Performance Bounds for Active Binary Testing with Information Maximization

Aditya Chattopadhyay [1] **Benjamin D. Haeffele** [1] **René Vidal** [2] **Donald Geman** [1]

Abstract

In many applications like experimental design, group testing, and medical diagnosis, the state of a random variable Y is revealed by successively observing the outcomes of binary tests about Y. New tests are selected adaptively based on the history of outcomes observed so far. If the number of states of Y is finite, the process ends when Y can be predicted with a desired level of confidence or all available tests have been used. Finding the strategy that minimizes the expected number of tests needed to predict Y is virtually impossible in most real applications. Therefore, the commonly used strategy is the greedy heuristic of Information Maximization (InfoMax) that selects tests sequentially in order of information gain. Despite its widespread use, existing guarantees on its performance are often vacuous when compared to its empirical efficiency. In this paper, for the first time to the best of our knowledge, we establish tight non-vacuous bounds on InfoMax's performance. Our analysis is based on the assumption that at any iteration of the greedy strategy, there is always a binary test available whose conditional probability of being 'true', given the history, is within δ units of one-half. This assumption is motivated by practical applications where the available set of tests often satisfies this property for modest values of δ, say, $0.1 \leq \delta \leq 0.4$. Specifically, we analyze two distinct scenarios: (i) all tests are functions of Y, and (ii) test outcomes are corrupted by a binary symmetric channel. For both cases, our bounds guarantee the near-optimal performance of InfoMax for modest δ values. It requires only a small multiplicative factor of the entropy of Y, in terms of the average number of tests needed to make accurate predictions.

[1]Johns Hopkins University, USA [2]University of Pennsylvania, USA. Correspondence to: Aditya Chattopadhyay <achatto1@jh.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

1. Introduction

Many applications of machine learning in science and engineering can be posed as an *active testing* problem of sequentially carrying out tests to predict a target variable Y such that the expected number of tests needed is minimized. Perhaps the simplest example is the classical parlour game "twenty questions", where the objective might be to identify a famous person one player thinks of (the Y in this case) by asking the minimum number of questions about Y on average, where each of these questions can be viewed as a test about Y.[1] Other examples include Bayesian optimal experimental design (Lindley, 1956), sensor fault detection (Zheng et al., 2012) and medical diagnosis (Peng et al., 2018). Since computing the optimal sequence of tests for such scenarios is NP-complete in general (Hyafil & Rivest, 1976), the "greedy" heuristic of choosing tests in each iteration that reduce the uncertainty about Y the most, given the outcomes observed so far, is commonly employed in practice. More precisely, this is mathematically equivalent to choosing the test whose outcome has maximum mutual information with Y given the sequence of test outcomes observed so far and is popularly known as the Information Maximization (InfoMax) algorithm. InfoMax has found numerous uses in recent applications (Sznitman & Jedynak, 2010; Branson et al., 2014; Geman et al., 2015; Ma et al., 2018; Foster et al., 2019; Cuturi et al., 2020; He et al., 2022; Chattopadhyay et al., 2022; 2023; Covert et al., 2023). Given the natural intuition behind InfoMax, one might ask how efficient this greedy heuristic is in practice. However, despite its popularity, theoretical guarantees about the performance of the InfoMax algorithm are scarce (Chen et al., 2015).

In this paper, we analyze the InfoMax algorithm for binary tests and derive bounds on its performance. Throughout this paper, by performance, we mean the expected number of tests needed to make accurate predictions. If one has access to all possible binary functions of Y as tests, then it is known that the performance of the greedy strategy is upper bounded by $H(Y) + 1$ (Garey & Graham, 1974), where $H(Y)$ denotes the entropy of Y. This is nearly optimal since $H(Y)$ is a lower bound on the best possible performance

[1]For example, one possible question could be "Is Y still alive?"

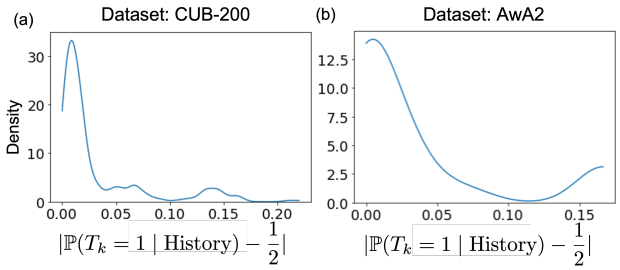

Figure 1. Distribution of the values of $|\mathbb{P}(T_k = 1 \mid \text{History}) - \frac{1}{2}|$ sampled over all iterations for all examples in the dataset, where T_k indicates the test selected at iteration k.

(Shannon, 1948). Unfortunately, for scenarios when one has access to only a restricted set of functions of Y, Loveland (1985) illustrated that it is possible to construct problems for which given a set of tests, $\mathcal{T}$, the greedy strategy requires at least $\frac{|\mathcal{Y}|}{16} \times \text{opt}(\mathcal{T}, Y)$ number of tests to identify Y, where $|\mathcal{Y}|$ is the number of values Y can take and $\text{opt}(\mathcal{T}, Y)$ is the performance of the optimal (not necessarily greedy) strategy for identifying Y given $\mathcal{T}$. Thus, as $|\mathcal{Y}|$ gets large, the greedy strategy can obtain dismal results when compared with the optimal strategy. How is it then that the greedy strategy is one of the most popular heuristics for sequential test selection?

In this paper, we argue that the often-observed competitive performance of the greedy strategy can be attributed to a property of the set of available tests $\mathcal{T}$ that we call δ-unpredictability. A set of tests is δ-unpredictable, if in every iteration of the greedy strategy either (i) the posterior over Y given the history of outcomes observed so far is sufficiently peaked, upon which Infomax terminates, or (ii) the selected test has a conditional probability of being 'true', given history, within $\frac{1}{2} \pm \delta$. For most real-world settings, the available $\mathcal{T}$ will be δ-unpredictable for $\delta \gg 0$ since there will almost never exist exact or close-to-exact bisecting tests. However, it would often be the case that for $\delta \ll \frac{1}{2}$, $\mathcal{T}$ would contain a test that satisfies condition (ii) for δ-unpredictability. This is primarily due to the availability of tests at different resolutions, with some tests being coarser (for example, tests of the form "Is $Y \in \{y_1, y_2, y_3\}$?", where y_i are different values Y can take) while others are finer (for example, tests of the form "Is $Y = y_1$?"). We support this claim with results from two real-world machine learning datasets, namely CUB-200-2011 (Wah et al., 2011) and AwA2 (Xian et al., 2018), depicted in Figure 1. In both datasets, in each iteration, there exists a test whose conditional probability of being 'true' is within $\frac{1}{2} \pm \delta$ for $\delta = 0.22$ for CUB-200-2011 and $\delta = 0.17$ for AwA2 respectively. More details in §4.2. In another example, Geman et al. (2015) employed δ-unpredictable $\mathcal{T}$ for visual scene annotation (in terms of objects in the scene, their attributes and relationships) and showed $\delta \approx 0.15$ works for their curated dataset of street scene images.

Inspired by these observations, we study the performance of the greedy strategy when $\mathcal{T}$ is δ-unpredictable for some $\delta \in [0, \frac{1}{2}]$. If $\delta = 0$, we have bisecting tests at each iteration. If we further assume the tests are functions of Y, then the set of possible values Y can take with positive probability at step k, referred to as the active set at step k, is effectively halved at each iteration depending on the test outcome. This is akin to binary search, which is known to converge in $H(Y)$ iterations (Flores & Madpis, 1971). Our contribution is to study what happens at values of δ between zero and one-half. We first study the case of oracle tests, that is, when all tests in $\mathcal{T}$ are functions of Y and bound the performance of the greedy strategy to be at most $\frac{H(Y)}{h(\frac{1}{2}+\delta)}$, which immediately improves upon bounds previously reported in literature (Garey & Graham, 1974; Loveland, 1985; Kosaraju et al., 1999; Dasgupta, 2004). Moreover, we show that this bound is tight by explicitly constructing an example for which the greedy strategy exactly achieves this bound. Building on this, we extend our analysis to incorporate noise in the test outcomes. In particular, we assume the test outcomes are corrupted by a Binary Symmetric Channel (BSC) and bound the performance of the greedy strategy to be at most $\frac{H(Y)}{h(\frac{1}{2}+\delta)-h(\alpha)}$, where α is the noise level in the BSC. In summary, our main contributions are the following.

- We first study the oracle case where tests are functions of Y. Assuming the given set of tests, $\mathcal{T}$, is δ-unpredictable for some $\delta \in [0, \frac{1}{2}]$, we prove that the greedy strategy needs at most $\frac{H(Y)}{h(\frac{1}{2}+\delta)}$ number of tests on average to identify (predict) Y. To the best of our knowledge, this is the first bound on the performance of the greedy strategy that explicitly depends on the entropy of Y. This is desirable since a lower bound on the average number of tests needed for any given $\mathcal{T}$ is given $H(Y)$. Moreover, we show that our bound is tight and cannot be improved upon.

- We then extend our analysis to the noisy case where we assume that test outcomes are corrupted via a BSC with crossover probability α (we refer to this as the noise level of the BSC in this paper). We bound the performance of the greedy strategy to be at most $\frac{H(Y)}{h(\frac{1}{2}+\delta)-h(\alpha)}$. Thus, as the noise level increases, more tests would be needed to predict Y. To the best of our knowledge, this is the first such result for the greedy strategy given noisy tests.

2. Related Work

Information Maximization (InfoMax) is a popular heuristic for sequentially selecting tests to make accurate predictions, which has been widely adopted across various fields under different names. One of the first proposals of this algorithm was in the context of optimal experimental design by (Lindley, 1956) where tests correspond to experiments one

can carry out to gather information about Y. Subsequently, this algorithm has been proposed under various names such as Probabilistic Bisection Method, (Horstein, 1963), Splitting Algorithm (Garey & Graham, 1974), Entropy Testing (Geman & Jedynak, 1996), Information Gain (for decision tree induction) (Breiman et al., 1984), Generalized Binary Search (Dasgupta, 2004), and Information Pursuit (Jahangiri et al., 2017). Inspired by its empirical success, there is a fifty year lineage of scattered work on the performance of this "greedy" strategy. We begin by reviewing works studying the oracle case, where tests are functions of Y, and conclude by mentioning recent efforts towards analyzing the more general case where test outcomes are corrupted by noise.

