\epsilon)}$, where d is the size of the observation alphabet.

We also establish that this result cannot be improved upon, even for the class of HMMs, in the following two senses: First, for HMMs with n hidden states, a window length of $\log n/\epsilon$ is information-theoretically necessary to achieve expected KL error ϵ, or ℓ_1 error $\sqrt{\epsilon}$. Second, the $d^{\Theta(\log n/\epsilon)}$ samples required to accurately estimate the Markov model when observations are drawn from an alphabet of size d is necessary for any computationally tractable learning/prediction algorithm, assuming the hardness of strongly refuting a certain class of CSPs.

*A full version of this paper is available at https://arxiv.org/abs/1612.02526. Vatsal, and Gregory's contributions were supported in part by NSF Award CCF-1704417 and by ONR Award N00014-17-1-2562. Sham's contributions were supported in part by NSF Award CCF-1703574.

STOC'18, June 25–29, 2018, Los Angeles, CA, USA

ACM ISBN 978-1-4503-5559-9/18/06…$15.00
https://doi.org/10.1145/3188745.3188954

CCS CONCEPTS

• **Theory of computation → Machine learning theory**; *Computational complexity and cryptography*; Random walks and Markov chains;

KEYWORDS

Sequential prediction, Hidden Markov Models

ACM Reference Format:
Vatsal Sharan, Sham Kakade, Percy Liang, and Gregory Valiant. 2018. Prediction with a Short Memory. In *Proceedings of 50th Annual ACM SIGACT Symposium on the Theory of Computing (STOC'18).* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3188745.3188954

1 MEMORY, MODELING, AND PREDICTION

We consider the problem of predicting the next observation x_t given a sequence of past observations, $x_1, x_2, \ldots, x_{t-1}$, which could have complex and long-range dependencies. This *sequential prediction* problem is one of the most basic learning tasks and is encountered throughout natural language modeling, speech synthesis, financial forecasting, and a number of other domains that have a sequential or chronological element. The abstract problem has received much attention over the last half century from multiple communities including TCS, machine learning, and coding theory. The fundamental question is: *How do we consolidate and reference memories about the past in order to effectively predict the future?*

Given the immense practical importance of this prediction problem, there has been an enormous effort to explore different algorithms for storing and referencing information about the sequence, which have led to the development of several popular models such as n-gram models and Hidden Markov Models (HMMs). Recently, there has been significant interest in *recurrent neural networks* (RNNs) [1]—which encode the past as a real vector of fixed length that is updated after every observation—and specific classes of such networks, such as Long Short-Term Memory (LSTM) networks [2, 3]. Other recently popular models that have explicit notions of memory include neural Turing machines [4], memory networks [5], differentiable neural computers [6], attention-based models [7, 8], etc. These models have been quite successful (see e.g. [9, 10]); nevertheless, consistently learning long-range dependencies, in settings such as natural language, remains an extremely active area of research.

In parallel to these efforts to design systems that explicitly use memory, there has been much effort from the neuroscience community to understand how humans and animals are able to make

accurate predictions about their environment. Many of these efforts also attempt to understand the computational mechanisms behind the formation of memories (memory "consolidation") and retrieval [11–13].

Despite the long history of studying sequential prediction, many fundamental questions remain:

- How much memory is necessary to accurately predict future observations, and what properties of the underlying sequence determine this requirement?
- Must one remember significant information about the distant past or is a short-term memory sufficient?
- What is the computational complexity of accurate prediction?
- How do answers to the above questions depend on the metric that is used to evaluate prediction accuracy?

Aside from the intrinsic theoretical value of these questions, their answers could serve to guide the construction of effective practical prediction systems, as well as informing the discussion of the computational machinery of cognition and prediction/learning in nature.

In this work, we provide insights into the first three questions. We begin by establishing the following proposition, which addresses the first two questions with respect to the pervasively used metric of *average* prediction error:

Proposition 1. *Let $\mathcal{M}$ be any distribution over sequences with mutual information $I(\mathcal{M})$ between the past observations $\ldots, x_{t-2}, x_{t-1}$ and future observations $x_t, x_{t+1}, \ldots$. The best ℓ-th order Markov model, which makes predictions based only on the most recent ℓ observations, predicts the distribution of the next observation with average KL error $I(\mathcal{M})/\ell$ or average ℓ_1 error $\sqrt{I(\mathcal{M})/\ell}$, with respect to the actual conditional distribution of x_t given all past observations.*

The "best" ℓ-th order Markov model is the model which predicts x_t based on the previous ℓ observations, $x_{t-\ell}, \ldots, x_{t-1}$, according to the conditional distribution of x_t given $x_{t-\ell}, \ldots, x_{t-1}$ under the data generating distribution. If the output alphabet is of size d, then this conditional distribution can be estimated with small error given $O(d^{\ell+1})$ sequences drawn from the distribution. Without any additional assumptions on the data generating distribution beyond the bound on the mutual information, it is necessary to observe multiple sequences to make good predictions. This is because the distribution could be highly non-stationary, and have different behaviors at different times, while still having small mutual information. In some settings, such as the case where the data generating distribution corresponds to observations from an HMM, we will be able to accurately learn this "best" Markov model from a single sequence (see Theorem 1).

The intuition behind the statement and proof of this general proposition is the following: at time t, we either predict accurately and are unsurprised when x_t is revealed to us; or, if we predict poorly and are surprised by the value of x_t, then x_t must contain a significant amount of information about the history of the sequence, which can then be leveraged in our subsequent predictions of x_{t+1}, x_{t+2}, etc. In this sense, every timestep in which our prediction is 'bad', we learn some information about the past. Because the mutual information between the history of the sequence and the future is bounded by $I(\mathcal{M})$, if we were to make $I(\mathcal{M})$ consecutive bad predictions, we have captured nearly this amount of information about the history, and hence going forward, as long as the window we are using spans these observations, we should expect to predict well.

This general proposition, framed in terms of the mutual information of the past and future, has immediate implications for a number of well-studied models of sequential data, such as Hidden Markov Models (HMMs). For an HMM with n hidden states, the mutual information of the generated sequence is trivially bounded by $\log n$, which yields the following corollary to the above proposition. We state this proposition now, as it provides a helpful reference point in our discussion of the more general proposition.

Corollary 1. *Suppose observations are generated by a Hidden Markov Model with at most n hidden states. The best $\frac{\log n}{\epsilon}$-th order Markov model, which makes predictions based only on the most recent $\frac{\log n}{\epsilon}$ observations, predicts the distribution of the next observation with average KL error $\leq \epsilon$ or ℓ_1 error $\leq \sqrt{\epsilon}$, with respect to the optimal predictor that knows the underlying HMM and has access to all past observations.*

In the setting where the observations are generated according to an HMM with at most n hidden states, this "best" ℓ-th order Markov model is easy to learn given a *single* sufficiently long sequence drawn from the HMM, and corresponds to the naive "empirical" ℓ-th order Markov model (i.e. $(\ell+1)$-gram model) based on the previous observations. Specifically, this is the model that, given $x_{t-\ell}, x_{t-\ell+1}, \ldots, x_{t-1}$, outputs the observed (empirical) distribution of the observation that has followed this length ℓ sequence. (To predict what comes next in the phrase "…defer the details to the __" we look at the previous occurrences of this subsequence, and predict according to the empirical frequency of the subsequent word.) The following theorem makes this claim precise.

Theorem 1. *Suppose observations are generated by a Hidden Markov Model with at most n hidden states, and output alphabet of size d. For $\epsilon > 1/\log^{0.25} n$ there exists a window length $\ell = O(\frac{\log n}{\epsilon})$ and absolute constant c such that for any $T \geq d^{c\ell}$, if $t \in \{1, 2, \ldots, T\}$ is chosen uniformly at random, then the expected ℓ_1 distance between the true distribution of x_t given the entire history (and knowledge of the HMM), and the distribution predicted by the naive "empirical" ℓ-th order Markov model based on $x_0, \ldots, x_{t-1}$, is bounded by $\sqrt{\epsilon}$.*[1]

The above theorem states that the window length necessary to predict well is independent of the mixing time of the HMM in question, and holds even if the model does not mix. While the amount of data required to make accurate predictions using length ℓ windows scales exponentially in ℓ—corresponding to the condition in the above theorem that t is chosen uniformly between 0 and $T = d^{O(\ell)}$—our lower bounds, discussed in Section 1.3, argue that this exponential dependency is unavoidable.

[1] Theorem 1 does not have a guarantee on the average KL loss, such a guarantee is not possible as the KL loss as it can be unbounded, for example if there are rare characters which have not been observed so far.

1.1 Interpretation of Mutual Information of Past and Future

While the mutual information between the past observations and the future observations is an intuitive parameterization of the complexity of a distribution over sequences, the fact that it is the *right* quantity is a bit subtle. It is tempting to hope that this mutual information is a bound on the amount of memory that would be required to store all the information about past observations that is relevant to the distribution of future observations. This is *not* the case. Consider the following setting: Given a joint distribution over random variables X_{past} and X_{future}, suppose we wish to define a function f that maps X_{past} to a binary "advice"/memory string $f(X_{\text{past}})$, possibly of variable length, such that X_{future} is independent of X_{past}, given $f(X_{\text{past}})$. As is shown in Harsha et al. [14], there are joint distributions over $(X_{\text{past}}, X_{\text{future}})$ such that even on average, the minimum length of the advice/memory string necessary for the above task is exponential in the mutual information $I(X_{\text{past}}; X_{\text{future}})$. This setting can also be interpreted as a two-player communication game where one player generates X_{past} and the other generates X_{future} given limited communication (i.e. the ability to communicate $f(X_{\text{past}})$).[2]

Given the fact that this mutual information is not even an upper bound on the amount of memory that an optimal algorithm (computationally unbounded, and with complete knowledge of the distribution) would require, Proposition 1 might be surprising.

1.2 Implications of Proposition 1 and Corollary 1

These results show that a Markov model—a model that cannot capture long-range dependencies or structure of the data—can predict accurately on *any* data-generating distribution (even those corresponding to complex models such as RNNs), provided the order of the Markov model scales with the complexity of the distribution, as parameterized by the mutual information between the past and future. Strikingly, this parameterization is indifferent to whether the dependencies in the sequence are relatively short-range as in an HMM that mixes quickly, or very long-range as in an HMM that mixes slowly or does not mix at all. Independent of the nature of these dependencies, provided the mutual information is small, accurate prediction is possible based only on the most recent few observation. (See Figure 1 for a concrete illustration of this result in the setting of an HMM that does not mix and has long-range dependencies.)

At a time when increasingly complex models such as recurrent neural networks and neural Turing machines are in vogue, these results serve as a baseline theoretical result. They also help explain the practical success of simple Markov models such as Kneser-Ney smoothing [15, 16] for machine translation and speech recognition systems in the past. Although recent recurrent neural networks have yielded empirical gains (see e.g. [9, 10]), current models still lack the ability to consistently capture long-range dependencies.[3] In some settings, such as natural language, capturing such long-range dependencies seems crucial for achieving human-level results. Indeed, the main message of a narrative is not conveyed in any single short segment. More generally, higher-level intelligence seems to be about the ability to judiciously decide what aspects of the observation sequence are worth remembering and updating a model of the world based on these aspects.

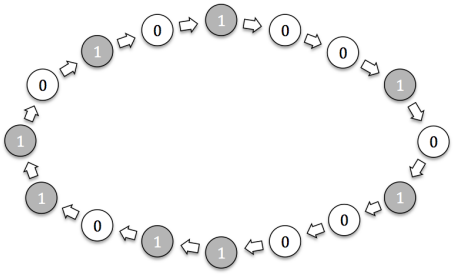

Figure 1: A depiction of a HMM on n states, that repeats a given length n binary sequence of outputs, and hence does not mix. Corollary 1 and Theorem 1 imply that accurate prediction is possible based only on short sequences of $O(\log n)$ observations.