Oracle tests. This refers to the situation where the tests T are determined by Y, that is, the entropy $H(T \mid Y) = 0$. Shannon (1948) showed that when $\mathcal{T}$ (the set of available tests) is complete (that is, we have a test for every function of Y), the greedy strategy requires at most one test more than the optimal strategy on average. This result was extended by (Sandelius, 1961) who showed that greedy is in fact optimal when Y is uniformly distributed. Usually, for practical applications, $\mathcal{T}$ will almost always be incomplete. For example, in the popular "twenty question" parlor game involving famous people, we cannot test if Y is in every possible subset of famous people using questions about presence or absence of single human attributes like "writer", "female", "living", "French", etc. Subsequently Kosaraju et al. (1999) and Dasgupta (2004) proved that in the case of incomplete tests, the greedy strategy would require at most $\mathcal{O}\left(\ln(\frac{1}{\min_{y\in\mathcal{Y}} \mathbb{P}(Y=y)}) \times \text{opt}(\mathcal{T}, Y)\right)$ number of queries on average. Here $\text{opt}(\mathcal{T}, Y)$ is, as defined in the Introduction, the performance of the optimal strategy for identifying Y. This generic bound is often vacuous (too loose) in practice as we also show empirically in §4.2. The idea of assuming the existence of δ-unpredictable tests in each iteration of the greedy strategy was considered in earlier work (Garey & Graham, 1974; Loveland, 1985). However, their analysis technique is significantly different from ours and results in an upper bound of $\frac{\log_2 |\mathcal{Y}|}{|(\frac{1}{2}-\delta)\log_2(\frac{1}{2}-\delta)|} + \frac{1+2\delta}{1-2\delta}$, which is larger (i.e., looser) than ours. See §4.2 for an extended discussion comparing these bounds with our proposed bound.

Noisy tests. This refers to the situation where the tests T are not determined by Y, that is, the entropy $H(T \mid Y)$ is positive. Unlike the oracle case, the performance of the greedy strategy in this case is sparsely explored. It is known that InfoMax is optimal in the restricted case where $Y \in \mathbb{R}$ and $\mathcal{T}$ is a set of noisy indicator functions for all possible finite unions of intervals along the real line (Jedynak et al., 2012). Subsequent works (Tsiligkaridis et al., 2014; Chung et al., 2017) focus on designing the set of noisy tests such that Infomax (and related adaptive/non-adaptive testing strategies) is optimal . However, often in practice the set of available tests are fixed *a priori* and come from domain knowledge. For example, in medical diagnosis, the set of questions is typically about the symptoms of the patient. In such situations, it will be very unlikely that one will have access to the "optimal" test in each iteration. More general results are obtained by reducing the noisy case to the oracle case. For instance, (Nowak, 2008) assumed that the tests are "repeatable", that is, any given test can be independently replicated any number of times to obtain the true outcome (de-noise) with high probability. Thus, by repeating the same test multiple times, its outcome can be made deterministic given Y (with high confidence) and the results discussed for the oracle case apply with an additional cost for repeating the test. However, this is not very realistic since in practice we rarely have access to "repeatable" tests. Golovin et al. (2010) analyzed greedy active learning algorithms in the presence of noise by considering the tests to be functions of Y *and* some noise variable η with known joint distribution $\mathbb{P}(Y, \eta)$, and thereafter applied the bounds known from the oracle case. Finally, (Chen et al., 2015) explored the near-optimality of information maximization for the more practical scenario where noise is persistent, that is, tests are not "repeatable". Compared to our work, (Chen et al., 2015) studies the setting "What is the maximum amount of mutual information one can obtain about Y by carrying out k tests following the greedy strategy?", whereas we are interested in bounding the mean number of tests required to make an accurate prediction about Y.

3. Problem Setting and Preliminaries

As is common convention, we will use capital letters for random variables and lowercase letters for their realizations. We will use the symbol $\mathbb{P}(\mathcal{E})$ to denote the probability of event $\mathcal{E}$. Moreover, we will often refer to the Information Maximization (InfoMax) algorithm simply as the *greedy strategy*.

Information maximization. InfoMax is a greedy strategy for selecting tests sequentially in order of information gain. More formally, let Y be a discrete random variable taking values in $\mathcal{Y}$ and let $\mathcal{T}$ be a given finite set of available tests, whose outcomes are informative about the value of Y. All random variables ($\mathcal{T}$ and Y) are defined on a common sample space Ω. Given this setup, for any collection of tests (binary, noisy or otherwise), the InfoMax algorithm proceeds iteratively as follows:

$$T_1 = \arg\max_{T\in\mathcal{T}} I(T; Y); \quad T_{k+1} = \arg\max_{T\in\mathcal{T}} I(T; Y \mid \mathcal{A}(t_{1:k})). \tag{1}$$

Here $T_{k+1} \in \mathcal{T}$ refers to the new test selected by InfoMax at step $k+1$, based on the history of outcomes to previously asked tests (denoted as $t_{1:k}$), and $t_{k+1} \in \{0, 1\}$ indicates the corresponding outcome of the test asked in iteration

$k+1$. The conditioning event $\mathcal{A}(t_{1:k})$ is defined as the event $\{\omega \in \Omega : T_i(\omega) = t_i : i \in \{1, 2, \ldots, k\}\}$, where t_i is the observed outcome for carrying out test T_i. We refer to these events as *active sets*. We will use the concept of active sets in our analysis of InfoMax. The algorithm terminates after L iterations if either $\max_{y \in \mathcal{Y}} \mathbb{P}(Y = y \mid \mathcal{A}(t_{1:L})) > \gamma$ (a hyper-parameter that can be interpreted as desired accuracy level) or after all tests have been carried out. Refer to Figure 4 in the appendix for a flowchart diagram illustrating the InfoMax algorithm. Having described the InfoMax algorithm, we next define (δ, γ)-unpredictable set of tests which encapsulates our assumption of existence of approximately bisecting sets as discussed in the Introduction.

Unpredictable set of tests. As motivated in the Introduction, there exists scenarios when the greedy strategy can perform poorly compared to the optimal strategy. This calls for some assumptions on $\mathcal{T}$ to ensure the good performance of the greedy strategy that is often observed. In this work, we assume that at each iteration of the greedy algorithm there exists a test that δ-approximately bisects the current active set. Formally,

Definition 3.1. [**(δ, γ)-unpredictable set of tests**] A set of tests $\mathcal{T}$ is said to be (δ, γ)-unpredictable[2] if it is non-empty and at any iteration $k+1$ of InfoMax (assuming there remain tests in $\mathcal{T}$ that have not yet been carried out), either

- The probability of posterior mode is greater than or equal to γ, i.e., $\max_{y \in \mathcal{Y}} \mathbb{P}(Y = y \mid \mathcal{A}(t_{1:k})) \geq \gamma$, upon which the algorithm terminates; or
- There exists a test $T_{k+1} \in \mathcal{T}$ such that,

$$\left| \mathbb{P}\left(T_{k+1} = 1 \mid \mathcal{A}(t_{1:k})\right) - \frac{1}{2} \right| \leq \delta, \tag{2}$$

  where $t_{1:k}$ denotes the history of test outcomes after k iterations.

The γ parameter controls the termination criterion for the greedy strategy. In the extreme case where we require Y to be identifiable, $\gamma = 1$. For simplicity, in such scenarios, we will drop γ from the notation and refer to such sets as δ-unpredictable set of tests, implicitly meaning that the algorithm terminates only when Y is identified or all tests in $\mathcal{T}$ have been carried out. We will further discuss the motivation for this definition and its implication on the performance of the greedy strategy in the subsequent sections.

[2]The word unpredictable comes from the fact that if a test $T' \in \mathcal{T}$ at iteration k exactly bisects the current active set, then, one cannot predict the outcome of T' based on the history of test outcomes observed up to the first $k-1$ iterations better than a random (unbiased) coin flip.

4. Performance Bounds For Oracle Tests

In this section, we analyze the performance of InfoMax when all tests in $\mathcal{T}$ are functions of Y, hence the name *oracle tests*. Throughout this section, we will denote the outcome of test T as $T(Y)$ to explicitly remind the reader that T is a function of Y. Effectively the sample space Ω (as defined in §3) can be taken to be $\mathcal{Y}$. Since the tests are not noisy, it is reasonable to expect that they collectively determine Y, that is, the value of Y is uniquely determined if we observe $\{T(Y),\ \forall T \in \mathcal{T}\}$). As a result we will drop γ from the notation and only refer to $\mathcal{T}$ as being a δ-unpredictable set of tests.

4.1. A new bound on the performance of the greedy information maximization algorithm

Relationship with entropy maximization. In the oracle case, where Y determines the test outcomes (i.e., the outcome of any test is a function of Y, $t = T(Y),\ \forall T \in \mathcal{T}$), the InfoMax algorithm as described in equation 1 is equivalent to sequentially finding the test T that achieves the maximum conditional entropy given history. Equivalently,

$$T_1 = \arg\max_{T \in \mathcal{T}} H(T); \quad T_{k+1} = \arg\max_{T \in \mathcal{T}} H(T \mid \mathcal{A}(t_{1:k})). \tag{3}$$

The equivalence of equation 3 and equation 1 can be seen by noticing that $H(T \mid Y, \mathcal{A}(t_{1:k})) = 0$ when all tests are functions of Y (Cover, 1999). Note that the active set in this case is now simply a subset of $\mathcal{Y}$, that is, $\mathcal{A}(t_{1:k}) = \{y \in \mathcal{Y} : T_i(y) = t_i : i \in \{1, 2, ..., k\}\}$.

Motivation for assuming $\mathcal{T}$ is δ-unpredictable. The motivation for assuming a given $\mathcal{T}$ is δ-unpredictable is as follows. The entropy of a binary random variable is maximized when its success probability is $p = \frac{1}{2}$. Equation 3 can be reinterpreted as sequentially selecting tests from $\mathcal{T}$ that have success probability closest to $\frac{1}{2}$ given the history of test outcomes observed so far. Specifically,

$$\begin{aligned} T_1 &= \arg\min_{T \in \mathcal{T}} \left| \mathbb{P}(T(Y) = 1) - \frac{1}{2} \right|; \text{ and} \\ T_{k+1} &= \arg\min_{T \in \mathcal{T}} \left| \mathbb{P}(T(Y) = 1 \mid \mathcal{A}(t_{1:k}) - \frac{1}{2} \right|; \end{aligned} \tag{4}$$

While it will generally not be possible to find a perfectly bisecting test, it is reasonable to assume that there exists some δ, such that at any iteration, a test can be found in $\mathcal{T}$ whose success probability, conditioned on the history of test outcomes observed so far, is within $\frac{1}{2} \pm \delta$ (motivated in §1). Note, the condition in equation 4 is equivalent to choosing the binary test whose conditional entropy is closest to 1, that is,

$$T_{k+1} = \arg\min_{T \in \mathcal{T}} \left| H(T \mid \mathcal{A}(t_{1:k})) - 1 \right|.$$

Bounding the performance of the greedy strategy. If $\mathcal{T}$ is δ-unpredictable for very small δ we can intuitively expect the number of queries needed on average to identify Y to be roughly of the order of $H(Y)$ (since we have almost bisecting tests). On the other hand, for large δ (close to $\frac{1}{2}$), any given set of tests would be δ-unpredictable (according to definition 3.1) and we would expect the number of queries needed on average to blow up. The following theorem captures this intuition and provides a bound on the expected number of tests needed by the greedy strategy as a function of both δ and the entropy of Y.

Theorem 4.1. *Fix any $\delta \in [0, \frac{1}{2}]$. Let h be the binary entropy function. Given a δ-unpredictable $\mathcal{T}$ that collectively determine Y, the average number of tests needed by the information maximization algorithm to identify Y is at most*[3]

$$B_{Oracle} := \frac{H(Y)}{h(\frac{1}{2}+\delta)}. \tag{5}$$

Proof. (Sketch only; see Appendix §A.1.1 for a complete proof.) Our result is based on the insight that given a sequence of k test outcomes, if an additional test T_{k+1} is carried out using the greedy strategy then the following two scenarios are possible: (i) The additional test will have a conditional entropy $H(T_{k+1} \mid \mathcal{A}(t_{1:k})) \geq h(\frac{1}{2}+\delta)$; or (ii) the current active set, $\mathcal{A}(t_{1:k-1})$, is a singleton since Y has already been determined and thus, $H(T_{k+1} \mid \mathcal{A}(t_{1:k-1})) = 0$. Combining these two scenarios and taking expectation over the k tests and their corresponding outcomes and then summing k from 0 to $|\mathcal{T}| - 1$ gives the desired result. □

To highlight the importance of this result, recall from coding theory that given any set of tests, the optimal strategy cannot be better than $H(Y)$, which thus serves as a lower bound for the greedy strategy given any $\mathcal{T}$. To the best of our knowledge, our result is the first one to upper bound the performance of the greedy strategy to be at most a multiplicative factor of the entropy of Y. This multiplicative factor degrades "modestly" with δ and so even for a large value of $\delta \approx 0.4$ (recall $\delta \in [0, \frac{1}{2}]$), which is far from a bisecting split, our result guarantees that the average number of tests under the greedy strategy is at most roughly twice the entropy of Y. In many practical applications, the set of tests available to the user is easily δ-unpredictable for such a large value of 0.4 which strongly reinforces the experience that *greedy works well in practice*. We illustrate a few such examples on real-world datasets in §4.2.

[3]We thank an anonymous reviewer for suggesting this bound as an improvement over the bound we had in an earlier version of the paper, which was $\frac{H(Y)}{-\log_2(\frac{1}{2}+\delta)}$.

4.2. Comparison with previous bounds

Having described our bound, we compare it with bounds previously reported in literature. The assumption of a δ-unpredictable $\mathcal{T}$ was previously considered by Garey & Graham (1974) for the case where Y is uniformly distributed, and subsequently by Loveland (1985) for any distribution on Y. Both papers get the same bound and so we compare with the bound in (Loveland, 1985), which we will refer to as the B_{Lov}. Their analysis technique is significantly different from ours and as a result they obtain a very different upper bound on the average number of queries needed to identify Y,

$$\mathrm{B}_{\text{Lov}} := \frac{\log_2 |\mathcal{Y}|}{-(\frac{1}{2}-\delta)\log_2(\frac{1}{2}-\delta)} + \frac{1+2\delta}{1-2\delta}, \tag{6}$$

where $|\mathcal{Y}|$ is the number of discrete values Y can take. Our bound in equation 5 is uniformly a tighter bound than equation 6 for all values of δ. We discuss this in detail below.

It can be easily seen (from the definition of binary entropy function) that for any value of $\delta \in [0, \frac{1}{2}]$, the denominator in the first term of B_{Lov} is smaller than the denominator in $\mathrm{B}_{\text{Oracle}}$;

$$-(\frac{1}{2}-\delta)\log_2(\frac{1}{2}-\delta) \leq h(\frac{1}{2}-\delta) = h(\frac{1}{2}+\delta).$$

The second term $\frac{1+2\delta}{1-2\delta}$ is always non-negative. Finally, the numerator $\log_2 |\mathcal{Y}|$ is always lower bounded by the entropy of Y (which is the numerator in $\mathrm{B}_{\text{Oracle}}$). Hence, our bound is always tighter than (as an upper bound on the performance of the greedy algorithm) B_{Lov}.

Next, we demonstrate on two machine learning datasets (CUB-200-2011 (Wah et al., 2011) and AwA2 (Xian et al., 2018)) that the given set of tests $\mathcal{T}$ is δ-unpredictable for modest values of δ (0.22 and 0.17 respectively) and subsequently show that our bound is closer to the true mean number of tests the greedy strategy requires on these datasets to identify Y than previously known bounds.

Experimental setup. Our examples are inspired from the classical "twenty questions" (20Q) game where one player thinks of an entity, and the goal of the other player is to guess the object correctly by asking the minimum number of questions about the object.[4] We now describe the two datasets employed in detail.

- **20Q with birds.** In our first example, we play 20Q with birds. One player thinks of a bird species Y and the other player asks questions about different visual attributes about the chosen bird in order to identify Y. For this purpose we use the CUB-200-2011 dataset. The dataset

[4]The tests in this context are the questions and their respective outcomes are the answers to the question.

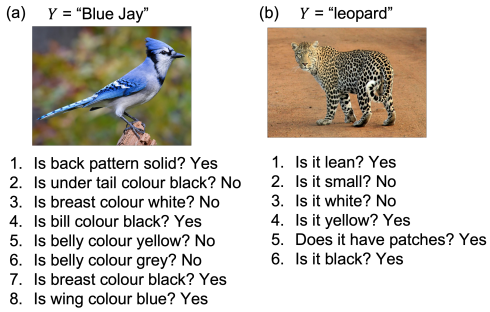

Figure 2. Example runs of the greedy strategy on the two tasks considered here; **(a)** 20Q with birds; **(b)** 20Q with animals. The test outcomes are obtained from ground truth annotations.

consists of images of 200 different bird species, each annotated with answers to 312 binary questions about visual attributes like wing colour, body shape, beak shape etc. See Appendix §B.1 for a complete list of all the binary questions used. It is reasonable to assume that given all 312 attributes, Y is determined and that every visual attribute question is a function of Y. However, the image annotations are noisy. To remedy this, in accordance with prior work (Koh et al., 2020), we modify the annotations in the following manner; if more than half the images for a particular class have value x for a certain attribute, we set the annotation for that attribute of all images from that class to x.

- **20Q with animals.** For our second example, we play 20Q with animals. For this purpose, we use the AwA2 dataset (Xian et al., 2018). The datasets consists of images of 50 different animal classes each annotated with answers to 85 binary attributes such as number of legs, skin color, eating habits, habitat etc. Appendix §B.2 for a complete list of all the binary questions used. Every attribute answer is a deterministic outcome of the label Y and together they determine Y, that is, knowing the answers to all 85 attribute questions allows for identifying Y.

Given a dataset, we construct a $\mathcal{T}$ by including a test for every binary attribute in the dataset in the form of a question about the presence or absence of that attribute. The outcome of any given test is its corresponding answer evaluated on the given sample point (obtained from the dataset annotations). We carry out information maximization to sequentially carry out tests until the class, Y, has been determined. We use the empirical probabilities in the dataset to compute all the entropic quantities required for running the greedy strategy (algorithm in equation 3). Figure 2a shows an example run of the greedy strategy for the CUB-200 dataset and Figure 2b shows an example from the AwA2 dataset.

Empirical computation of δ for a given dataset. For every class $y \in \mathcal{Y}$, for every iteration k and selected test $T_{k,y}$, we record the quantity $\delta_{k,y} := \left|P(T_{k,y} = 1 \mid t_{1:k-1}) - \frac{1}{2}\right|$. $T_{k,y}$ is the test selected by the greedy strategy in iteration k

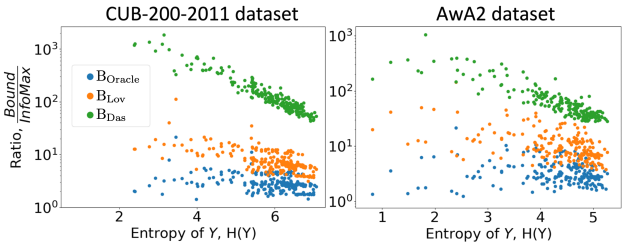

Figure 3. Comparison of the relative ratio of different bounds to the actual average number of tests needed by InfoMax, $\frac{\text{Bound}}{\text{InfoMax}}$, for different simulated prior distributions over Y.

for when the class label of the input sample point is $Y = y$. Figure 1 shows the distribution for $\delta_{k,y}$ over all iterations and labels $y \in \mathcal{Y}$ for the two datasets. Y is considered to be uniform in both datasets since they are balanced. It is clear from the figure that the given set of tests for these examples is δ-unpredictable for $\delta = 0.22$ and $\delta = 0.17$ respectively, since for this value all tests selected by the greedy strategy (in any iteration for any data-point) will satisfy equation 2.

Empirical comparison of bounds for uniform Y. We first compare the mean number of queries needed by the greedy strategy for this task with various upper bounds to evaluate their tightness in Table 1 when Y is taken to be uniform (as is in the given datasets). In addition to $\mathrm{B}_{\mathrm{Lov}}$, we also compare with with the bounds derived by Dasgupta (2004) and Kosaraju et al. (1999), which make no assumption on $\mathcal{T}$ (and differ by only a constant factor). Dasgupta's bound is given by:

$$\mathrm{B}_{\mathrm{Das}} := 4 \ln \left(\frac{1}{\min_Y p(Y)} \right) \times \mathrm{opt}(\mathcal{T}, Y), \qquad (7)$$

where $\mathrm{opt}(\mathcal{T}, Y)$ is as defined in the Introduction. Since computing $\mathrm{opt}(\mathcal{T}, Y)$ is intractable, in our comparisons we use $H(Y)$ as a proxy (recall $H(Y) \leq \mathrm{opt}(\mathcal{T}, Y)$).

Table 1. Comparison of different bounds with the empirical performance of the greedy strategy (InfoMax in column 4)

Task	δ	$H(Y)$	InfoMax	$\mathrm{B}_{\mathrm{Oracle}}$	$\mathrm{B}_{\mathrm{Lov}}$	$\mathrm{B}_{\mathrm{Das}}$
20Q with birds	0.22	7.64	7.70	**8.93**	17.44	161.86
20Q with animals	0.17	5.64	5.73	**6.17**	12.69	88.31

Notice that for both datasets, our bound is much tighter than existing bounds with $\mathrm{B}_{\mathrm{Das}}$ being extremely loose (understandably) since it makes no assumptions about $\mathcal{T}$.

Empirical comparison of bounds for non-uniform Y. We expect our bounds to be much tighter than previous bounds as the entropy of Y decreases. To test this, we took the same two datasets as before and simulated different prior distributions over Y. We did so by carrying out the following steps:

1. Sample a prior distribution over Y, denoted as $\mathbb{P}(Y)$,

by sampling from the symmetric Dirichlet distribution (the concentration parameter of the Dirichlet was sampled uniformly between $[0, 1]$).

2. Construct an augmented dataset by repeating every label $Y = y$ (in the original dataset) $\lfloor 1000\mathbb{P}(y) \rfloor$ times, where $\lfloor . \rfloor$ is the floor function to ensure an integer value and 1000 is a chosen hyper-parameter to ensure we have enough samples to accurately estimate the sampled prior $\mathbb{P}(Y)$ (obtained in the previous step).

The above two steps result in one simulation of a non-uniform prior distribution. For both datasets (CUB-200 and AwA2) we carried out 200 simulations, and present the ratio between the evaluated bound[5] and the performance of the InfoMax algorithm for each simulation as a scatter plot in Figure 3. Our results indicate that the ratio of our bound, across all simulated priors, is about an order of magnitude smaller than both $\mathrm{B}_{\mathrm{Lov}}$ and $\mathrm{B}_{\mathrm{Das}}$. Moreover, this gap increases as the entropy of Y decreases.