Thus, for such settings, Proposition 1, can actually be interpreted as a kind of negative result—that *average* error is not a good metric for training and evaluating models, since models such as the Markov model which are indifferent to the time scale of the dependencies can still perform well under it as long as the number of dependencies is not too large. It is important to note that average prediction error *is* the metric that ubiquitously used in practice, both in the natural language processing domain and elsewhere. Our results suggest that a different metric might be essential to driving progress towards systems that attempt to capture long-range dependencies and leverage memory in meaningful ways. We discuss this possibility of alternate prediction metrics more in Section 1.4.

For many other settings, such as financial prediction and lower level language prediction tasks such as those used in OCR, average prediction error *is* actually a meaningful metric. For these settings, the result of Proposition 1 is extremely positive: no matter the nature of the dependencies in the financial markets, it is sufficient to learn a Markov model. As one obtains more and more data, one can learn a higher and higher order Markov model, and average prediction accuracy should continue to improve.

For these applications, the question now becomes a computational question: the naive approach to learning an ℓ-th order Markov model in a domain with an alphabet of size d might require $\Omega(d^\ell)$ space to store, and data to learn. *From a computational standpoint, is there a better algorithm? What properties of the underlying sequence imply that such models can be learned, or approximated more efficiently or with less data?*

[2] It is worth noting that if the advice/memory string s is sampled first, and then X_{past} and X_{future} are defined to be random functions of s, then the length of s *can* be related to $I(X_{\text{past}}; X_{\text{future}})$ (see [14]). This latter setting where s is generated first corresponds to allowing shared randomness in the two-player communication game; however, this is not relevant to the sequential prediction problem.

[3] One amusing example is the recent sci-fi short film *Sunspring* whose script was automatically generated by an LSTM. Locally, each sentence of the dialogue (mostly) makes sense, though there is no cohesion over longer time frames, and no overarching plot trajectory (despite the brilliant acting).

Our computational lower bounds, described below, provide some perspective on these computational considerations.

1.3 Lower bounds

Our positive results show that accurate prediction is possible via an algorithmically simple model—a Markov model that only depends on the most recent observations—which can be learned in an algorithmically straightforward fashion by simply using the empirical statistics of short sequences of examples, compiled over a sufficient amount of data. Nevertheless, the Markov model has d^ℓ parameters, and hence requires an amount of data that scales as $\Omega(d^\ell)$ to learn, where d is a bound on the size of the observation alphabet. This prompts the question of whether it is possible to learn a successful predictor based on significantly less data.

We show that, even for the special case where the data sequence is generated from an HMM over n hidden states, this is not possible in general, assuming a natural complexity-theoretic assumption. An HMM with n hidden states and an output alphabet of size d is defined via only $O(n^2 + nd)$ parameters and $O_\epsilon(n^2 + nd)$ samples *are* sufficient, from an information theoretic standpoint, to learn a model that will predict accurately. While learning an HMM is computationally hard (see e.g. [17]), this begs the question of whether accurate (average) prediction can be achieved via a computationally efficient algorithm and and an amount of data significantly less than the $d^{\Theta(\log n)}$ that the naive Markov model would require.

Our main lower bound shows that there exists a family of HMMs such that the $d^{\Omega(\log n/\epsilon)}$ sample complexity requirement is necessary for any computationally efficient algorithm that predicts accurately on average, assuming a natural complexity-theoretic assumption. Specifically, we show that this hardness holds, provided that the problem of strongly refuting a certain class of CSPs is hard, which was conjectured in Feldman et al. [18] and studied in related works Allen et al. [19] and Kothari et al. [20]. See Section 5 for a description of this class and discussion of the conjectured hardness.

Theorem 2. *Assuming the hardness of strongly refuting a certain class of CSPs, for all sufficiently large n and any $\epsilon \in (1/n^c, 0.1)$ for some fixed constant c, there exists a family of HMMs with n hidden states and an output alphabet of size d such that any algorithm that runs in time polynomial in d, namely time $f(n,\epsilon)\cdot d^{g(n,\epsilon)}$ for any functions f, g, and achieves average KL or ℓ_1 error ϵ (with respect to the optimal predictor) for a random HMM in the family must observe $d^{\Omega(\log n/\epsilon)}$ observations from the HMM.*

As the mutual information of the generated sequence of an HMM with n hidden states is bounded by $\log n$, Theorem 2 directly implies that there are families of data-generating distributions $\mathcal{M}$ with mutual information $I(\mathcal{M})$ and observations drawn from an alphabet of size d such that any computationally efficient algorithm requires $d^{\Omega(I(\mathcal{M})/\epsilon)}$ samples from $\mathcal{M}$ to achieve average error ϵ. The above bound holds when d is large compared to $\log n$ or $I(\mathcal{M})$, but a different but equally relevant regime is where the alphabet size d is small compared to the scale of dependencies in the sequence (for example, when predicting characters [21]). We show lower bounds in this regime of the same flavor as those of Theorem 2 except based on the problem of learning a noisy parity function; the (very slightly) subexponential algorithm of Blum et al. [22] for this task means that we lose at least a superconstant factor in the exponent in comparison to the positive results of Proposition 1.

Proposition 2. *Let $f(k)$ denote a lower bound on the amount of time and samples required to learn parity with noise on uniformly random k-bit inputs. For all sufficiently large n and $\epsilon \in (1/n^c, 0.1)$ for some fixed constant c, there exists a family of HMMs with n hidden states such that any algorithm that achieves average prediction error ϵ (with respect to the optimal predictor) for a random HMM in the family requires at least $f(\Omega(\log n/\epsilon))$ time or samples.*

Finally, we also establish the *information theoretic* optimality of the results of Proposition 1, in the sense that among (even computationally unbounded) prediction algorithms that predict based only on the most recent ℓ observations, an average KL prediction error of $\Omega(I(\mathcal{M})/\ell)$ and ℓ_1 error $\Omega(\sqrt{I(\mathcal{M})/\ell})$ with respect to the optimal predictor, is necessary.

Proposition 3. *There is an absolute constant $c < 1$ such that for all $0 < \epsilon < 1/4$ and sufficiently large n, there exists an HMM with n hidden states such that it is not information-theoretically possible to obtain average KL prediction error less than ϵ or ℓ_1 error less than $\sqrt{\epsilon}$ (with respect to the optimal predictor) while using only the most recent $c\log n/\epsilon$ observations to make each prediction.*

1.4 Future Directions

As mentioned above, for the settings in which capturing long-range dependencies seems essential, it is worth re-examining the choice of "average prediction error" as the metric used to train and evaluate models. One possibility, that has a more worst-case flavor, is to only evaluate the algorithm at a chosen set of time steps instead of all time steps. Hence the naive Markov model can no longer do well just by predicting well on the time steps when prediction is easy. In the context of natural language processing, learning with respect to such a metric intuitively corresponds to training a model to do well with respect to, say, a question answering task instead of a language modeling task. A fertile middle ground between average error (which gives too much reward for correctly guessing common words like "a" and "the"), and worst-case error might be a re-weighted prediction error that provides more reward for correctly guessing less common observations. It seems possible, however, that the techniques used to prove Proposition 1 can be extended to yield analogous statements for such error metrics.

In cases where average error is appropriate, given the upper bounds of Proposition 1, it is natural to consider what additional structure might be present that avoids the (conditional) computational lower bounds of Theorem 2. One possibility is a *robustness* property—for example the property that a Markov model would continue to predict well even when each observation were obscured or corrupted with some small probability. The lower bound instance rely on parity based constructions and hence are very sensitive to noise and corruptions. For learning over *product* distributions, there are well known connections between noise stability and approximation by low-degree polynomials [23, 24]. Additionally, low-degree polynomials can be learned agnostically over *arbitrary* distributions via polynomial regression [25]. It is tempting to hope that this thread could be made rigorous, by establishing a connection between natural notions of noise stability over arbitrary distributions,

and accurate low-degree polynomial approximations. Such a connection could lead to significantly better sample complexity requirements for prediction on such "robust" distributions of sequences, perhaps requiring only $\text{poly}(d, I(\mathcal{M}), 1/\epsilon)$ data. Additionally, such sample-efficient approaches to learning succinct representations of large Markov models may inform the many practical prediction systems that currently rely on Markov models.

1.5 Related Work

Parameter Estimation. It is interesting to compare using a Markov model for prediction with methods that attempt to *properly* learn an underlying model. For example, method of moments algorithms [26, 27] allow one to estimate a certain class of Hidden Markov model with polynomial sample and computational complexity. These ideas have been extended to learning neural networks [28] and input-output RNNs [29]. Using different methods, Arora et al. [30] showed how to learn certain random deep neural networks. Learning the model directly can result in better sample efficiency, and also provide insights into the structure of the data. The major drawback of these approaches is that they usually require the true data-generating distribution to be in (or extremely close to) the model family that we are learning. This is a very strong assumption that often does not hold in practice.

Universal Prediction and Information Theory. On the other end of the spectrum is the class of no-regret online learning methods which assume that the data generating distribution can even be adversarial [31]. However, the nature of these results are fundamentally different from ours: whereas we are comparing to the perfect model that can look at the infinite past, online learning methods typically compare to a fixed set of experts, which is much weaker. We note that information theoretic tools have also been employed in the online learning literature to show near-optimality of Thompson sampling with respect to a fixed set of experts in the context of online learning with prior information [32], Proposition 1 can be thought of as an analogous statement about the strong performance of Markov models with respect to the optimal predictions in the context of sequential prediction.

There is much work on sequential prediction based on KL-error from the information theory and statistics communities. The philosophy of these approaches are often more adversarial, with perspectives ranging from minimum description length [33, 34] and individual sequence settings [35], where no model of the data distribution process is assumed. Regarding worst case guarantees (where there is no data generation process), and *regret* as the notion of optimality, there is a line of work on both minimax rates and the performance of Bayesian algorithms, the latter of which has favorable guarantees in a sequential setting. Regarding minimax rates, [36] provides an exact characterization of the minimax strategy, though the applicability of this approach is often limited to settings where the number strategies available to the learner is relatively small (i.e., the normalizing constant in [36] must exist). More generally, there has been considerable work on the regret in information-theoretic and statistical settings, such as the works in [35, 37–43].

Regarding log-loss more broadly, there is considerable work on information consistency (convergence in distribution) and minimax rates with regards to statistical estimation in parametric and non-parametric families [44–49]. In some of these settings, e.g. minimax risk in parametric, i.i.d. settings, there are characterizations of the regret in terms of mutual information [45].

There is also work on universal lossless data compression algorithm, such as the celebrated Lempel-Ziv algorithm [50]. Here, the setting is rather different as it is one of coding the entire sequence (in a block setting) rather than prediction loss.

Sequential Prediction in Practice. Our work was initiated by the desire to understand the role of memory in sequential prediction, and the belief that modeling long-range dependencies is important for complex tasks such as understanding natural language. There have been many proposed models with explicit notions of memory, including recurrent neural networks [51], Long Short-Term Memory (LSTM) networks[2, 3], attention-based models [7, 8], neural Turing machines [4], memory networks [5], differentiable neural computers [6], etc. While some of these models often fail to capture long range dependencies (for example, in the case of LSTMs, it is not difficult to show that they forget the past exponentially quickly if they are "stable" [1]), the empirical performance in some settings is quite promising (see, e.g. [9, 10]).

2 PROOF SKETCH OF THEOREM 1

We provide a sketch of the proof of Theorem 1, which gives stronger guarantees than Proposition 1 but only applies to sequences generated from an HMM. The core of this proof is the following lemma that guarantees that the Markov model that knows the true marginal probabilities of all short sequences, will end up predicting well. Additionally, the bound on the expected prediction error will hold in expectation over *only* the randomness of the HMM during the short window, and with high probability over the randomness of when the window begins (our more general results hold in expectation over the randomness of when the window begins). For settings such as financial forecasting, this additional guarantee is particularly pertinent; you do not need to worry about the possibility of choosing an "unlucky" time to begin your trading regime, as long as you plan to trade for a duration that spans an entire short window. Beyond the extra strength of this result for HMMs, the proof approach is intuitive and pleasing, in comparison to the more direct information-theoretic proof of Proposition 1. We first state the lemma and sketch its proof, and then conclude the section by describing how this yields Theorem 1.

Lemma 4. *Consider an HMM with n hidden states, let the hidden state at time $s = 0$ be chosen according to an arbitrary distribution π, and denote the observation at time s by x_s. Let OPT_s denote the conditional distribution of x_s given observations $x_0, \ldots, x_{s-1}$, and knowledge of the hidden state at time $s = 0$. Let M_s denote the conditional distribution of x_s given only $x_0, \ldots, x_{s-1}$, which corresponds to the naive s-th order Markov model that knows only the joint probabilities of sequences of the first s observations. Then with probability at least $1 - 1/n^{c-1}$ over the choice of initial state, for $\ell = c \log n/\epsilon^2$, $c \geq 1$ and $\epsilon \geq 1/\log^{0.25} n$,*

$$\mathbb{E}\Big[\sum_{s=0}^{\ell-1} \|OPT_s - M_s\|_1\Big] \leq 4\epsilon\ell,$$

where the expectation is with respect to the randomness in the outputs $x_0, \ldots, x_{\ell-1}$.

The proof of the this lemma will hinge on establishing a connection between OPT_s—the Bayes optimal model that knows the HMM and the initial hidden state h_0, and at time s predicts the true distribution of x_s given $h_0, x_0, \ldots, x_{s-1}$—and the naive order s Markov model M_s that knows the joint probabilities of sequences of s observations (given that the initial state is drawn according to π), and predicts accordingly. This latter model is precisely the same as the model that knows the HMM and distribution π (but not h_0), and outputs the conditional distribution of x_s given the observations.

To relate these two models, we proceed via a martingale argument that leverages the intuition that, at each time step either $OPT_s \approx M_s$, or, if they differ significantly, we expect the sth observation x_s to contain a significant amount of information about the hidden state at time zero, h_0, which will then improve M_{s+1}. Our submartingale will precisely capture the sense that for any s where there is a significant deviation between OPT_s and M_s, we expect the probability of the initial state being h_0 conditioned on $x_0, \ldots, x_s$, to be significantly more than the probability of h_0 conditioned on $x_0, \ldots, x_{s-1}$.

More formally, let H_0^s denote the distribution of the hidden state at time 0 conditioned on $x_0, \ldots, x_s$ and let h_0 denote the true hidden state at time 0. Let $H_0^s(h_0)$ be the probability of h_0 under the distribution H_0^s. We show that the following expression is a submartingale:

$$\log\left(\frac{H_0^s(h_0)}{1 - H_0^s(h_0)}\right) - \frac{1}{2}\sum_{i=0}^{s} \|OPT_i - M_i\|_1^2.$$

The fact that this is a submartingale is not difficult: Define R_s as the conditional distribution of x_s given observations $x_0, \cdots, x_{s-1}$ and initial state drawn according to π but *not* being at hidden state h_0 at time 0. Note that M_s is a convex combination of OPT_s and R_s, hence $\|OPT_s - M_s\|_1 \leq \|OPT_s - R_s\|_1$. To verify the submartingale property, note that by Bayes Rule, the change in the LHS at any time step s is the log of the ratio of the probability of observing the output x_s according to the distribution OPT_s and the probability of x_s according to the distribution R_s. The expectation of this is the KL-divergence between OPT_s and R_s, which can be related to the ℓ_1 error using Pinsker's inequality.

At a high level, the proof will then proceed via concentration bounds (Azuma's inequality), to show that, with high probability, if the error from the first $\ell = c \log n/\epsilon^2$ timesteps is large, then $\log\left(\frac{H_0^{\ell-1}(h_0)}{1-H_0^{\ell-1}(h_0)}\right)$ is also likely to be large, in which case the posterior distribution of the hidden state, $H_0^{\ell-1}$ will be sharply peaked at the true hidden state, h_0, unless h_0 had negligible mass (less than n^{-c}) in distribution π.

There are several slight complications to this approach, including the fact that the submartingale we construct does not necessarily have nicely concentrated or bounded differences, as the first term in the submartingale could change arbitrarily. We address this by noting that the first term should not decrease too much except with tiny probability, as this corresponds to the posterior probability of the true hidden state sharply dropping. For the other direction, we can simply "clip" the deviations to prevent them from exceeding $\log n$ in any timestep, and then show that the submartingale property continues to hold despite this clipping by proving the following modified version of Pinsker's inequality:

Lemma 1. *(Modified Pinsker's inequality) For any two distributions* $\mu(x)$ *and* $\nu(x)$ *defined on* $x \in X$, *define the* C*-truncated KL divergence as* $\tilde{D}_C(\mu \parallel \nu) = \mathbb{E}_\mu\left[\log\left(\min\left\{\frac{\mu(x)}{\nu(x)}, C\right\}\right)\right]$ *for some fixed* C *such that* $\log C \geq 8$. *Then* $\tilde{D}_C(\mu \parallel \nu) \geq \frac{1}{2}\|\mu - \nu\|_1^2$.

Given Lemma 4, the proof of Theorem 1 follows relatively easily. Recall that Theorem 1 concerns the expected prediction error at a timestep $t \leftarrow \{0, 1, \ldots, d^{c\ell}\}$, based on the model M_{emp} corresponding to the empirical distribution of length ℓ windows that have occurred in $x_0, \ldots, x_t$,. The connection between the lemma and theorem is established by showing that, with high probability, M_{emp} is close to $M_{\hat{\pi}}$, where $\hat{\pi}$ denotes the empirical distribution of (unobserved) hidden states $h_0, \ldots, h_t$, and $M_{\hat{\pi}}$ is the distribution corresponding to drawing the hidden state $h_0 \leftarrow \hat{\pi}$ and then generating $x_0, x_1, \ldots, x_\ell$. We provide the full proof in Appendix 8.

3 DEFINITIONS AND NOTATION

Before proving our general Proposition 1, we first introduce the necessary notation. For any random variable X, we denote its distribution as $Pr(X)$. The mutual information between two random variables X and Y is defined as $I(X;Y) = H(Y) - H(Y|X)$ where $H(Y)$ is the entropy of Y and $H(Y|X)$ is the conditional entropy of Y given X. The conditional mutual information $I(X;Y|Z)$ is defined as:

$$I(X;Y|Z) = H(X|Z) - H(X|Y,Z) = \mathbb{E}_{x,y,z} \log \frac{Pr(X|Y,Z)}{Pr(X|Z)}$$
$$= \mathbb{E}_{y,z} D_{KL}(Pr(X|Y,Z) \parallel Pr(X|Z)),$$

where $D_{KL}(p \parallel q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$ is the KL divergence between the distributions p and q. Note that we are slightly abusing notation here as $D_{KL}(Pr(X|Y,Z) \parallel Pr(X|Z))$ should technically be $D_{KL}(Pr(X|Y=y,Z=z) \parallel Pr(X|Z=z))$. But we will ignore the assignment in the conditioning when it is clear from the context. Mutual information obeys the following chain rule: $I(X_1, X_2; Y) = I(X_1; Y) + I(X_2; Y|X_1)$.

Given a distribution over infinite sequences, $\{x_t\}$ generated by some model $\mathcal{M}$ where x_t is a random variable denoting the output at time t, we will use the shorthand x_i^j to denote the collection of random variables for the subsequence of outputs $\{x_i, \cdots, x_j\}$. The distribution of $\{x_t\}$ is *stationary* if the joint distribution of any subset of the sequence of random variables $\{x_t\}$ is invariant with respect to shifts in the time index. Hence $Pr(x_{i_1}, x_{i_2}, \cdots, x_{i_n}) = Pr(x_{i_1+l}, x_{i_2+l}, \cdots, x_{i_n+l})$ for any l if the process is stationary.

We are interested in studying how well the output x_t can be predicted by an algorithm which only looks at the past ℓ outputs. The predictor $\mathcal{A}_\ell$ maps a sequence of ℓ observations to a predicted distribution of the next observation. We denote the predictive distribution of $\mathcal{A}_\ell$ at time t as $Q_{\mathcal{A}_\ell}(x_t | x_{t-\ell}^{t-1})$. We refer to the Bayes optimal predictor using only windows of length ℓ as $\mathcal{P}_\ell$, hence the prediction of $\mathcal{P}_\ell$ at time t is $Pr(x_t | x_{t-\ell}^{t-1})$. Note that $\mathcal{P}_\ell$ is just the

naive ℓ-th order Markov predictor provided with the true distribution of the data. We denote the Bayes optimal predictor that has access to the entire history of the model as $\mathcal{P}_\infty$, the prediction of $\mathcal{P}_\infty$ at time t is $Pr(x_t|x_{-\infty}^{t-1})$. We will evaluate average performance of the predictions of $\mathcal{A}_\ell$ and $\mathcal{P}_\ell$ with respect to $\mathcal{P}_\infty$ over a long time window $[0:T-1]$.

The crucial property of the distribution that is relevant to our results is the mutual information between past and future observations. For a stochastic process $\{x_t\}$ generated by some model $\mathcal{M}$ we define the mutual information $I(\mathcal{M})$ of the model $\mathcal{M}$ as the mutual information between the past and future, averaged over the window $[0:T-1]$,

$$I(\mathcal{M}) = \lim_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} I(x_{-\infty}^{t-1}; x_t^{\infty}). \tag{3.1}$$

If the process $\{x_t\}$ is stationary, then $I(x_{-\infty}^{t-1}; x_t^{\infty})$ is the same for all time steps hence $I(\mathcal{M}) = I(x_{-\infty}^{-1}; x_0^{\infty})$. If the average does not converge and hence the limit in (3.1) does not exist, then we can define $I(\mathcal{M}, [0:T-1])$ as the mutual information for the window $[0:T-1]$, and the results hold true with $I(\mathcal{M})$ replaced by $I(\mathcal{M}, [0:T-1])$.

We now define the metrics we consider to compare the predictions of $\mathcal{P}_\ell$ and $\mathcal{A}_\ell$ with respect to $\mathcal{P}_\infty$. Let $F(P,Q)$ be some measure of distance between two predictive distributions. In this work, we consider the KL-divergence, ℓ_1 distance and the relative zero-one loss between the two distributions. The KL-divergence and ℓ_1 distance between two distributions are defined in the standard way. We define the relative zero-one loss as the difference between the zero-one loss of the optimal predictor $\mathcal{P}_\infty$ and the algorithm $\mathcal{A}_\ell$. We define the expected loss of any predictor $\mathcal{A}_\ell$ with respect to the optimal predictor $\mathcal{P}_\infty$ and a loss function F as follows:

$$\delta_F^{(t)}(\mathcal{A}_\ell) = \mathbb{E}_{x_{-\infty}^{t-1}}\Big[F(Pr(x_t|x_{-\infty}^{t-1}), Q_{\mathcal{A}_\ell}(x_t|x_{t-\ell}^{t-1}))\Big],$$

$$\delta_F(\mathcal{A}_\ell) = \lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\delta_F^{(t)}(\mathcal{A}_\ell).$$

We also define $\hat{\delta}_F^{(t)}(\mathcal{A}_\ell)$ and $\hat{\delta}_F(\mathcal{A}_\ell)$ for the algorithm $\mathcal{A}_\ell$ in the same fashion as the error in estimating $P(x_t|x_{t-\ell}^{t-1})$, the true conditional distribution of the model $\mathcal{M}$.

$$\hat{\delta}_F^{(t)}(\mathcal{A}_\ell) = \mathbb{E}_{x_{t-\ell}^{t-1}}\Big[F(Pr(x_t|x_{t-\ell}^{t-1}), Q_{\mathcal{A}_\ell}(x_t|x_{t-\ell}^{t-1}))\Big],$$

$$\hat{\delta}_F(\mathcal{A}_\ell) = \lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\hat{\delta}_F^{(t)}(\mathcal{A}_\ell).$$

4 PREDICTING WELL WITH SHORT WINDOWS

To establish our general proposition, which applies beyond the HMM setting, we provide an elementary and purely information theoretic proof.