4.3. Is our bound tight?

In the previous subsection, we show that our proposed bound improves upon previous known bounds on the performance of the greedy strategy. Nevertheless, is it possible to get a better bound? Stated precisely, given a value δ, is it possible to obtain a tighter bound than $\mathrm{B}_{\mathrm{Oracle}}$ that holds for all $\mathcal{T}$ that is δ-unpredictable? The answer is negative since given a δ, we can always construct a discrete random variable Y and a set of tests $\mathcal{T}$ that is δ-unpredictable such that the greedy strategy takes exactly $\frac{H(Y)}{h(\frac{1}{2}+\delta)}$ number of tests to identify Y. Since Theorem 4.1 must hold for any set of tests that is δ-unpredictable this immediately implies that our bound is tight. We will now describe the construction of such an example in detail.

Given some $\delta \in [0, \frac{1}{2}]$. Take any integer $n \in \mathbb{N}$ and let Y be a random variable taking n different values with the following probability mass function for any $k \in \{1, 2, ..., n-1\}$

$$\mathbb{P}(Y = k) = (\frac{1}{2} - \delta)(\frac{1}{2} + \delta)^{k-1}.$$

The remaining mass of $(\frac{1}{2} + \delta)^{n-1}$ is placed on the last element n, that is, $\mathbb{P}(Y = n) = (\frac{1}{2} + \delta)^{n-1}$. The entropy of this distribution is (see Lemma A.1 in the Appendix for a proof.)

$$H(Y) = h(\frac{1}{2} + \delta) \sum_{i=0}^{n-2} (\frac{1}{2} + \delta)^i,$$

Given this setup, take the set of tests, $\mathcal{T}$, to be the set of all singleton tests of the form "Is $Y = y$?" for all $y \in \{1, 2, ..., n\}$. This $\mathcal{T}$ is δ-unpredictable. This is because, in every iteration k, the most informative test (that is, the test with the highest entropy conditioned on history) will be the test "Is $Y = k$?" and this test will have a conditional probability of being 'true' δ units away from one-half thus satisfying equation 2. We formally show this in the proof of Lemma A.2 in the Appendix.

The greedy strategy with this $\mathcal{T}$ will take $\frac{H(Y)}{h(\frac{1}{2}+\delta)}$ tests on average to identify Y (refer Lemma A.2) which coincides with our bound in Theorem 4.1.

5. Performance Bounds For Noisy Tests

Here, we analyze the performance of the greedy strategy when all tests in $\mathcal{T}$ are noisy, that is $\forall T \in \mathcal{T}$, the conditional entropy $H(T \mid Y) > 0$. As discussed in §2, the performance of the greedy strategy under noise is poorly understood. Unlike prior work (Nowak, 2008), our analysis does not assume that tests can be repeated any number of times to average the noise out. This is because in many applications the same test cannot be repeated again or will give the same outcome (Chen et al., 2015).[6] Instead we consider an explicit noise model for the tests and analyze the performance of the greedy strategy for that model.

Binary Symmetric Channel (BSC) Noise Model. We first study the case where test outcomes are corrupted by a BSC, which is perhaps the most well-studied and simplest model for understanding the effects of noise in communication channels. We make the following assumptions.

- For every $T \in \mathcal{T}$ there exists random variables $D_T(Y)$, which is a function of Y, and N_T such that $T = D_T(Y) \oplus N_T$. The symbol $\oplus$ denotes the Exclusive OR (XOR) operation. $D_T(Y)$ can be understood as the true outcome for test T if there was no noise. N_T is the noise variable that corrupts the test outcome.

- For every $T \in \mathcal{T}$, we assume N_T is independent of Y with prior probability $\mathbb{P}(N_T = 1) = \alpha$ for some $\alpha \in [0, \frac{1}{2}]$. Moreover, we assume all the noise variables, $\{N_T : T \in \mathcal{T}\}$, are independent and hence the noise variables are i.i.d..

We now describe our analysis of how the greedy strategy performs under this noise model.

[5] δ was computed empirically for each simulated prior distribution using the same procedure described before.

[6] Note, while we do not assume the same test can be repeated, there can be multiple tests in $\mathcal{T}$ that are (conditionally) statistically identical. For example, in the famous 20Q game let y_1 = "Queen Victoria" and y_2 = "Charles Darwin" be the only two states with non-trivial mass. Then, both tests "Is Y female?" and "Is Y a queen?" have statistically identical outcomes but are different tests.

5.1. A bound of the performance of the greedy strategy for noisy tests

In general, when noise is present in the test outcomes, InfoMax (equation 1) is not equivalent to entropy maximization (equation 3). As a result we cannot interpret the greedy strategy as selecting the test at each iteration whose success probability given the history of test outcomes observed so far is close to $\frac{1}{2}$. However, as we show in Lemma 5.1, under our noise model, we can interpret the greedy strategy as choosing the test $\hat{T}$ in each iteration whose true outcome ($D_{\hat{T}}$) has success probability (given history) closest to a half. We now state our lemma which is inspired from (Jedynak et al., 2012) where a similar result was derived for the case where $Y = \mathbb{R}$ and the tests are unions of intervals along $\mathbb{R}$.

Lemma 5.1. *Under the BSC noise model, at any iteration $k+1$, the InfoMax algorithm will pick test*

$$T_{k+1} = \arg\min_{T \in \mathcal{T}} \left| \mathbb{P}(D_T = 1 \mid \mathcal{A}(t_{1:k})) - \frac{1}{2} \right|,$$

where $\mathcal{A}(t_{1:k})$ is the active set after k iterations.

The lemma is proved using standard information-theoretic identities coupled with the properties of our noise model. Refer Append §A.2.1 for a detailed proof. This result is in line with intuition since the noisy component of every test (N_T) is independent of Y and hence uninformative for prediction. Thus, the selection of the most informative next test is governed solely by how well it's true outcome approximately bisects the current active set $\mathcal{A}(t_{1:k})$.

A natural question to ask next is, *If a given set of tests $\mathcal{T}$ is (δ, γ)-unpredictable then what can we conclude about the probability, $\mathbb{P}(D_{T_{k+1}} = 1 \mid \mathcal{A}(t_{1:k}))$, of the the chosen test*? The following lemma answers this.

Lemma 5.2. *Under the BSC model with noise parameter $\alpha \in [0, \frac{1}{2}]$, if $\mathcal{T}$ is (δ, γ)-unpredictable according to definition 3.1, then in any iteration $k+1$, the greedy strategy will either choose a test $T_{k+1} \in \mathcal{T}$ such that*

$$\left| \mathbb{P}\left(D_{T_{k+1}} = 1 \mid \mathcal{A}(t_{1:k})\right) - \frac{1}{2} \right| \leq \frac{\delta}{1 - 2\alpha}, \tag{8}$$

or terminate according to γ stopping criterion. Moreover, given α and Y, it is not possible to have a (δ, γ)-unpredictable $\mathcal{T}$ for $\delta > \frac{1}{2} - \alpha$ with $\gamma > \max_{y \in \mathcal{Y}} \mathbb{P}(Y = y)$, which is the probability of the mode of the prior distribution over Y.

Refer to Appendix §A.3 for a proof. The above result has two consequences.

1. It shows that for a fixed δ, as the noise level α increases from 0 to $\frac{1}{2}$ (it's maximum possible value) the ability of the true outcome $D_T = D_T(Y)$ for any given test $T \in \mathcal{T}$ to approximately bisect the current active set deteriorates by a factor of $\frac{1}{1-2\alpha}$ compared to the observed test outcome $T = t$. Based on this, one can conjecture that as the noise level increases, more and more tests would be needed to identify Y, because the ability of the true outcomes to approximately bisect the current active set degrades.

2. It shows that the maximum possible value of δ is bounded by the noise level α. In particular, by inverting the result in equation 8 (see Appendix §A.3) we see that if $\left| \mathbb{P}\left(D_{T_{k+1}} = 1 \mid \mathcal{A}(t_{1:k})\right) - \frac{1}{2} \right| \leq \delta'$ for some constant $\delta' \in [0, \frac{1}{2}]$, then this implies $\delta = \delta'(1-2\alpha) \in [0, \frac{1}{2} - \alpha]$. Thus, unlike the oracle case, it is not possible to have a set of noisy tests which is (δ, γ)-unpredictable for $\delta > \frac{1}{2} - \alpha$ with γ greater than the probability of the mode of the prior distribution over Y. The condition on γ is needed to ensure that InfoMax would require at least one test to gather information about Y, otherwise the prior distribution would suffice to make a prediction. In hindsight, this result makes sense since according to our noise model, every test outcome is corrupted independently of all other tests and hence, there will always be some uncertainty in a certain test's outcome regardless of how many tests have been carried out so far.

Having stated all the ingredients we will now present our main result for the greedy strategy under the BSC noise model.

Theorem 5.3. *Fix noise level $\alpha \in [0, \frac{1}{2}]$ for the BSC model. Fix $\delta \in [0, \frac{1}{2} - \alpha]$. Given a (δ, γ)-unpredictable $\mathcal{T}$, the average number of tests needed by the InfoMax algorithm to predict Y with confidence at least γ under the BSC model is at most*

$$B_{Noisy} := \frac{H(Y)}{h(\frac{1}{2} + \delta) - h(\alpha)}. \tag{9}$$

A complete proof can be found in Appendix §A.4. The main idea behind the proof of this theorem is similar to the proof sketch for Theorem 4.1 with the additional nuance that carrying out tests after the posterior has peaked will still incur a non-zero conditional entropy due to the presence of noise. This nuance results in the additional $h(\alpha)$ term in the denominator of the bound (compared to equation 5). Notice that our bound does not explicitly depend on γ, we defer a discussion on this for later.

To the best of our knowledge, this is a first such result for the InfoMax algorithm for noisy tests. Observe that in the absence of noise, that is, when $\alpha = 0$, we recover our bound for the oracle case (Theorem 4.1). In the extreme case, when $\alpha = \frac{1}{2}$, none of the tests are informative about Y. As a result, InfoMax would require an infinite amount of tests to make progress since the distribution over Y will remain

unchanged no matter how many tests are carried out (in other words, the posterior over Y would stay same as the prior). This is reflected in our bound where the denominator converges to zero as $\alpha \to \frac{1}{2}$ (recall $\delta \leq \frac{1}{2} - \alpha$ from Lemma 5.2). Refer Figure 5 in the Appendix for a plot of the ratio $\frac{\mathrm{B}_{\text{Noisy}}}{\mathrm{B}_{\text{Ours}}}$ for different (δ, α) values.