Proposition 1. *For any data-generating distribution $\mathcal{M}$ with mutual information $I(\mathcal{M})$ between past and future observations, the best ℓ-th order Markov model $\mathcal{P}_\ell$ obtains average KL-error, $\delta_{KL}(\mathcal{P}_\ell) \le I(\mathcal{M})/\ell$ with respect to the optimal predictor with access to the infinite history. Also, any predictor $\mathcal{A}_\ell$ with $\hat{\delta}_{KL}(\mathcal{A}_\ell)$ average KL-error in estimating the joint probabilities over windows of length ℓ gets average error $\delta_{KL}(\mathcal{A}_\ell) \le I(\mathcal{M})/\ell + \hat{\delta}_{KL}(\mathcal{A}_\ell)$.*

Proof. We bound the expected error by splitting the time interval 0 to $T-1$ into blocks of length ℓ. Consider any block starting at time τ. We find the average error of the predictor from time τ to $\tau+\ell-1$ and then average across all blocks.

To begin, note that we can decompose the error as the sum of the error due to not knowing the past history beyond the most recent ℓ observations and the error in estimating the true joint distribution of the data over a ℓ length block. Consider any time t. Recall the definition of $\delta_{KL}^{(t)}(\mathcal{A}_\ell)$,

$$\begin{aligned}
\delta_{KL}^{(t)}(\mathcal{A}_\ell) &= \mathbb{E}_{x_{-\infty}^{t-1}}\Big[D_{KL}(Pr(x_t|x_{-\infty}^{t-1}) \parallel Q_{\mathcal{A}_\ell}(x_t|x_{t-\ell}^{t-1}))\Big] \\
&= \mathbb{E}_{x_{-\infty}^{t-1}}\Big[D_{KL}(Pr(x_t|x_{-\infty}^{t-1}) \parallel Pr(x_t|x_{t-\ell}^{t-1}))\Big] \\
&\quad + \mathbb{E}_{x_{-\infty}^{t-1}}\Big[D_{KL}(Pr(x_t|x_{t-\ell}^{t-1}) \parallel Q_{\mathcal{A}_\ell}(x_t|x_{t-\ell}^{t-1}))\Big] \\
&= \delta_{KL}^{(t)}(\mathcal{P}_\ell) + \hat{\delta}_{KL}^{(t)}(\mathcal{A}_\ell).
\end{aligned}$$

Therefore, $\delta_{KL}(\mathcal{A}_\ell) = \delta_{KL}(\mathcal{P}_\ell) + \hat{\delta}_{KL}(\mathcal{A}_\ell)$. It is easy to verify that $\delta_{KL}^{(t)}(\mathcal{P}_\ell) = I(x_{-\infty}^{t-\ell-1}; x_t|x_{t-\ell}^{t-1})$. This relation formalizes the intuition that the current output (x_t) has significant extra information about the past $(x_{-\infty}^{t-\ell-1})$ if we cannot predict it as well using the ℓ most recent observations $(x_{t-\ell}^{t-1})$, as can be done by using the entire past $(x_{-\infty}^{t-1})$. We will now upper bound the total error for the window $[\tau, \tau+\ell-1]$. We expand $I(x_{-\infty}^{\tau-1}; x_\tau^{\infty})$ using the chain rule,

$$I(x_{-\infty}^{\tau-1}; x_\tau^{\infty}) = \sum_{t=\tau}^{\infty} I(x_{-\infty}^{\tau-1}; x_t|x_\tau^{t-1}) \ge \sum_{t=\tau}^{\tau+\ell-1} I(x_{-\infty}^{\tau-1}; x_t|x_\tau^{t-1}).$$

Note that $I(x_{-\infty}^{\tau-1}; x_t|x_\tau^{t-1}) \ge I(x_{-\infty}^{t-\ell-1}; x_t|x_{t-\ell}^{t-1}) = \delta_{KL}^{(t)}(\mathcal{P}_\ell)$ as $t-\ell \le \tau$ and $I(X,Y;Z) \ge I(X;Z|Y)$. The proposition now follows from averaging the error across the ℓ time steps and using Eq. 3.1 to average over all blocks of length ℓ in the window $[0, T-1]$,

$$\frac{1}{\ell}\sum_{t=\tau}^{\tau+\ell-1}\delta_{KL}^{(t)}(\mathcal{P}_\ell) \le \frac{1}{\ell} I(x_{-\infty}^{\tau-1}; x_\tau^{\infty}) \implies \delta_{KL}(\mathcal{P}_\ell) \le \frac{I(\mathcal{M})}{\ell}.$$

□

Note that Proposition 1 also directly gives guarantees for the scenario where the task is to predict the distribution of the next block of outputs instead of just the next immediate output, because KL-divergence obeys the chain rule.

The following easy corollary, relating KL error to ℓ_1 error yields the following statement, which also trivially applies to zero/one loss with respect to that of the optimal predictor, as the expected relative zero/one loss at any time step is at most the ℓ_1 loss at that time step.

Corollary 2. *For any data-generating distribution $\mathcal{M}$ with mutual information $I(\mathcal{M})$ between past and future observations, the best ℓ-th order Markov model $\mathcal{P}_\ell$ obtains average ℓ_1-error $\delta_{\ell_1}(\mathcal{P}_\ell) \le \sqrt{I(\mathcal{M})/2\ell}$ with respect to the optimal predictor that has access to the infinite history. Also, any predictor $\mathcal{A}_\ell$ with $\hat{\delta}_{\ell_1}(\mathcal{A}_\ell)$ average ℓ_1-error in estimating the joint probabilities gets average prediction error $\delta_{\ell_1}(\mathcal{A}_\ell) \le \sqrt{I(\mathcal{M})/2\ell} + \hat{\delta}_{\ell_1}(\mathcal{A}_\ell)$.*

Proof. We again decompose the error as the sum of the error in estimating $\hat{P}$ and the error due to not knowing the past history using the triangle inequality.

$$\begin{aligned}\delta^{(t)}_{\ell_1}(\mathcal{A}_\ell) &= \mathbb{E}_{x^{t-1}_{-\infty}}\left[\|Pr(x_t|x^{t-1}_{-\infty}) - Q_{\mathcal{A}_\ell}(x_t|x^{t-1}_{t-\ell})\|_1\right] \\ &\le \mathbb{E}_{x^{t-1}_{-\infty}}\left[\|Pr(x_t|x^{t-1}_{-\infty}) - Pr(x_t|x^{t-1}_{t-\ell})\|_1\right] \\ &\quad + \mathbb{E}_{x^{t-1}_{-\infty}}\left[\|Pr(x_t|x^{t-1}_{t-\ell}) - Q_{\mathcal{A}_\ell}(x_t|x^{t-1}_{t-\ell})\|_1\right] \\ &= \delta^{(t)}_{\ell_1}(\mathcal{P}_\ell) + \hat{\delta}^{(t)}_{\ell_1}(\mathcal{A}_\ell)\end{aligned}$$

Therefore, $\delta_{\ell_1}(\mathcal{A}_\ell) \le \delta_{\ell_1}(\mathcal{P}_\ell) + \hat{\delta}_{\ell_1}(\mathcal{A}_\ell)$. By Pinsker's inequality and Jensen's inequality, $\delta^{(t)}_{\ell_1}(\mathcal{A}_\ell)^2 \le \delta^{(t)}_{KL}(\mathcal{A}_\ell)/2$. Using Proposition 1,

$$\delta_{KL}(\mathcal{A}_\ell) = \frac{1}{T}\sum_{t=0}^{T-1}\delta^{(t)}_{KL}(\mathcal{A}_\ell) \le \frac{I(\mathcal{M})}{\ell}$$

Therefore, using Jensen's inequality again, $\delta_{\ell_1}(\mathcal{A}_\ell) \le \sqrt{I(\mathcal{M})/2\ell}$. □

5 LOWER BOUND FOR LARGE ALPHABETS

Our lower bounds for the sample complexity in the large alphabet case leverage a class of Constraint Satisfaction Problems (CSPs) with high *complexity*. A class of (Boolean) k-CSPs is defined via a predicate—a function $P : \{0,1\}^k \to \{0,1\}$. An instance of such a k-CSP on n variables $\{x_1, \cdots, x_n\}$ is a collection of sets (clauses) of size k whose k elements consist of k variables or their negations. Such an instance is *satisfiable* if there exists an assignment to the variables $x_1, \ldots, x_n$ such that the predicate P evaluates to 1 for every clause. More generally, the *value* of an instance is the maximum, over all 2^n assignments, of the ratio of number of satisfied clauses to the total number of clauses.

Our lower bounds are based on the presumed hardness of distinguishing *random* instances of a certain class of CSP, versus instances of the CSP with *high value*. There has been much work attempting to characterize the difficulty of CSPs—one notion which we will leverage is the *complexity* of a class of CSPs, first defined in Feldman et al. [18] and studied in Allen et al. [19] and Kothari et al. [20]:

Definition 1. The *complexity* of a class of k-CSPs defined by predicate $P : \{0,1\}^k \to \{0,1\}$ is the largest r such that there exists a distribution supported on the support of P that is $(r-1)$-wise independent (i.e. "uniform"), and no such r-wise independent distribution exists.

Example 1. Both k-XOR and k-SAT are well-studied classes of k-CSPs, corresponding, respectively, to the predicates P_{XOR} that is the XOR of the k Boolean inputs, and P_{SAT} that is the OR of the inputs. These predicates both support $(k-1)$-wise uniform distributions, but not k-wise uniform distributions, hence their complexity is k. In the case of k-XOR, the uniform distribution over $\{0,1\}^k$ restricted to the support of P_{XOR} is $(k-1)$-wise uniform. The same distribution is also supported by k-SAT.

A random instance of a CSP with predicate P is an instance such that all the clauses are chosen uniformly at random (by selecting the k variables uniformly, and independently negating each variable with probability 1/2). A random instance will have value close to $\mathbb{E}[P]$, where $\mathbb{E}[P]$ is the expectation of P under the uniform distribution. In contrast, a planted instance is generated by first fixing a satisfying assignment $\boldsymbol{\sigma}$ and then sampling clauses that are satisfied, by uniformly choosing k variables, and picking their negations according to a $(r-1)$-wise independent distribution associated with the predicate. Hence a planted instance always has value 1. A noisy planted instance with planted assignment $\boldsymbol{\sigma}$ and noise level η is generated by sampling consistent clauses (as above) with probability $1-\eta$ and random clauses with probability η, hence with high probability it has value close to $1-\eta+\eta\mathbb{E}[P]$. Our hardness results are based on distinguishing whether a CSP instance is random versus has a high value (value close to $1-\eta+\eta\mathbb{E}[P]$).

As one would expect, the difficulty of distinguishing random instances from noisy planted instances, decreases as the number of sampled clauses grows. The following conjecture of Feldman et al. [18] asserts a sharp boundary on the number of clauses, below which this problem becomes computationally intractable, while remaining information theoretically easy.

Conjectured CSP Hardness [Conjecture 1] [18]: *Let Q be any distribution over k-clauses and n variables of complexity r and $0 < \eta < 1$. Any polynomial-time (randomized) algorithm that, given access to a distribution D that equals either the uniform distribution over k-clauses U_k or a (noisy) planted distribution $Q^\eta_{\boldsymbol{\sigma}} = (1-\eta)Q_{\boldsymbol{\sigma}} + \eta U_k$ for some $\boldsymbol{\sigma} \in \{0,1\}^n$ and planted distribution $Q_{\boldsymbol{\sigma}}$, decides correctly whether $D = Q^\eta_{\boldsymbol{\sigma}}$ or $D = U_k$ with probability at least 2/3 needs $\tilde{\Omega}(n^{r/2})$ clauses.*

Feldman et al. [18] proved the conjecture for the class of *statistical algorithms*.[4] Recently, Kothari et al. [20] showed that the natural Sum-of-Squares (SOS) approach requires $\tilde{\Omega}(n^{r/2})$ clauses to refute random instances of a CSP with complexity r, hence proving Conjecture 1 for any polynomial-size semidefinite programming relaxation for refutation. Note that $\tilde{\Omega}(n^{r/2})$ is tight, as Allen et al. [19] give a SOS algorithm for refuting random CSPs beyond this regime. Other recent papers such as Daniely and Shalev-Shwartz [53] and Daniely [54] have also used presumed hardness of strongly refuting random k-SAT and random k-XOR instances with a small number of clauses to derive conditional hardness for various learning problems.

A first attempt to encode a k-CSP as a sequential model is to construct a model which outputs k randomly chosen literals for the first k time steps 0 to $k-1$, and then their (noisy) predicate value for the final time step k. Clauses from the CSP correspond to samples from the model, and the algorithm would need to solve the CSP to predict the final time step k. However, as all the outputs up to the final time step are random, the trivial prediction algorithm that guesses randomly and does not try to predict the output at time k, would be near optimal. To get strong lower bounds, we will

[4] Statistical algorithms are an extension of the statistical query model. These are algorithms that do not directly access samples from the distribution but instead have access to estimates of the expectation of any bounded function of a sample, through a "statistical oracle". Feldman et al. [52] point out that almost all algorithms that work on random data also work with this limited access to samples, refer to Feldman et al. [52] for more details and examples.

output $m > 1$ functions of the k literals after k time steps, while still ensuring that all the functions remain collectively hard to invert without a large number of samples.

We use elementary results from the theory of error correcting codes to achieve this, and prove hardness due to a reduction from a specific family of CSPs to which Conjecture 1 applies. By choosing k and m carefully, we obtain the near-optimal dependence on the mutual information and error ϵ—matching the upper bounds implied by Proposition 1. We provide a short outline of the argument, followed by the detailed proof in the appendix.

5.1 Sketch of Lower Bound Construction

We construct a sequential model $\mathcal{M}$ such that making good predictions on the model requires distinguishing random instances of a k-CSP C on n variables from instances of C with a high value. The output alphabet of $\mathcal{M}$ is $\{a_i\}$ of size $2n$. We choose a mapping from the $2n$ characters $\{a_i\}$ to the n variables $\{x_i\}$ and their n negations $\{\bar{x}_i\}$. For any clause C and planted assignment $\boldsymbol{\sigma}$ to the CSP C, let $\boldsymbol{\sigma}(C)$ be the k-bit string of values assigned by $\boldsymbol{\sigma}$ to literals in C. The model $\mathcal{M}$ will output k characters from time 0 to $k-1$ chosen uniformly at random, which correspond to literals in the CSP C; hence the k outputs correspond to a clause C of the CSP. For some m (to be specified later) we will construct a binary matrix $\mathbf{A} \in \{0,1\}^{m\times k}$, which will correspond to a good error-correcting code. For the time steps k to $k+m-1$, with probability $1-\eta$ the model outputs $\mathbf{y} \in \{0,1\}^m$ where $\mathbf{y} = \mathbf{Av} \mod 2$ and $\mathbf{v} = \boldsymbol{\sigma}(C)$ with C being the clause associated with the outputs of the first k time steps. With the remaining probability, η, the model outputs m uniformly random bits. Note that the mutual information $I(\mathcal{M})$ is at most m as only the outputs from time k to $k+m-1$ can be predicted.

We claim that $\mathcal{M}$ can be simulated by an HMM with $2^m(2k+m)+m$ hidden states. This can be done as follows. For every time step from 0 to $k-1$ there will be 2^{m+1} hidden states, for a total of $k2^{m+1}$ hidden states. Each of these hidden states has two labels: the current value of the m bits of $\mathbf{y}$, and an "output label" of 0 or 1 corresponding to the output at that time step having an assignment of 0 or 1 under the planted assignment $\boldsymbol{\sigma}$. The output distribution for each of these hidden states is either of the following: if the state has an "output label" 0 then it is uniform over all the characters which have an assignment of 0 under the planted assignment $\boldsymbol{\sigma}$, similarly if the state has an "output label" 1 then it is uniform over all the characters which have an assignment of 1 under the planted assignment $\boldsymbol{\sigma}$. Note that the transition matrix for the first k time steps simply connects a state h_1 at the $(i-1)$th time step to a state h_2 at the ith time step if the value of $\mathbf{y}$ corresponding to h_1 should be updated to the value of $\mathbf{y}$ corresponding to h_2 if the output at the ith time step corresponds to the "output label" of h_2. For the time steps k through $(k+m-1)$, there are 2^m hidden states for each time step, each corresponding to a particular choice of $\mathbf{y}$. The output of an hidden state corresponding to the $(k+i)$th time step with a particular label $\mathbf{y}$ is simply the ith bit of $\mathbf{y}$. Finally, we need an additional m hidden states to output m uniform random bits from time k to $(k+m-1)$ with probability η. This accounts for a total of $k2^{m+1}+m2^m+m$ hidden states. After $k+m$ time steps the HMM transitions back to one of the starting states at time 0 and repeats. Note that the larger m is with respect to k, the higher the cost (in terms of average prediction error) of failing to correctly predict the outputs from time k to $(k+m-1)$. Tuning k and m allows us to control the number of hidden states and average error incurred by a computationally constrained predictor.

We define the CSP C in terms of a collection of predicates $P(\mathbf{y})$ for each $\mathbf{y} \in \{0,1\}^m$. While Conjecture 1 does not directly apply to C, as it is defined by a collection of predicates instead of a single one, we will later show a reduction from a related CSP C_0 defined by a single predicate for which Conjecture 1 holds. For each $\mathbf{y}$, the predicate $P(\mathbf{y})$ of C is the set of $\mathbf{v} \in \{0,1\}^k$ which satisfy $\mathbf{y} = \mathbf{Av} \mod 2$. Hence each clause has an additional label $\mathbf{y}$ which determines the satisfying assignments, and this label is just the output of our sequential model $\mathcal{M}$ from time k to $k+m-1$. Hence for any planted assignment $\boldsymbol{\sigma}$, the set of satisfying clauses C of the CSP C are all clauses such that $\mathbf{Av} = \mathbf{y} \mod 2$ where $\mathbf{y}$ is the label of the clause and $\mathbf{v} = \boldsymbol{\sigma}(C)$. We define a (noisy) planted distribution over clauses $Q^{\eta}_{\boldsymbol{\sigma}}$ by first uniformly randomly sampling a label $\mathbf{y}$, and then sampling a consistent clause with probability $(1-\eta)$, otherwise with probability η we sample a uniformly random clause. Let U_k be the uniform distribution over all k-clauses with uniformly chosen labels $\mathbf{y}$. We will show that Conjecture 1 implies that distinguishing between the distributions $Q^{\eta}_{\boldsymbol{\sigma}}$ and U_k is hard without sufficiently many clauses. This gives us the hardness results we desire for our sequential model $\mathcal{M}$: if an algorithm obtains low prediction error on the outputs from time k through $(k+m-1)$, then it can be used to distinguish between instances of the CSP C with a high value and random instances, as no algorithm obtains low prediction error on random instances. Hence hardness of strongly refuting the CSP C implies hardness of making good predictions on $\mathcal{M}$.

We now sketch the argument for why Conjecture 1 implies the hardness of strongly refuting the CSP C. We define another CSP C_0 which we show reduces to C. The predicate P of the CSP C_0 is the set of all $\mathbf{v} \in \{0,1\}^k$ such that $\mathbf{Av} = 0 \mod 2$. Hence for any planted assignment $\boldsymbol{\sigma}$, the set of satisfying clauses of the CSP C_0 are all clauses such that $\mathbf{v} = \boldsymbol{\sigma}(C)$ is in the nullspace of $\mathbf{A}$. As before, the planted distribution over clauses is uniform on all satisfying clauses with probability $(1-\eta)$, with probability η we add a uniformly random k-clause. For some $\gamma \geq 1/10$, if we can construct $\mathbf{A}$ such that the set of satisfying assignments $\mathbf{v}$ (which are the vectors in the nullspace of $\mathbf{A}$) supports a $(\gamma k-1)$-wise uniform distribution, then by Conjecture 1 any polynomial time algorithm cannot distinguish between the planted distribution and uniformly randomly chosen clauses with less than $\tilde{\Omega}(n^{\gamma k/2})$ clauses. We show that choosing a matrix $\mathbf{A}$ whose null space is $(\gamma k-1)$-wise uniform corresponds to finding a binary linear code with rate at least $1/2$ and relative distance γ, the existence of which is guaranteed by the Gilbert-Varshamov bound.

We next sketch the reduction from C_0 to C. The key idea is that the CSPs C_0 and C are defined by linear equations. If a clause $C = (x_1, x_2, \cdots, x_k)$ in C_0 is satisfied with some assignment $\mathbf{t} \in \{0,1\}^k$ to the variables in the clause then $\mathbf{At} = 0 \mod 2$. Therefore, for some $\mathbf{w} \in \{0,1\}^k$ such that $\mathbf{Aw} = \mathbf{y} \mod 2$, $\mathbf{t} + \mathbf{w} \mod 2$ satisfies

$\mathbf{A}(\mathbf{t}+\mathbf{w}) = \mathbf{y} \mod 2$. A clause $C' = (x'_1, x'_2, \cdots, x'_k)$ with assignment $\mathbf{t}+\mathbf{w} \mod 2$ to the variables can be obtained from the clause C by switching the literal $x'_i = \bar{x}_i$ if $\mathbf{w}_i = 1$ and retaining $x'_i = x_i$ if $\mathbf{w}_i = 0$. Hence for any label $\mathbf{y}$, we can efficiently convert a clause C in C_0 to a clause C' in C which has the desired label $\mathbf{y}$ and is only satisfied with a particular assignment to the variables if C in C_0 is satisfied with the same assignment to the variables. It is also not hard to ensure that we uniformly sample the consistent clause C' in C if the original clause C was a uniformly sampled consistent clause in C_0.

We provide a small example to illustrate the sequential model constructed above. Let $k = 3$, $m = 1$ and $n = 3$. Let $\mathbf{A} \in \{0,1\}^{1\times 3}$. The output alphabet of the model $\mathcal{M}$ is $\{a_i, 1 \le i \le 6\}$. The letter a_1 maps to the variable x_1, a_2 maps to $\bar{x}_1$, similarly $a_3 \to x_2, a_4 \to \bar{x}_2, a_5 \to x_3, a_6 \to \bar{x}_3$. Let $\boldsymbol{\sigma}$ be some planted assignment to $\{x_1, x_2, x_3\}$, which defines a particular model $\mathcal{M}$. If the output of the model $\mathcal{M}$ is a_1, a_3, a_6 for the first three time steps, then this corresponds to the clause with literals, $(x_1, x_2, \bar{x}_3)$. For the final time step, with probability $(1-\eta)$ the model outputs $y = \mathbf{A}\mathbf{v} \mod 2$, with $\mathbf{v} = \boldsymbol{\sigma}(C)$ for the clause $C = (x_1, x_2, \bar{x}_3)$ and planted assignment $\boldsymbol{\sigma}$, and with probability η it outputs a uniform random bit. For an algorithm to make a good prediction at the final time step, it needs to be able to distinguish if the output at the final time step is always a random bit or if it is dependent on the clause, hence it needs to distinguish random instances of the CSP from planted instances.

6 LOWER BOUND FOR SMALL ALPHABETS

Our lower bounds for the sample complexity in the binary alphabet case are based on the average case hardness of the decision version of the parity with noise problem, and the reduction is straightforward.

In the parity with noise problem on n bit inputs we are given examples $\mathbf{v} \in \{0,1\}^n$ drawn uniformly from $\{0,1\}^n$ along with their noisy labels $\langle \mathbf{s}, \mathbf{v}\rangle + \epsilon \mod 2$ where $\mathbf{s} \in \{0,1\}^n$ is the (unknown) support of the parity function, and $\epsilon \in \{0,1\}$ is the classification noise such that $Pr[\epsilon = 1] = \eta$ where $\eta < 0.05$ is the noise level.