Information-theoretic interpretation of our bound. Recall that since we assume $\mathcal{T}$ is (δ, γ)-unpredictable, in each iteration k (until termination), InfoMax will select a test T_k whose entropy $H(T_k \mid \mathcal{A}(t_{1:k-1})) \geq h(\frac{1}{2} + \delta)$. Thus, the denominator in equation 9 can be upper bounded as follows,

$$\begin{aligned} h(\frac{1}{2} + \delta) - h(\alpha) &\leq H(T_k \mid \mathcal{A}(t_{1:k-1})) - H(T_k \mid Y) \\ &= I(T_k; Y \mid \mathcal{A}(t_{1:k-1}). \end{aligned} \tag{10}$$

In the first inequality we substituted $h(\alpha) = H(T_k \mid Y)$ (which comes directly from the definition of our BSC model). Thus, in each iteration, InfoMax will pick a test that contains at least $h(\frac{1}{2} + \delta) - h(\alpha)$ units of conditional mutual information about Y. Since, the total information contained in Y is characterized by its entropy, $H(Y)$, it makes intuitive sense to expect that $\frac{H(Y)}{h(\frac{1}{2}+\delta)-h(\alpha)}$ number of tests would be needed at most by InfoMax to obtain all the information about Y that is present in $\mathcal{T}$. This gives an information-theoretic justification of our bound in Theorem 5.3 that seems to indicate that our bound for the performance of InfoMax under BSC noise is tight.

Relation between γ, δ and α. Although $\mathrm{B}_{\text{Noisy}}$ does not explicitly depend on the confidence level (for prediction) γ, there is an implicit dependence since the noise level α and the unpredictability level δ put a constrain on the maximum achievable value of γ using InfoMax. Thus, $\mathrm{B}_{\text{Noisy}}$ should be interpreted as an upper bound on Infomax's performance with the maximum achievable γ for a given (δ, α) pair. This is because, if a given $\mathcal{T}$ is (δ, γ_0)-unpredictable, then it is also (δ, γ)-unpredictable for any $\gamma \leq \gamma_0$.

We now present our result on the maximum possible γ, given a (δ, α) pair.

Lemma 5.4. *Fix a noise level $\alpha \in [0, \frac{1}{2}]$ for the BSC model. For a given $\delta \in [0, \frac{1}{2} - \alpha]$, there does not exist a (δ, γ)-unpredictable $\mathcal{T}$ for any* $\gamma > 1 - \frac{\left(\frac{1}{2} - \frac{\delta}{1-2\alpha}\right)\alpha}{\left(\frac{1}{2} + \frac{\delta}{1-2\alpha}\right)(1-\alpha) + \left(\frac{1}{2} - \frac{\delta}{1-2\alpha}\right)\alpha}$.

Figure 6 in the Appendix shows the maximum achievable γ for different values of (δ, α). Notice, when $\alpha = 0$, there is no noise in the test outcomes and hence it is possible to identify Y with a suitably chosen $\mathcal{T}$. However, as α increases, the confidence (γ) with which we can make a prediction decreases due to uncertainty induced by the noise.

We conclude by a brief discussion on the limitations of this work in the next section.

Limitations

The following are the limitations of this work.

- Our analysis assumes tests are δ-unpredictable for some fixed value of δ, however *a priori* it is not known how to find δ such that the given set of tests would be δ-unpredictable.
- Moreover, the BSC noise model assumes i.i.d. noise; however, in practice noise is often dependent on Y, and test outcomes are often not conditionally independent of each other.
- Finally, our bound $\mathrm{B}_{\text{Noisy}}$ does not explicitly depend on γ but there is an implicit dependence since the noise level α and the unpredictability level δ put a constraint on the maximum achievable value for γ using InfoMax (see Lemma 5.4).

We hope future work addresses these limitations by studying more complex noise models, designing testable conditions to verify if a given $\mathcal{T}$ is δ-unpredictable for a given value of δ or not, and incorporating a more explicit dependence on γ in the performance bounds for InfoMax.

Acknowledgements

This research was supported by the Army Research Office under the Multidisciplinary University Research Initiative contract W911NF-17-1-0304, and the NSF–Simons Research Collaboration on the Mathematical and Scientific Foundations of Deep Learning (NSF grant 2031985, Simons grant 814201). Moreover, the authors acknowledge Hamed Hassani for their insightful feedback, which helped improve the presentation of this work, and for providing intuition during our initial discussions about proving the tightness of our derived bounds in Section 4.3. Finally, we would like to thank our anonymous reviewer for suggesting the bound in Theorem 4.1 as an improvement over our bound in an earlier version of this paper.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Branson, S., Van Horn, G., Wah, C., Perona, P., and Belongie, S. The ignorant led by the blind: A hybrid human–machine vision system for fine-grained categorization. *International Journal of Computer Vision*, 108:3–29, 2014.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. Cart. *Classification and regression trees*, 1984.

Chattopadhyay, A., Slocum, S., Haeffele, B. D., Vidal, R., and Geman, D. Interpretable by design: Learning predictors by composing interpretable queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Chattopadhyay, A., Chan, K. H. R., Haeffele, B. D., Geman, D., and Vidal, R. Variational information pursuit for interpretable predictions. In *The Eleventh International Conference on Learning Representations*, 2023. doi: 10.48550/arXiv.2302.02876.

Chen, Y., Hassani, S. H., Karbasi, A., and Krause, A. Sequential information maximization: When is greedy near-optimal? In *Conference on Learning Theory*, pp. 338–363. PMLR, 2015.

Chung, H. W., Sadler, B. M., Zheng, L., and Hero, A. O. Unequal error protection querying policies for the noisy 20 questions problem. *IEEE Transactions on Information Theory*, 64(2):1105–1131, 2017.

Cover, T. M. *Elements of information theory*. John Wiley & Sons, 1999.

Covert, I. C., Qiu, W., Lu, M., Kim, N. Y., White, N. J., and Lee, S.-I. Learning to maximize mutual information for dynamic feature selection. In *International Conference on Machine Learning*, pp. 6424–6447. PMLR, 2023.

Cuturi, M., Teboul, O., Berthet, Q., Doucet, A., and Vert, J.-P. Noisy adaptive group testing using bayesian sequential experimental design. *arXiv preprint arXiv:2004.12508*, 2020.

Dasgupta, S. Analysis of a greedy active learning strategy. *Advances in neural information processing systems*, 17, 2004.

Flores, I. and Madpis, G. Average binary search length for dense ordered lists. *Communications of the ACM*, 14(9): 602–603, 1971.

Foster, A., Jankowiak, M., Bingham, E., Horsfall, P., Teh, Y. W., Rainforth, T., and Goodman, N. Variational bayesian optimal experimental design. *Advances in Neural Information Processing Systems*, 32, 2019.

Garey, M. R. and Graham, R. L. Performance bounds on the splitting algorithm for binary testing. *Acta Informatica*, 3 (4):347–355, 1974.

Geman, D. and Jedynak, B. An active testing model for tracking roads in satellite images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):1–14, 1996.

Geman, D., Geman, S., Hallonquist, N., and Younes, L. Visual turing test for computer vision systems. *Proceedings of the National Academy of Sciences*, 112(12):3618–3623, 2015.

Golovin, D., Krause, A., and Ray, D. Near-optimal bayesian active learning with noisy observations. *Advances in Neural Information Processing Systems*, 23, 2010.

He, W., Mao, X., Ma, C., Huang, Y., Hernàndez-Lobato, J. M., and Chen, T. Bsoda: a bipartite scalable framework for online disease diagnosis. In *Proceedings of the ACM Web Conference 2022*, pp. 2511–2521, 2022.

Horstein, M. Sequential transmission using noiseless feedback. *IEEE Transactions on Information Theory*, 9(3): 136–143, 1963.

Hyafil, L. and Rivest, R. L. Constructing optimal binary decision trees is np-complete. *Information processing letters*, 5(1):15–17, 1976.

Jahangiri, E., Yoruk, E., Vidal, R., Younes, L., and Geman, D. Information pursuit: A bayesian framework for sequential scene parsing. *arXiv preprint arXiv:1701.02343*, 2017.

Jedynak, B., Frazier, P. I., and Sznitman, R. Twenty questions with noise: Bayes optimal policies for entropy loss. *Journal of Applied Probability*, 49(1):114–136, 2012.

Koh, P. W., Nguyen, T., Tang, Y. S., Mussmann, S., Pierson, E., Kim, B., and Liang, P. Concept bottleneck models. In *International Conference on Machine Learning*, pp. 5338–5348. PMLR, 2020.

Kosaraju, S. R., Przytycka, T. M., and Borgstrom, R. On an optimal split tree problem. In *Algorithms and Data Structures: 6th International Workshop, WADS'99 Vancouver, Canada, August 11–14, 1999 Proceedings*, pp. 157–168. Springer, 1999.

Lindley, D. V. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27 (4):986–1005, 1956.

Loveland, D. W. Performance bounds for binary testing with arbitrary weights. *Acta Informatica*, 22(1):101–114, 1985.

Ma, C., Tschiatschek, S., Palla, K., Hernández-Lobato, J. M., Nowozin, S., and Zhang, C. Eddi: Efficient dynamic discovery of high-value information with partial vae. *arXiv preprint arXiv:1809.11142*, 2018.

Nowak, R. Generalized binary search. In *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, pp. 568–574. IEEE, 2008.

Peng, Y.-S., Tang, K.-F., Lin, H.-T., and Chang, E. Refuel: Exploring sparse features in deep reinforcement learning for fast disease diagnosis. *Advances in neural information processing systems*, 31, 2018.

Sandelius, M. On an optimal search procedure. *The American Mathematical Monthly*, 68(2):133–134, 1961.

Shannon, C. E. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Sznitman, R. and Jedynak, B. Active testing for face detection and localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(10):1914–1920, 2010.

Tsiligkaridis, T., Sadler, B. M., and Hero, A. O. Collaborative 20 questions for target localization. *IEEE Transactions on Information Theory*, 60(4):2233–2252, 2014.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. 2011.

Xian, Y., Lampert, C. H., Schiele, B., and Akata, Z. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2251–2265, 2018.

Zheng, A. X., Rish, I., and Beygelzimer, A. Efficient test selection in active diagnosis via entropy approximation. *arXiv preprint arXiv:1207.1418*, 2012.