Let $Q_{\mathbf{s}}^{\eta}$ be the distribution over examples of the parity with noise instance with $\mathbf{s}$ as the support of the parity function and η as the noise level. Let U_n be the distribution over examples and labels where each label is chosen uniformly from $\{0,1\}$ independent of the example. The strength of of our lower bounds depends on the level of hardness of parity with noise. Currently, the fastest algorithm for the problem due to Blum et al. [22] runs in time and samples $2^{n/\log n}$. We define the function $f(n)$ as follows–

Definition 2. Define $f(n)$ to be the function such that for a uniformly random support $\mathbf{s} \in \{0,1\}^n$, with probability at least $(1-1/n^2)$ over the choice of $\mathbf{s}$, any (randomized) algorithm that can distinguish between $Q_{\mathbf{s}}^{\eta}$ and U_n with success probability greater than $2/3$ over the randomness of the examples and the algorithm, requires $f(n)$ time or samples.

Our model will be the natural sequential version of the parity with noise problem, where each example is coupled with several parity bits. We denote the model as $\mathcal{M}(\mathbf{A}_{m\times n})$ for some $\mathbf{A} \in \{0,1\}^{m\times n}, m \le n/2$. From time 0 through $(n-1)$ the outputs of the model are i.i.d. and uniform on $\{0,1\}$. Let $\mathbf{v} \in \{0,1\}^n$ be the vector of outputs from time 0 to $(n-1)$. The outputs for the next m time steps are given by $\mathbf{y} = \mathbf{A}\mathbf{v} + \boldsymbol{\epsilon} \mod 2$, where $\boldsymbol{\epsilon} \in \{0,1\}^m$ is the random noise and each entry ϵ_i of $\boldsymbol{\epsilon}$ is an i.i.d random variable such that $Pr[\epsilon_i = 1] = \eta$, where η is the noise level. Note that if $\mathbf{A}$ is full row-rank, and $\mathbf{v}$ is chosen uniformly at random from $\{0,1\}^n$, the distribution of $\mathbf{y}$ is uniform on $\{0,1\}^m$. Also $I(\mathcal{M}(\mathbf{A})) \le m$ as at most the binary bits from time n to $n+m-1$ can be predicted using the past inputs. As for the large alphabet case, $\mathcal{M}(\mathbf{A}_{m\times n})$ can be simulated by an HMM with $2^m(2n+m)+m$ hidden states (see Section 5.1).

We define a set of $\mathbf{A}$ matrices, which specifies a family of sequential models. Let $\mathcal{S}$ be the set of all $(m\times n)$ matrices $\mathbf{A}$ such that the $\mathbf{A}$ is full row rank. We need this restriction as otherwise the bits of the output $\mathbf{y}$ will be dependent. We denote $\mathcal{R}$ as the family of models $\mathcal{M}(\mathbf{A})$ for $\mathbf{A} \in \mathcal{S}$. Lemma 2 shows that with high probability over the choice of $\mathbf{A}$, distinguishing outputs from the model $\mathcal{M}(\mathbf{A})$ from random examples U_n requires $f(n)$ time or examples.

Lemma 2. *Let $\boldsymbol{A}$ be chosen uniformly at random from the set $\mathcal{S}$. Then, with probability at least $(1-1/n)$ over the choice $\boldsymbol{A} \in \mathcal{S}$, any (randomized) algorithm that can distinguish the outputs from the model $\mathcal{M}(\boldsymbol{A})$ from the distribution over random examples U_n with success probability greater than $2/3$ over the randomness of the examples and the algorithm needs $f(n)$ time or examples.*

The proof of Proposition 2 follows from Lemma 2 and is similar to the proof for the large alphabet case.

7 INFORMATION THEORETIC LOWER BOUNDS

We show that *information theoretically*, windows of length $cI(\mathcal{M})/\epsilon^2$ are necessary to get expected relative zero-one loss less than ϵ. As the expected relative zero-one loss is at most the ℓ_1 loss, which can be bounded by the square of the KL-divergence, this automatically implies that our window length requirement is also tight for ℓ_1 loss and KL loss. In fact, it's very easy to show the tightness for the KL loss: choose the simple model which emits uniform random bits from time 0 to $n-1$ and repeats the bits from time 0 to $m-1$ for time n through $n+m-1$. One can then choose n, m to get the desired error ϵ and mutual information $I(\mathcal{M})$. To get a lower bound for the zero-one loss we use the probabilistic method to argue that there exists an HMM such that long windows are required to perform optimally with respect to the zero-one loss for that HMM. We now state the lower bound and sketch the proof idea.

Proposition 3. *There is an absolute constant c such that for all $0 < \epsilon < 0.5$ and sufficiently large n, there exists an HMM with n states such that it is not information theoretically possible to get average relative zero-one loss or ℓ_1 loss less than ϵ using windows of length smaller than $c\log n/\epsilon^2$, and KL loss less than ϵ using windows of length smaller than $c\log n/\epsilon$.*

We illustrate the construction in Fig. 2 and provide the high-level proof idea with respect to Fig. 2 below.

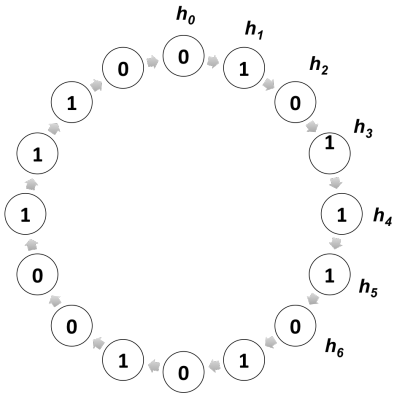

Figure 2: Lower bound construction, $n = 16$.

We want show that no predictor $\mathcal{P}$ using windows of length $\ell = 3$ can make a good prediction. The transition matrix of the HMM is a permutation and the output alphabet is binary. Each state is assigned a label which determines its output distribution. The states labeled 0 emit 0 with probability $0.5+\epsilon$ and the states labeled 1 emit 1 with probability $0.5+\epsilon$. We will randomly and uniformly choose the labels for the hidden states. Over the randomness in choosing the labels for the permutation, we will show that the expected error of the predictor $\mathcal{P}$ is large, which means that there must exist some permutation such that the predictor $\mathcal{P}$ incurs a high error. The rough proof idea is as follows. Say the Markov model is at hidden state h_2 at time 2, this is unknown to the predictor $\mathcal{P}$. The outputs for the first three time steps are (x_0, x_1, x_2). The predictor $\mathcal{P}$ only looks at the outputs from time 0 to 2 for making the prediction for time 3. We show that with high probability over the choice of labels to the hidden states and the outputs (x_0, x_1, x_2), the output (x_0, x_1, x_2) from the hidden states (h_0, h_1, h_2) is close in Hamming distance to the label of some other segment of hidden states, say (h_4, h_5, h_6). Hence any predictor using only the past 3 outputs cannot distinguish whether the string (x_0, x_1, x_2) was emitted by (h_0, h_1, h_2) or (h_4, h_5, h_6), and hence cannot make a good prediction for time 3 (we actually need to show that there are many segments like (h_4, h_5, h_6) whose label is close to (x_0, x_1, x_2)). The proof proceeds via simple concentration bounds.

8 PROOF OF THEOREM 1

Theorem 1. *Suppose observations are generated by a Hidden Markov Model with at most n hidden states, and output alphabet of size d. For $\epsilon > 1/\log^{0.25} n$ there exists a window length $\ell = O(\frac{\log n}{\epsilon})$ and absolute constant c such that for any $T \geq d^{c\ell}$, if $t \in \{1, 2, \ldots, T\}$ is chosen uniformly at random, then the expected ℓ_1 distance between the true distribution of x_t given the entire history (and knowledge of the HMM), and the distribution predicted by the naive "empirical" ℓ-th order Markov model based on $x_0, \ldots, x_{t-1}$, is bounded by $\sqrt{\epsilon}$.*

Proof. Let π_t be a distribution over hidden states such that the probability of the ith hidden state under π_t is the empirical frequency of the ith hidden state from time 1 to $t-1$ normalized by $(t-1)$. For $0 \leq s \leq \ell - 1$, consider the predictor $\mathcal{P}_t$ which makes a prediction for the distribution of observation x_{t+s} given observations $x_t, \ldots, x_{t+s-1}$ based on the true distribution of x_t under the HMM, conditioned on the observations $x_t, \ldots, x_{t+s-1}$ and the distribution of the hidden state at time t being π_t. We will show that in expectation over t, $\mathcal{P}_t$ gets small error averaged across the time steps $0 \leq s \leq \ell - 1$, with respect to the optimal prediction of x_{t+s} with knowledge of the true hidden state h_t at time t. In order to show this, we need to first establish that the true hidden state h_t at time t does not have very small probability under π_t, with high probability over the choice of t.

Lemma 3. *With probability $1-2/n$ over the choice of $t \in \{1, \ldots, T\}$, the hidden state h_t at time t has probability at least $1/n^3$ under π_t.*

Proof. Consider the ordered set $\mathcal{S}_i$ of time indices t where the hidden state $h_t = i$, sorted in increasing order. We first argue that picking a time step t where the hidden state h_t is a state j which occurs rarely in the sequence is not very likely. For sets corresponding to hidden states j which have probability less than $1/n^2$ under π_T, the cardinality $|\mathcal{S}_j| \leq T/n^2$. The sum of the cardinality of all such small sets is at most T/n, and hence the probability that a uniformly random $t \in \{1, \ldots, T\}$ lies in one of these sets is at most $1/n$.

Now consider the set of time indices $\mathcal{S}_i$ corresponding to some hidden state i which has probability at least $1/n^2$ under π_T. For all t which are not among the first T/n^3 time indices in this set, the hidden state i has probability at least $1/n^3$ under π_t. We will refer to the first T/n^3 time indices in any set S_i as the "bad" time steps for the hidden state i. Note that the fraction of the "bad" time steps corresponding to any hidden state which has probability at least $1/n^2$ under π_T is at most $1/n$, and hence the total fraction of these "bad" time steps across all hidden states is at most $1/n$. Therefore using a union bound, with failure probability $2/n$, the hidden state h_t at time t has probability at least $1/n^3$ under π_t. □

Consider any time index t, for simplicity assume $t = 0$, and let OPT_s denote the conditional distribution of x_s given observations $x_0, \ldots, x_{s-1}$, and knowledge of the hidden state at time $s = 0$. Let M_s denote the conditional distribution of x_s given only $x_0, \ldots, x_{s-1}$, given that the hidden state at time 0 has the distribution π_0.

Lemma 4. *For $\epsilon > 1/n$, if the true hidden state at time 0 has probability at least $1/n^c$ under π_0, then for $\ell = c \log n/\epsilon^2$,*

$$\mathbb{E}\Big[\frac{1}{\ell}\sum_{s=0}^{\ell-1} \|OPT_s - M_s\|_1\Big] \leq 4\epsilon,$$

where the expectation is with respect to the randomness in the outputs from time 0 to $\ell - 1$.

By Lemma 3, for a randomly chosen $t \in \{1, \ldots, T\}$ the probability that the hidden state i at time 0 has probability less than $1/n^3$ in the prior distribution π_t is at most $2/n$. As the ℓ_1 error at any time step can be at most 2, using Lemma 4, the expected average error of the predictor $\mathcal{P}_t$ across all t is at most $4\epsilon + 4/n \leq 8\epsilon$ for $\ell = 3\log n/\epsilon^2$.