A. Appendix

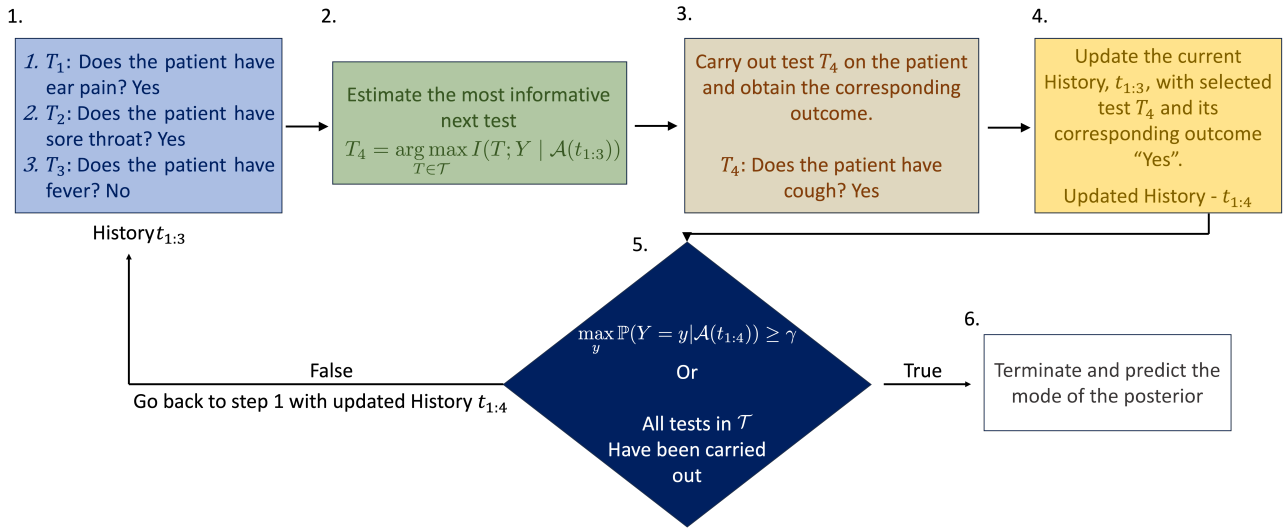

Figure 4. **Illustration of the greedy Information Maximization algorithm.** A flow chart depicting one iteration (k = 4) of the algorithm. As an example we consider the task of disease diagnosis where Y denotes the disease a patient is suffering from (for example, tuberculosis, common cold etc.) and the set of tests $\mathcal{T}$ here corresponds to questions about different symptoms the patient may be experiencing. The corresponding binary answer (Yes/No) to every test indicates the outcome of that test.

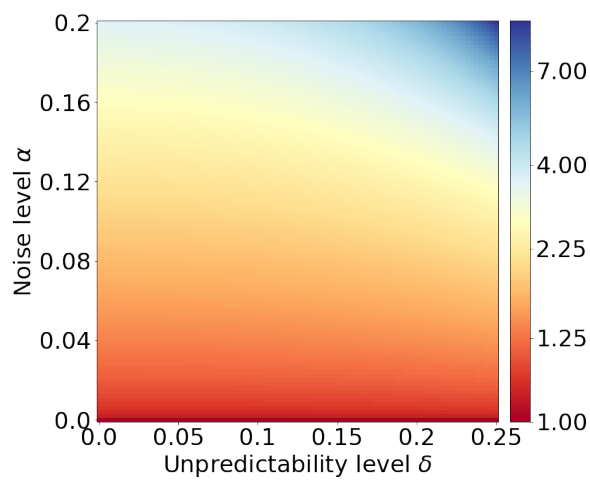

Figure 5. **Comparing our bound for the performance of InfoMax under noisy tests vs. oracle tests.** We plot the ratio $\rho := \frac{\mathrm{B_{Noisy}}}{\mathrm{B_{Oracle}}}$ for different values of δ and α. Best viewed in colour.

A.1. Proofs

For ease of reading, we rewrite all the theorems, lemmas and corollaries from the main paper in this section (unnnumbered) before presenting their respective proofs.

A.1.1. Proof of Theorem 4.1

Theorem. Fix any $\delta \in [0, \frac{1}{2}]$. Let h be the binary entropy function. Given a δ-unpredictable $\mathcal{T}$ that collectively determine Y, the average number of tests needed by the information maximization algorithm to identify Y is at most $\frac{H(Y)}{h(\frac{1}{2}+\delta)}$.

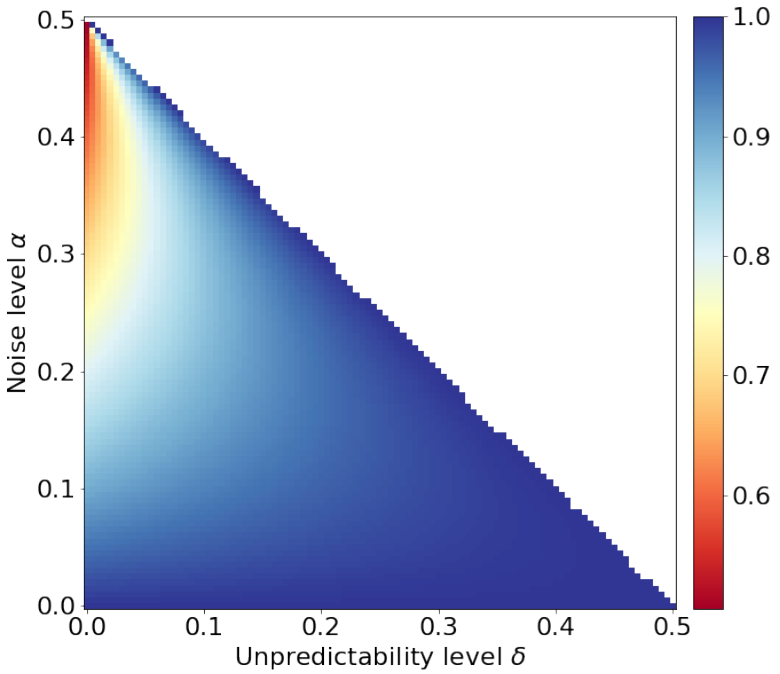

Figure 6. **Maximum achievable γ for a given (δ, α) pair.** We plot the bound in Lemma 5.4 for different values of δ and α. Best viewed in colour. The white area represents the region where $\delta > \frac{1}{2} - \alpha$ which is not possible according to Lemma 5.2.

Proof. Assume the InfoMax algorithm has run for k iterations. There are two cases,

Case 1. Y has not been identified. Then, since $\mathcal{T}$ is δ-unpredictable we know for the test T_{k+1} selected in iteration $k+1$

$$H(T_{k+1} \mid T_1 = t_1, \ldots, T_k = t_k) \geq h(\frac{1}{2} + \delta) \tag{11}$$

The inequality is obtained using the concavity of the binary entropy function since the chosen test satisfies $|\mathbb{P}(T_{k+1} = 1 \mid T_1 = t_1, \ldots, T_k = t_k) - \frac{1}{2}| \leq \delta$ (from definition 3.1).

Case 2. Y has been identified, at which point the algorithm would terminate and all further tests would have 0 conditional entropy since their outcome will be determined by the value of Y that has been identified, that is,

$$H(T_{k+1} \mid T_1 = t_1, \ldots, T_k = t_k) = 0 \tag{12}$$

Define τ to be the random variable indicating the stopping time for a single run of the InfoMax algorithm. Equation 11 and equation 12 can be combined into

$$H(T_{k+1} \mid T_1 = t_1, \ldots, T_k = t_k) \geq h(\frac{1}{2} + \delta)\mathbb{1}(\tau > k), \tag{13}$$

where $\mathbb{1}$ is the indicator random variable.

Taking expectation on both sides,

$$H(T_{k+1} \mid T_1, T_2, ..., T_k) \geq h(\frac{1}{2} + \delta)\mathbb{P}(\tau > k), \tag{14}$$

Summing k from 0 to $|\mathcal{T}|-1$ we obtain

$$\begin{aligned} h(\frac{1}{2}+\delta)\sum_{k=0}^{|\mathcal{T}|-1}\mathbb{P}(\tau>k) &\leq H(T_1,T_2,...,T_{|\mathcal{T}|}) \\ &\leq H(Y) \\ \implies \mathbb{E}[\tau] &\leq \frac{H(Y)}{h(\frac{1}{2}+\delta)}, \end{aligned} \tag{15}$$

which is the desired bound. The second inequality is obtained since all the test outcomes are functions of Y. □

A.2. More details about the example showing tightness of our bound in Theorem 4.1

For ease of readers, we rewrite the example before proceeding to the lemmas.

Given some $\delta \in [0, \frac{1}{2}]$. Take any integer $n \in \mathbb{N}$ and let Y be a random variable taking n different values with the following probability mass function for any $k \in \{1, 2, ..., n-1\}$

$$\mathbb{P}(Y=k) = (\frac{1}{2}-\delta)(\frac{1}{2}+\delta)^{k-1}. \tag{16}$$

The remaining mass of $(\frac{1}{2}+\delta)^{n-1}$ is placed on the last element n, that is, $\mathbb{P}(Y=n) = (\frac{1}{2}+\delta)^{n-1}$.

Lemma A.1. *The entropy of the above described distribution is*

$$H(Y) = h(\frac{1}{2}+\delta)\sum_{i=0}^{n-2}(\frac{1}{2}+\delta)^i,$$

where h is the binary entropy function.

Proof. We will proceed by induction.
Base case: n = 2. In this case the distribution over Y is as follows,

$$\mathbb{P}(Y=1) = \frac{1}{2}-\delta; \mathbb{P}(Y=1) = \frac{1}{2}+\delta.$$

The entropy for this distribution is $h(\frac{1}{2}+\delta)$.

Induction step: n = j. Assume if $\hat{Y}$ can take j values, with the distribution as specified above in our example (equation 16), then

$$H(\hat{Y}) = h(\frac{1}{2}+\delta)\sum_{i=0}^{j-2}(\frac{1}{2}+\delta)^i.$$

Now, let Y be a random variable that takes $j+1$ values. Y can be constructed from $\hat{Y}$ as follows,

$$\mathbb{P}(Y=k+1) = \mathbb{P}(\hat{Y}=k)(\frac{1}{2}+\delta) \;\; \forall k \in \{1, 2, ..., j\}. \tag{17}$$

Set $\mathbb{P}(Y=1) = \frac{1}{2}-\delta$.

Now, divide the values Y can take into two groups $g_1 = 1$ and $g_2 = \{2, 3, ..., j+1\}$. Let G be a random variable taking values g_1 and g_2. Then

$$\mathbb{P}(G=g_1) = \mathbb{P}(Y=1) = \frac{1}{2}-\delta \text{ and } \mathbb{P}(G=g_2) = 1-\mathbb{P}(Y=1) = \frac{1}{2}+\delta.$$

The entropy of Y can then be computed as follows.

$$\begin{aligned} H(Y) &= H(G) + H(Y \mid G) \\ &= h(\frac{1}{2}+\delta) + \mathbb{P}(G=g_1)H(Y \mid G=g_1) + \mathbb{P}(G=g_2)H(Y \mid G=g_2) \\ &= h(\frac{1}{2}+\delta) + (\frac{1}{2}+\delta)H(\hat{Y}) \\ &= h(\frac{1}{2}+\delta)\sum_{i=0}^{j-1}(\frac{1}{2}+\delta)^i. \end{aligned} \tag{18}$$

The first equality is a property of entropy under grouping of elements a random variable (Cover, 1999). In the second equality, we used the fact that $H(Y \mid G=g_1)=0$ since g_1 is a singleton set. Moreover, $H(Y \mid G=g_2)=H(\hat{Y})$, since for all $k \in \{2,3,...,j+1\}$ $P(Y=k \mid G=g_2) = \frac{P(Y=k)}{P(G=g_2)} = \frac{P(Y=k)}{\frac{1}{2}+\delta} = P(\hat{Y}=k-1)$ (equation 17). Thus, proving the desired lemma. □

Lemma A.2. *Let Y take n values with distribution as described in equation 16. If $\mathcal{T}$ is the set of all singleton tests of the form "Is Y = y?" for all $y \in \{1,2,...n\}$, then the InfoMax algorithm will take exactly $\frac{H(Y)}{h(\frac{1}{2}+\delta)}$ tests on average to identify Y.*

Proof. We will first show by induction that at step k, if the algorithm has not identified Y and terminated already, then InfoMax will pick the test "Is $Y=k$?". Thus, InfoMax will sequentially eliminate the possible values Y can take starting from 1 in iteration 1, 2 in iteration 2 and so on.