Now consider the predictor $\hat{\mathcal{P}}_t$ which for $0 \le s \le \ell - 1$ predicts x_{t+s} given $x_t, \ldots, x_{t+s-1}$ according to the empirical distribution of x_{t+s} given $x_t, \ldots, x_{t+s-1}$, based on the observations up to time t. We will now argue that the predictions of $\hat{\mathcal{P}}_t$ are close in expectation to the predictions of $\mathcal{P}_t$. Recall that prediction of $\mathcal{P}_t$ at time $t+s$ is the true distribution of x_t under the HMM, conditioned on the observations $x_t, \ldots, x_{t+s-1}$ and the distribution of the hidden state at time t being drawn from π_t. For any $s < \ell$, let P_1 refer to the prediction of $\hat{\mathcal{P}}_t$ at time $t+s$ and P_2 refer to the prediction of $\mathcal{P}_t$ at time $t+s$. We will show that $\|P_1 - P_2\|_1$ is small in expectation over t.

We do this using a martingale concentration argument. Consider any string r of length s. Let $Q_1(r)$ be the empirical probability of the string r up to time t and $Q_2(r)$ be the true probability of the string r given that the hidden state at time t is distributed as π_t. Our aim is to show that $|Q_1(r) - Q_2(r)|$ is small. Define the random variable

$$Y_\tau = Pr[[x_\tau : x_{\tau+s-1}] = r | h_\tau] - I([x_\tau : x_{\tau+s-1}] = r),$$

where I denotes the indicator function and Y_0 is defined to be 0. We claim that $Z_\tau = \sum_{i=0}^{\tau} Y_i$ is a martingale with respect to the filtration $\{\phi\}, \{h_1\}, \{h_2, x_1\}, \{h_3, x_2\}, \ldots, \{h_{t+1}, x_t\}$. To verify, note that,

$$\begin{aligned}\mathbb{E}[Y_\tau | \{h_1\}, \{h_2, x_1\}, \ldots, \{h_\tau, x_{\tau-1}\}] &= Pr[[x_\tau : x_{\tau+s-1}] = r | h_\tau] \\ &\quad - E[I([x_\tau : x_{\tau+s-1}] = r) | \{h_1\}, \{h_2, x_1\}, \ldots, \{x_{\tau-1}, h_\tau\}] \\ &= Pr[[x_\tau : x_{\tau+s-1}] = r | h_\tau] - E[I([x_\tau : x_{\tau+s-1}] = r) | h_\tau] = 0.\end{aligned}$$

Therefore $\mathbb{E}[Z_\tau | \{h_1\}, \{h_2, x_1\}, \ldots, \{h_\tau, x_{\tau-1}\}] = Z_{\tau-1}$, and hence Z_τ is a martingale. Also, note that $|Z_\tau - Z_{\tau-1}| \le 1$ as $0 \le Pr[[x_\tau : x_{\tau+s-1}] = r | h_\tau] \le 1$ and $0 \le I([x_\tau : x_{\tau+s-1}] = r) \le 1$. Hence using Azuma's inequality (Lemma 8),

$$Pr[|Z_{t-s}| \ge K] \le 2e^{-K^2/(2t)}.$$

Note that $Z_{t-s}/(t-s) = Q_2(r) - Q_1(r)$. By Azuma's inequality and doing a union bound over all $d^s \le d^\ell$ strings r of length s, for $c \ge 4$ and $t \ge T/n^2 = d^{c\ell}/n^2 \ge d^{c\ell/2}$, we have $\|Q_1 - Q_2\|_1 \le 1/d^{c\ell/20}$ with failure probability at most $2d^\ell e^{-\sqrt{t}/2} \le 1/n^2$. Similarly, for all strings of length $s+1$, the estimated probability of the string has error at most $1/d^{c\ell/20}$ with failure probability $1/n^2$. As the conditional distribution of x_{t+s} given observations $x_t, \ldots, x_{t+s-1}$ is the ratio of the joint distributions of $\{x_t, \ldots, x_{t+s-1}, x_{t+s}\}$ and $\{x_t, \ldots, x_{t+s-1}\}$, therefore as long as the empirical distributions of the length s and length $s+1$ strings are estimated with error at most $1/d^{c\ell/20}$ and the string $\{x_t, \ldots, x_{t+s-1}\}$ has probability at least $1/d^{c\ell/40}$, the conditional distributions P_1 and P_2 satisfy $\|P_1 - P_2\|_1 \le 1/n^2$. By a union bound over all $d^s \le d^\ell$ strings and for $c \ge 100$, the total probability mass on strings which occur with probability less than $1/d^{c\ell/40}$ is at most $1/d^{c\ell/50} \le 1/n^2$ for $c \ge 100$. Therefore $\|P_1 - P_2\|_1 \le 1/n^2$ with overall failure probability $3/n^2$, hence the expected ℓ_1 distance between P_1 and P_2 is at most $1/n$.

By using the triangle inequality and the fact that the expected average error of $\mathcal{P}_t$ is at most 8ϵ for $\ell = 3\log n/\epsilon^2$, it follows that the expected average error of $\hat{\mathcal{P}}_t$ is at most $8\epsilon + 1/n \le 7\epsilon$. Note that the expected average error of $\hat{\mathcal{P}}_t$ is the average of the expected errors of the empirical s-th order Markov models for $0 \le s \le \ell - 1$. Hence for $\ell = 3\log n/\epsilon^2$ there must exist at least some $s < \ell$ such that the s-th order Markov model gets expected ℓ_1 error at most 9ϵ.

8.1 Proof of Lemma 4

Let the prior for the distribution of the hidden states at time 0 be π_0. Let the true hidden state h_0 at time 0 be 1 without loss of generality. We refer to the output at time t by x_s. Let $H_0^s(i) = Pr[h_0 = i | x_0^s]$ be the posterior probability of the ith hidden state at time 0 after seeing the observations x_0^s up to time t and having the prior π_0 on the distribution of the hidden states at time 0. Let $u_s = H_0^s(1)$ and $v_s = 1 - u_s$. Define $P_i^s(j) = Pr[x_s = j | x_0^{s-1}, h_0 = i]$ as the distribution of the output at time t conditioned on the hidden state at time 0 being i and observations x_0^{s-1}. Note that $OPT_s = P_1^s$. As before, define R_s as the conditional distribution of x_s given observations $x_0, \cdots, x_{s-1}$ and initial distribution π but *not* being at hidden state h_0 at time 0 i.e. $R_s = (1/v_s)\sum_{i=2}^n H_0^s(i) P_i^s$. Note that M_s is a convex combination of OPT_s and R_s, i.e. $M_s = u_s OPT_s + v_s R_s$. Hence $\|OPT_s - M_s\|_1 \le \|OPT_s - R_s\|_1$. Define $\delta_s = \|OPT_s - M_s\|_1$.

Our proof relies on a martingale concentration argument, and in order to ensure that our martingale has bounded differences we will ignore outputs which cause a significant drop in the posterior of the true hidden state at time 0. Let B be the set of all outputs j at some time t such that $\frac{OPT_s(j)}{R_s(j)} \le \frac{\epsilon^4}{\log n}$. Note that, $\sum_{j \in B} OPT_s(j) \le \frac{\epsilon^4 \sum_{j \in B} R_s(j)}{c \log n} \le \frac{\epsilon^4}{\log n}$. Hence by a union bound, with failure probability at most ϵ^2 any output j such that $\frac{OPT_s(j)}{R_s(j)} \le \frac{\epsilon^4}{\log n}$ is not emitted in a window of length $c\log n/\epsilon^2$. Hence we will only concern ourselves with sequences of outputs such that the output j emitted at each step satisfies $\frac{OPT_s(j)}{R_s(j)} \le \frac{\epsilon^4}{\log n}$, let the set of all such outputs be $\mathcal{S}_1$, note that $Pr(x_0^s \notin \mathcal{S}_1) \le \epsilon^2$. Let $\mathbb{E}_{\mathcal{S}_1}[X]$ be the expectation of any random variable X conditioned on the output sequence being in the set $\mathcal{S}_1$.

Consider the sequence of random variables $X_s = \log u_s - \log v_s$ for $s \in [-1, \ell - 1]$. Let $X_{-1} = \log(\pi_1) - \log(1 - \pi_1)$. Let $\Delta_{s+1} = X_{s+1} - X_s$ be the change in X_s on seeing the output x_{s+1} at time $s+1$. Let the output at time $s+1$ be j. We will first find an expression for Δ_{s+1}. The posterior probabilities after seeing the $(s+1)$th output get updated according to Bayes rule,

$$\begin{aligned}H_0^{s+1}(1) &= Pr[h_0 = 1 | x_0^s, x[s+1] = j] \\ &= \frac{Pr[h_0 = 1 | x_0^s] Pr[x[s+1] = j | h_0 = 1, x_0^s]}{Pr[x[s+1] = j | x_0^s]} \\ \implies u_{s+1} &= \frac{u_s OPT_{s+1}(j)}{Pr[x[s+1] = j | x_0^s]}.\end{aligned}$$

Let $Pr[x[s+1] = j | x_0^s] = d_j$. Note that $H_0^{s+1}(i) = H_0^s(i) P_i^{s+1}(j)/d_j$ if the output at time $s+1$ is j. We can write,

$$\begin{aligned}R_{s+1} &= \Big(\sum_{i=2}^n H_0^s(i) P_i^{s+1}\Big)/v_s \\ v_{s+1} &= \sum_{i=2}^n H_0^{s+1}(i) = \Big(\sum_{i=2}^n H_0^s(i) P_i^{s+1}(j)\Big)/d_j \\ &= v_s R_{s+1}(j)/d_j.\end{aligned}$$

Therefore we can write Δ_{s+1} and its expectation $\mathbb{E}[\Delta_{s+1}]$ as,

$$\Delta_{s+1} = \log \frac{OPT_{s+1}(j)}{R_{s+1}(j)}$$

$$\implies \mathbb{E}[\Delta_{s+1}] = \sum_j OPT_{s+1}(j) \log \frac{OPT_{s+1}(j)}{R_{s+1}(j)} = D(OPT_{s+1} \parallel R_{s+1}).$$

We define $\tilde{\Delta}_{s+1}$ as $\tilde{\Delta}_{s+1} := \min\{\Delta_{s+1}, \log\log n\}$ to keep martingale differences bounded. $\mathbb{E}[\tilde{\Delta}_{s+1}]$ then equals a truncated version of the KL-divergence which we define as follows.

Definition 3. For any two distributions $\mu(x)$ and $\nu(x)$, define the truncated KL-divergence as $\tilde{D}_C(\mu \parallel \nu) = \mathbb{E}\left[\log\left(\min\left\{\mu(x)/\nu(x), C\right\}\right)\right]$ for some fixed C.

We are now ready to define our martingale. Consider the sequence of random variables $\tilde{X}_s := \tilde{X}_{s-1} + \tilde{\Delta}_s$ for $t \in [0, \ell-1]$, with $\tilde{X}_{-1} := X_{-1}$. Define $\tilde{Z}_s := \sum_{s=1}^{n}\left(\tilde{X}_s - \tilde{X}_{s-1} - \delta_s^2/2\right)$. Note that $\Delta_s \geq \tilde{\Delta}_s \implies X_s \geq \tilde{X}_s$.

Lemma 5. *$\mathbb{E}_{\mathcal{S}_1}[\tilde{X}_s - \tilde{X}_{s-1}] \geq \delta_s^2/2$, where the expectation is with respect to the output at time t. Hence the sequence of random variables $\tilde{Z}_s := \sum_{i=0}^{s}\left(\tilde{X}_s - \tilde{X}_{s-1} - \delta_s^2/2\right)$ is a submartingale with respect to the outputs.*

Proof. By definition $\tilde{X}_s - \tilde{X}_{s-1} = \tilde{\Delta}_s$ and $\mathbb{E}[\tilde{\Delta}_s] = \tilde{D}_C(OPT_s \parallel R_s), C = \log n$. By taking an expectation with respect to only sequences $\mathcal{S}_1$ instead of all possible sequences, we are removing events which have a negative contribution to $\mathbb{E}[\tilde{\Delta}_s]$, hence

$$\mathbb{E}_{\mathcal{S}_1}[\tilde{\Delta}_s] \geq \mathbb{E}[\tilde{\Delta}_s] = \tilde{D}_C(OPT_s \parallel R_s).$$

We can now apply Lemma 6.