Base case: step = 1. The test closest to one-half is T_1 ="Is $Y=1$?" with $P(T_1=1)=\frac{1}{2}-\delta$. All the remaining singleton tests in $\mathcal{T}$ will have probability of being true less than $\frac{1}{2}-\delta$. Thus, the hypothesis $Y=1$ will be eliminated from the active set in the first iteration.

Induction step: step = *k*. Assume the induction hypothesis holds that the first $k-1$ values of Y have been eliminated so far, of the remaining values,

$$\mathbb{P}\Big(Y=k \mid \mathcal{A}(t_{1:k-1})\Big) = \frac{\mathbb{P}(Y=k)}{\mathbb{P}\Big(\mathcal{A}(t_{1:k-1})\Big)} = \frac{(\frac{1}{2}-\delta)(\frac{1}{2}+\delta)^{k-1}}{(\frac{1}{2}+\delta)^{k-1}} = \frac{1}{2}-\delta.$$

Now, if $1<k<n-1$, then it can be easily checked that all the remaining hypothesis in the active set $\mathcal{A}(t_{1:k-1})$ will have their probability mass strictly less than $\frac{1}{2}-\delta$. Thus, at iteration k, hypothesis $Y=k$ will be eliminated from the active set.

Alternatively, if $k=n-1$, then the only two remaining values of Y in the active set are $n-1$ and n. Observe that both tests, "Is $Y=n-1$?" or "Is $Y=n$?", will lead to termination as Y would have been identified. Moreover, both tests are complements of each other (around one-half) and have exactly the same conditional entropy (given the first $n-2$ observed outcomes). Thus, we can assume without loss of generality that at step $k=n-1$, InfoMax will pick the test "Is $Y=n-1$?".

Based on this, we conclude that the average number of tests needed by InfoMax to identify Y is,

$$\text{Avg \# tests} = \sum_{i=1}^{n-1}\mathbb{P}(Y=i)i + \mathbb{P}(Y=n)(n-1) = \sum_{i=1}^{n-1}(\frac{1}{2}-\delta)(\frac{1}{2}+\delta)^{i-1}i + (\frac{1}{2}+\delta)^{n-1}(n-1). \tag{19}$$

Let $S := \sum_{i=1}^{n-1}(\frac{1}{2}+\delta)^{i-1}i$. Rewriting Avg # tests in terms of S we get,

$$\begin{aligned} \text{Avg \# tests} &= S(\frac{1}{2}-\delta) + (\frac{1}{2}+\delta)^{n-1}(n-1) \\ &= S - (\frac{1}{2}+\delta)S + (\frac{1}{2}+\delta)^{n-1}(n-1) \\ &= \sum_{i=0}^{n-2}(\frac{1}{2}+\delta)^i \end{aligned} \tag{20}$$

Thus, proving the desired lemma since from Lemma A.1 we know that $\sum_{i=0}^{n-2}(\frac{1}{2}+\delta)^i = \frac{H(Y)}{h(\frac{1}{2}+\delta)}$. □

A.2.1. Proof of Lemma 5.1

Lemma. Under the BSC noise model, at any iteration $k+1$, the information maximization algorithm will pick the test T_{k+1}, such that

$$T_{k+1} = \arg\min_{T \in \mathcal{T}} \left| \mathbb{P}(D_T = 1 \mid \mathcal{A}(t_{1:k})) - \frac{1}{2} \right|,$$

where $\mathcal{A}(t_{1:k})$ is the active set after k iterations.

Proof. Let h be the binary entropy function. At any iteration $k+1$, the mutual information for any $T \in \mathcal{T}$ can be written as follows,

$$\begin{aligned} &I(T, Y \mid \mathcal{A}(t_{1:k}))) \\ &= H(T \mid \mathcal{A}(t_{1:k}))) - H(T \mid Y, \mathcal{A}(t_{1:k}))) \\ &= h\left(\sum_{y \in \mathcal{Y}} \mathbb{P}\big(y \mid \mathcal{A}(t_{1:k})\big) \mathbb{P}(T = 1 \mid y) \right) - \sum_{y \in \mathcal{Y}} \mathbb{P}(y \mid \mathcal{A}(t_{1:k})) h\left(\mathbb{P}(T = 1 \mid y) \right) \end{aligned} \tag{21}$$

Define $\delta_T := \sum_{\{y \in \mathcal{Y} : D_T(y) = 1\}} \mathbb{P}(y \mid \mathcal{A}(t_{1:k}))$, that is, δ_T is the total posterior mass on Y (given the current active set) subject to the constraint that $D_T(Y) = 1$. We can rewrite equation 21 as,

$$\begin{aligned} I(T, Y \mid \mathcal{A}(t_{1:k}))) &= h\left(\delta_T(1-\alpha) + (1-\delta_T)\alpha \right) - \delta_T h(1-\alpha) - (1-\delta_T) h(\alpha) \\ &= h\left(\delta_T(1-\alpha) + (1-\delta_T)\alpha \right) - h(\alpha). \end{aligned} \tag{22}$$

Equation 22 shows that the mutual information between T and Y given the current active set is just the mutual information of a noisy binary symmetric channel, for which we know the maximal value is attained at $\delta_T = \frac{1}{2}$. Moreover, the binary entropy function is concave thus proving the desired lemma.

□

A.3. Proof of Lemma 5.2

Lemma. Under the BSC model with noise parameter $\alpha \in [0, \frac{1}{2}]$, if $\mathcal{T}$ is (δ, γ)-unpredictable according to definition 3.1, then in any iteration $k+1$, the greedy strategy will either choose a test $T_{k+1} \in \mathcal{T}$ such that

$$\left| \mathbb{P}\left(D_{T_{k+1}} = 1 \mid \mathcal{A}(t_{1:k}) \right) - \frac{1}{2} \right| \leq \frac{\delta}{1 - 2\alpha}, \tag{23}$$

or terminate according to γ stopping criterion. Moreover, given α, it is not possible to have a (δ, γ)-unpredictable $\mathcal{T}$ for $\delta > \frac{1}{2} - \alpha$.

Proof. We know from the BSC noise model that

$$\begin{aligned} \mathbb{P}(T_{k+1} = 1 \mid \mathcal{A}(t_{1:k})) = \mathbb{P}\big(T_{k+1} = 1 \mid D_{T_{k+1}} = 1\big) \mathbb{P}\big(D_{T_{k+1}} = 1 \mid \mathcal{A}(t_{1:k})\big) \\ + \mathbb{P}\big(T_{k+1} = 1 \mid D_{T_{k+1}} = 0\big) \mathbb{P}\big(D_{T_{k+1}} = 0 \mid \mathcal{A}(t_{1:k})\big). \end{aligned} \tag{24}$$

Let $x := \mathbb{P}\big(D_{T_{k+1}} = 1 \mid \mathcal{A}(t_{1:k})\big)$. The above equation can be rewritten as

$$\mathbb{P}\big(T_{k+1} = 1 \mid \mathcal{A}(t_{1:k})\big) = \mathbb{P}\big(T_{k+1} = 1 \mid D_{T_{k+1}} = 1\big) x + \mathbb{P}\big(T_{k+1} = 1 \mid D_{T_{k+1}} = 0\big)(1 - x). \tag{25}$$

From our noise model we know $\mathbb{P}\big(T_{k+1} = 1 \mid D_{T_{k+1}} = 1\big) = 1 - \alpha$ and $\mathbb{P}\big(T_{k+1} = 1 \mid D_{T_{k+1}} = 0\big) = \alpha$. Substituting this in equation 25 we get,

$$\begin{aligned} \mathbb{P}\big(T_{k+1} = 1 \mid \mathcal{A}(t_{1:k})\big) &= (1-\alpha)x + (1-x)\alpha \\ \implies x &= \frac{\mathbb{P}\big(T_{k+1} = 1 \mid \mathcal{A}(t_{1:k})\big) - \alpha}{1 - 2\alpha} \end{aligned} \tag{26}$$

Since $\mathcal{T}$ is (δ, γ)-unpredictable, the chosen test T_{k+1} at iteration $k+1$ satisfies

$$\frac{1}{2} - \delta \leq \mathbb{P}\left(T_{k+1} = 1 \mid \mathcal{A}(t_{1:k})\right) \leq \frac{1}{2} + \delta \tag{27}$$

Combining equation 26 and equation 27 we obtain,

$$\begin{aligned} \frac{\frac{1}{2}-\alpha}{1-2\alpha} - \frac{\delta}{1-2\alpha} \leq \quad x \quad &\leq \frac{\frac{1}{2}-\alpha}{1-2\alpha} + \frac{\delta}{1-2\alpha} \\ \implies -\frac{\delta}{1-2\alpha} \leq x - \frac{1}{2} &\leq \frac{\delta}{1-2\alpha}, \end{aligned} \tag{28}$$

proving the lemma.

Moreover, notice that equation 28 can be inverted. In particular if we assume $x := \mathbb{P}\big(D_{T_{k+1}} = 1 \mid \mathcal{A}(t_{1:k})\big)$ is between $\frac{1}{2} \pm \delta'$ for some $\delta' \in [0, \frac{1}{2}]$, we can conclude from equation 26 that

$$\left|\mathbb{P}\big(T = 1 \mid \mathcal{A}(t_{1:k})\big) - \frac{1}{2}\right| \leq \delta'(1-2\alpha). \tag{29}$$

It is immediate from equation 29 that $\mathbb{P}\big(T = 1 \mid \mathcal{A}(t_{1:k})\big)$ cannot be more than $\frac{1}{2} - \alpha$ units away from one-half (obtained by setting $\delta' = \frac{1}{2}$ in the above inequality).

The result in equation 29 will be used to re-parameterize δ for a given (δ, γ)-unpredictable set of tests in terms of δ' to get rid of the constraint on δ in terms of the noise level α. □

A.4. Proof of Theorem 5.3

Theorem. Fix noise level $\alpha \in [0, \frac{1}{2}]$ for the BSC model. Fix $\delta \in [0, \frac{1}{2} - \alpha]$. Let h be the binary entropy function. Given a (δ, γ)-unpredictable $\mathcal{T}$, the average number of tests needed by the InfoMax algorithm to predict Y with confidence at least γ under the BSC model is at most

$$\mathrm{B}_{\text{Noisy}} := \frac{H(Y)}{h(\frac{1}{2} + \delta) - h(\alpha)}. \tag{30}$$

Proof. Assume the InfoMax algorithm has run for k iterations. There are two cases,

Case 1. Y has not been predicted with confidence γ, that is, $\max_{y \in \mathcal{Y}} |\mathbb{P}(Y = y \mid T_1 = t_1, \ldots, T_k = t_l)| < \gamma \; \forall l \leq k$. Then, since $\mathcal{T}$ is (δ, γ)-unpredictable we know for the test T_{k+1} selected in iteration $k+1$

$$H(T_{k+1} \mid T_1 = t_1, \ldots, T_k = t_k) \geq h(\frac{1}{2} + \delta) \tag{31}$$

The inequality is obtained using the concavity of the binary entropy function since the chosen test satisfies $|\mathbb{P}(T_{k+1} = 1 \mid T_1 = t_1, \ldots, T_k = t_k) - \frac{1}{2}| \leq \delta$ (from definition 3.1, lemma 5.1 and the fact that $|\mathbb{P}(D_{T_{k+1}} = 1 \mid T_1 = t_1, \ldots, T_k = t_k) - \frac{1}{2}|$ is directly proportional to $|\mathbb{P}(T_{k+1} = 1 \mid T_1 = t_1, \ldots, T_k = t_k) - \frac{1}{2}|$ [equation 29]).