Lemma 6. *(Modified Pinsker's inequality) For any two distributions $\mu(x)$ and $\nu(x)$ defined on $x \in X$, define the C-truncated KL divergence as $\tilde{D}_C(\mu \parallel \nu) = \mathbb{E}_\mu\left[\log\left(\min\left\{\frac{\mu(x)}{\nu(x)}, C\right\}\right)\right]$ for some fixed C such that $\log C \geq 8$. Then $\tilde{D}_C(\mu \parallel \nu) \geq \frac{1}{2}\|\mu - \nu\|_1^2$.*

Hence $\mathbb{E}_{\mathcal{S}_1}[\tilde{\Delta}_s] \geq \frac{1}{2}\|OPT_s - R_s\|_1^2$. Hence $\mathbb{E}_{\mathcal{S}_1}[\tilde{X}_s - \tilde{X}_{s-1}] \geq \delta_s^2/2$. □

We now claim that our submartingale has bounded differences.

Lemma 7. $|\tilde{Z}_s - \tilde{Z}_{s-1}| \leq \sqrt{2}\log(c\log n/\epsilon^4)$.

Proof. Note that $(\delta_s^2 - \delta_{s-1}^2)/2$ can be at most 2. $Z_s - Z_{s-1} = \tilde{\Delta}_s$. By definition $\tilde{\Delta}_s \leq \log(\log n)$. Also, $\tilde{\Delta}_s \geq -\log(c\log n/\epsilon^4)$ as we restrict ourselves to sequences in the set $\mathcal{S}_1$. Hence $|\tilde{Z}_s - \tilde{Z}_{s-1}| \leq \log(c\log n/\epsilon^4) + 2 \leq \sqrt{2}\log(c\log n/\epsilon^4)$. □

We now apply Azuma-Hoeffding to get submartingale concentration bounds.

Lemma 8. *(Azuma-Hoeffding inequality) Let Z_i be a submartingale with $|Z_i - Z_{i-1}| \leq C$. Then $Pr[Z_s - Z_0 \leq -\lambda] \leq \exp\left(\frac{-\lambda^2}{2tC^2}\right)$*

Applying Lemma 8 we can show,

$$Pr[\tilde{Z}_{\ell-1} - \tilde{Z}_0 \leq -c\log n] \leq \exp\left(\frac{-c\log n}{4(1/\epsilon)^2\log^2(c\log n/\epsilon^4)}\right) \leq \epsilon^2, \tag{8.1}$$

for $\epsilon \geq 1/\log^{0.25} n$ and $c \geq 1$. We now bound the average error in the window 0 to $\ell - 1$. With failure probability at most ϵ^2 over the randomness in the outputs, $\tilde{Z}_{\ell-1} - \tilde{Z}_0 \geq -c\log n$ by Eq. 8.1. Let $\mathcal{S}_2$ be the set of all sequences in $\mathcal{S}_1$ which satisfy $\tilde{Z}_{\ell-1} - \tilde{Z}_0 \geq -c\log n$. Note that $X_0 = \tilde{X}_0 \geq \log(1/\pi_1)$. Consider the last point after which v_s decreases below ϵ^2 and remains below that for every subsequent step in the window. Let this point be τ, if there is no such point define τ to be $\ell - 1$. The total contribution of the error at every step after the τth step to the average error is at most a ϵ^2 term as the error after this step is at most ϵ^2. Note that $X_\tau \leq \log(1/\epsilon)^2 \implies \tilde{X}_\tau \leq \log(1/\epsilon)^2$ as $\tilde{X}_s \leq X_s$. Hence for all sequences in $\mathcal{S}_2$,

$$\tilde{X}_\tau \leq \log(1/\epsilon)^2$$

$$\implies \tilde{X}_\tau - \tilde{X}_{-1} \leq \log(1/\epsilon)^2 + \log(1/\pi_1)$$

$$\overset{(a)}{\implies} 0.5\sum_{s=0}^{\tau}\delta_s^2 \leq 2\log n + \log(1/\pi_1) + c\log n$$

$$\overset{(b)}{\implies} 0.5\sum_{s=0}^{\tau}\delta_s^2 \leq 2(c+1)\log n \leq 4c\log n$$

$$\overset{(c)}{\implies} \frac{\sum_{s=0}^{\ell-1}\delta_s^2}{c\log n/\epsilon^2} \leq 8\epsilon^2$$

$$\overset{(c)}{\implies} \frac{\sum_{s=0}^{\ell-1}\delta_s}{c\log n/\epsilon^2} \leq 3\epsilon,$$

where (a) follows by Eq. 8.1, and as $\epsilon \geq 1/n$; (b) follows as $\log(1/\pi_1) \leq c\log n$, and $c \geq 1$; (c) follows because $\log(1/\pi_1) \leq c\log n$); and (d) follows from Jensen's inequality. As the total probability of sequences outside $\mathcal{S}_2$ is at most $2\epsilon^2$, $\mathbb{E}[\sum_{s=0}^{\ell-1}\delta_s] \leq 4\epsilon$, whenever the hidden state i at time 0 has probability at least $1/n^c$ in the prior distribution π_0.

□

REFERENCES

[1] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5 (2):157–166, 1994.

[2] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[3] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with LSTM. *Neural computation*, 12(10):2451–2471, 2000.

[4] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

[5] J. Weston, S. Chopra, and A. Bordes. Memory networks. In *International Conference on Learning Representations (ICLR)*, 2015.

[6] Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476, 2016.

[7] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010, 2017.

[9] M. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1412–1421, 2015.

[10] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

[11] Zhe Chen and Matthew A Wilson. Deciphering neural codes of memory during sleep. *Trends in Neurosciences*, 2017.

[12] Zhe Chen, Andres D Grosmark, Hector Penagos, and Matthew A Wilson. Uncovering representations of sleep-associated hippocampal ensemble spike activity. *Scientific reports*, 6:32193, 2016.

[13] Matthew A Wilson, Bruce L McNaughton, et al. Reactivation of hippocampal ensemble memories during sleep. *Science*, 265(5172):676–679, 1994.

[14] Prahladh Harsha, Rahul Jain, David McAllester, and Jaikumar Radhakrishnan. The communication complexity of correlation. In *Twenty-Second Annual IEEE Conference on Computational Complexity (CCC'07)*, pages 10–23. IEEE, 2007.

[15] R. Kneser and H. Ney. Improved backing-off for m-gram language modeling. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 181–184, 1995.

[16] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. In *Association for Computational Linguistics (ACL)*, 1996.

[17] E. Mossel and S. Roch. Learning nonsingular phylogenies and hidden Markov models. In *Theory of computing*, pages 366–375, 2005.

[18] Vitaly Feldman, Will Perkins, and Santosh Vempala. On the complexity of random satisfiability problems with planted solutions. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pages 77–86. ACM, 2015.

[19] Sarah R Allen, Ryan O'Donnell, and David Witmer. How to refute a random CSP. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 689–708. IEEE, 2015.

[20] Pravesh K Kothari, Ryuhei Mori, Ryan O'Donnell, and David Witmer. Sum of squares lower bounds for refuting any CSP. *arXiv preprint arXiv:1701.04521*, 2017.

[21] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush. Character-aware neural language models. *arXiv preprint arXiv:1508.06615*, 2015.

[22] Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *Journal of the ACM (JACM)*, 50 (4):506–519, 2003.

[23] Ryan O'Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014.

[24] Eric Blais, Ryan OâĂŹDonnell, and Karl Wimmer. Polynomial regression under arbitrary product distributions. *Machine learning*, 80(2-3):273–294, 2010.

[25] Adam Tauman Kalai, Adam R Klivans, Yishay Mansour, and Rocco A Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008.

[26] D. Hsu, S. M. Kakade, and T. Zhang. A spectral algorithm for learning hidden Markov models. In *Conference on Learning Theory (COLT)*, 2009.

[27] A. Anandkumar, D. Hsu, and S. M. Kakade. A method of moments for mixture models and hidden Markov models. In *Conference on Learning Theory (COLT)*, 2012.

[28] H. Sedghi and A. Anandkumar. Training input-output recurrent neural networks through spectral methods. *arXiv preprint arXiv:1603.00954*, 2016.

[29] M. Janzamin, H. Sedghi, and A. Anandkumar. Beating the perils of non-convexity: Guaranteed training of neural networks using tensor methods. *arXiv preprint arXiv:1506.08473*, 2015.

[30] S. Arora, A. Bhaskara, R. Ge, and T. Ma. Provable bounds for learning some deep representations. In *International Conference on Machine Learning (ICML)*, pages 584–592, 2014.

[31] N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge University Press, 2006.

[32] Daniel Russo and Benjamin Van Roy. An information-theoretic analysis of thompson sampling. *The Journal of Machine Learning Research*, 17(1):2442–2471, 2016.

[33] A. Barron, J. Rissanen, and B. Yu. The minimum description length principle in coding and modeling. *IEEE Trans. Information Theory*, 44, 1998.

[34] P.D. Grunwald. A tutorial introduction to the minimum description length principle. *Advances in MDL: Theory and Applications*, 2005.

[35] A. Dawid. Statistical theory: The prequential approach. *J. Royal Statistical Society*, 1984.

[36] Y. Shtarkov. Universal sequential coding of single messages. *Problems of Information Transmission*, 23, 1987.

[37] K. S. Azoury and M. Warmuth. Relative loss bounds for on-line density estimation with the exponential family of distributions. *Machine Learning*, 43(3), 2001.

[38] D. P. Foster. Prediction in the worst case. *Annals of Statistics*, 19, 1991.

[39] M. Opper and D. Haussler. Worst case prediction over sequences under log loss. *The Mathematics of Information Coding, Extraction and Distribution*, 1998.

[40] Nicolo Cesa-Bianchi and Gabor Lugosi. Worst-case bounds for the logarithmic loss of predictors. *Machine Learning*, 43, 2001.

[41] V. Vovk. Competitive on-line statistics. *International Statistical Review*, 69, 2001.

[42] S. M. Kakade and A. Y. Ng. Online bounds for bayesian algorithms. *Proceedings of Neural Information Processing Systems*, 2004.

[43] M. W. Seeger, S. M. Kakade, and D. P. Foster. Worst-case bounds for some non-parametric bayesian methods, 2005.

[44] B. S. Clarke and A. R. Barron. Information-theoretic asymptotics of Bayes methods. *IEEE Transactions on Information Theory*, 36(3):453–471, 1990.

[45] David Haussler and Manfred Opper. Mutual information, metric entropy and cumulative relative entropy risk. *Annals Of Statistics*, 25(6):2451–2492, 1997.

[46] A. Barron. Information-theoretic characterization of Bayes performance and the choice of priors in parametric and nonparametric problems. In Bernardo, Berger, Dawid, and Smith, editors, *Bayesian Statistics 6*, pages 27–52, 1998.

[47] A. Barron, M. Schervish, and L. Wasserman. The consistency of posterior distributions in nonparametric problems. *Annals of Statistics*, 2(27):536–561, 1999.

[48] P. Diaconis and D. Freedman. On the consistency of Bayes estimates. *Annals of Statistics*, 14:1–26, 1986.

[49] T. Zhang. Learning bounds for a generalized family of Bayesian posterior distributions. *Proceedings of Neural Information Processing Systems*, 2006.

[50] J. Ziv and A. Lempel. Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 1978.

[51] D. Rumelhart, G. Hinton, and R. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–538, 1986.

[52] Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh Vempala, and Ying Xiao. Statistical algorithms and a lower bound for detecting planted cliques. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 655–664. ACM, 2013.

[53] Amit Daniely and Shai Shalev-Shwartz. Complexity theoretic limitations on learning DNF's. In *29th Annual Conference on Learning Theory*, pages 815–830, 2016.

[54] Amit Daniely. Complexity theoretic limitations on learning halfspaces. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing*, pages 105–117. ACM, 2016.