Case 2. The algorithm already encountered a posterior mass of $\max_{y \in \mathcal{Y}} \mathbb{P}\big(Y = y \mid T_1 = t_1, \ldots, T_k = t_l\big) \geq \gamma$ for some $l < k$ number of iterations. At this point, the algorithm should have terminated according to definition 3.1. If further tests are carried out, we are not guaranteed (from the assumption of (δ, γ)-unpredictability of $\mathcal{T}$) that there $\exists T_{k+1} \in \mathcal{T}$ such that $|\mathbb{P}(T_{k+1} = 1 \mid T_1 = t_1, \ldots, T_k = t_k) - \frac{1}{2}| \leq \delta$. However, under the BSC model we are guaranteed that every test outcome is flipped with α probability independent of the history. Using this insight, we obtain

$$\begin{aligned} H(T_{k+1} \mid T_1 = t_1, \ldots, T_k = t_k) &\geq H(T_{k+1} \mid D_{T_{k+1}}, T_1 = t_1, \ldots, T_k = t_k) \\ &= H(T_{k+1} \mid D_{T_{k+1}}) \\ &= h(\alpha) \end{aligned} \tag{32}$$

The first inequality is obtained using the fact that conditioning on random variables can only reduce the entropy. The second equality follows from the fact that conditioned on the true test $D_{T_{k+1}}$, T_{k+1} is independent of the history of

outcomes observed so far. The last equality is obtained by observing that $\mathbb{P}(T_{k+1} = 1 \mid D_{T_{k+1}} = 1) = 1 - \alpha$ and $\mathbb{P}(T_{k+1} = 1 \mid D_{T_{k+1}} = 0) = \alpha$ (from definition of BSC model) and the fact that the binary entropy function is symmetric around one-half (that is, $h(\alpha) = h(1-\alpha)$).

Define τ to be the random variable indicating the stopping time for a single run of the InfoMax algorithm. equation 31 and equation 32 can be combined into

$$H(T_{k+1} \mid T_1 = t_1, \ldots, T_k = t_k) \geq h(\frac{1}{2} + \delta)\mathbb{1}(\tau > k) + h(\alpha)\mathbb{1}(\tau \leq k), \tag{33}$$

where $\mathbb{1}$ is the indicator random variable.

Taking expectation on both sides,

$$H(T_{k+1} \mid T_1, T_2, ..., T_k) \geq h(\frac{1}{2} + \delta)\mathbb{P}(\tau > k) + h(\alpha)\mathbb{P}(\tau \leq k), \tag{34}$$

Summing k from 0 to $|\mathcal{T}| - 1$ we obtain

$$\begin{aligned} h(\frac{1}{2} + \delta) \sum_{k=0}^{|\mathcal{T}|-1} \mathbb{P}(\tau > k) + h(\alpha) \sum_{k=0}^{|\mathcal{T}|-1} \Big(1 - \mathbb{P}(\tau > k)\Big) &\leq H(T_1, T_2, ..., T_{|\mathcal{T}|}) \\ &\leq H(Y, T_1, T_2, ..., T_{|\mathcal{T}|}) \\ &= H(Y) + |\mathcal{T}|h(\alpha) \\ \implies h(\frac{1}{2} + \delta)\mathbb{E}[\tau] + h(\alpha)(|\mathcal{T}| - \mathbb{E}[\tau]) &\leq H(Y) + |\mathcal{T}|h(\alpha) \\ \implies \mathbb{E}[\tau] &\leq \frac{H(Y)}{h(\frac{1}{2} + \delta) - h(\alpha)}, \end{aligned} \tag{35}$$

which is the desired bound. The second inequality is obtained by using the fact that the joint entropy is always more than or equal to the marginal entropy of just the tests. The first equality is obtained by our noise model since given Y, the test outcomes are independent of each other and the only remaining uncertainty is the noise. □

A.5. Proof of Lemma 5.4

Lemma. Fix a noise level $\alpha \in [0, \frac{1}{2}]$ for the BSC model. For a given $\delta \in [0, \frac{1}{2} - \alpha]$, there does not exist a (δ, γ)-unpredictable $\mathcal{T}$ for any $\gamma > 1 - \frac{\left(\frac{1}{2} - \frac{\delta}{1-2\alpha}\right)\alpha}{\left(\frac{1}{2} + \frac{\delta}{1-2\alpha}\right)(1-\alpha) + \left(\frac{1}{2} - \frac{\delta}{1-2\alpha}\right)\alpha}$.

Proof. Let k be the last iteration of InfoMax for some given sample point. Let $y_0 \in \mathcal{Y}$ be the mode of the posterior $\mathbb{P}(Y \mid \mathcal{A}(t_{1:k}))$. Let T_k be the test selection at iteration k, and assume without loss of generality the outcome observed was $t_k = 1$. We can then express the posterior for $Y = y_0$ as,

$$\begin{aligned} &\mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k})\Big) \\ &= \frac{\mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k-1})\Big)\mathbb{P}\Big(T_k = 1 \mid Y = y_0\Big)}{\mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k-1})\Big)\mathbb{P}\Big(T_k = 1 \mid Y = y_0\Big) + \sum_{y \in \mathcal{Y} \setminus \{y_0\}} \mathbb{P}\Big(Y = y \mid \mathcal{A}(t_{1:k-1})\Big)\mathbb{P}\Big(T_k = 1 \mid Y = y\Big)} \\ &= \frac{\mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k-1})\Big)\mathbb{P}\Big(T_k = 1 \mid Y = y_0\Big)}{\mathbb{P}\Big(T_k = 1 \mid \mathcal{A}(t_{1:k-1})\Big)} \end{aligned} \tag{36}$$

From equation 36 it is clear than $\mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k})\Big)$ would be maximized when the relative contribution of y_0, $\mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k-1})\Big)\mathbb{P}\Big(T_k = 1 \mid Y = y_0\Big)$ to $\mathbb{P}\Big(T_k = 1 \mid \mathcal{A}(t_{1:k-1})\Big)$ is maximized. Thus, the best case scenario is

when T_k is the singleton test "Is $Y = y_0$?", which has almost the entirety of its probability mass of being 1 (Yes) due to y_0, that is,

$$\mathbb{P}\Big(\text{"Is } Y = y_0\text{?"} = 1 \mid \mathcal{A}(t_{1:k-1})\Big) \quad = \mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k-1})\Big)(1-\alpha) + \Big(1 - \mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k-1})\Big)\Big)\alpha. \tag{37}$$

The above equality is obtained by using the fact that for $Y = y_0$, the true outcome D_{T_k} for T_k is 1, but for every other $y \in \mathcal{Y} \setminus \{y_0\}$, the true outcome is 0.

Denote $x := \mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k-1})\Big)$. Then, with the above choice for T_k, we have

$$\mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k})\Big) = \frac{x(1-\alpha)}{x(1-\alpha) + (1-x)\alpha}, \tag{38}$$

which is monotonically increasing in x for $x \in [0, \frac{1}{2}]$.

Now let $\mathcal{T}$ be an arbitrary (δ, γ)-unpredictable set and let $\mathcal{T}' = \mathcal{T} \cup$ "Is $Y = y_0$?". Since $\mathcal{T}$ is (δ, γ)-unpredictable then $\mathcal{T}'$ is (δ, γ)-unpredictable as well. Let us first analyze $\mathcal{T}'$.

Since $\mathcal{T}'$ is (δ, γ)-unpredictable,

$$\begin{aligned}\mathbb{P}\Big(Y = y_0 \mid \mathcal{A}(t_{1:k})\Big) &\leq \frac{\left(\frac{1}{2} + \frac{\delta}{1-2\alpha}\right)(1-\alpha)}{\left(\frac{1}{2} + \frac{\delta}{1-2\alpha}\right)(1-\alpha) + \left(1 - \left(\frac{1}{2} + \frac{\delta}{1-2\alpha}\right)\right)\alpha} \\ &= 1 - \frac{\left(\frac{1}{2} - \frac{\delta}{1-2\alpha}\right)\alpha}{\left(\frac{1}{2} + \frac{\delta}{1-2\alpha}\right)(1-\alpha) + \left(\frac{1}{2} - \frac{\delta}{1-2\alpha}\right)\alpha},\end{aligned} \tag{39}$$

which is attained by setting $x = \left(\frac{1}{2} + \frac{\delta}{1-2\alpha}\right)$. Equation 39 is obtained by considering the following facts. First, $x = \mathbb{P}(D_{T_k} = 1 \mid \mathcal{A}(t_{1:k-1}))$, where recall D_{T_k} is the true (de-noised) outcome for the test $T_k :=$ "Is $Y = y_0$?". Second, from equation 26 it is clear that if $\mathcal{T}'$ is (δ, γ)-unpredictable then there cannot exist a singleton test whose corresponding true outcome has probability, $\mathbb{P}(D_{T_k} = 1 \mid \mathcal{A}(t_{1:k-1})) > \frac{1}{2} + \frac{\delta}{1-2\alpha}$. Combining these two facts along with the monotonicity of the posterior in x gives the first inequality in equation 39.

Finally, since we argued that the singleton test "Is $Y = y_0$?" maximizes the posterior in equation 36, the upper bound in equation 39 will hold for our given arbitrary (δ, γ)-unpredictable $\mathcal{T}$ (which may not contain this single test). This proves the lemma. □

B. Query set details

B.1. CUB-200

Following are the 312 binary questions used in the dataset.

1. Has bill shape::curved (up or down)?
2. Has bill shape::dagger?
3. Has bill shape::hooked?
4. Has bill shape::needle?
5. Has bill shape::hooked seabird?
6. Has bill shape::spatulate?
7. Has bill shape::all-purpose?
8. Has bill shape::cone?
9. Has bill shape::specialized?
10. Has wing color::blue?
11. Has wing color::brown?
12. Has wing color::iridescent?
13. Has wing color::purple?
14. Has wing color::rufous?

15. Has wing color::grey?
16. Has wing color::yellow?
17. Has wing color::olive?
18. Has wing color::green?
19. Has wing color::pink?
20. Has wing color::orange?
21. Has wing color::black?
22. Has wing color::white?
23. Has wing color::red?
24. Has wing color::buff?
25. Has upperparts color::blue?
26. Has upperparts color::brown?
27. Has upperparts color::iridescent?
28. Has upperparts color::purple?
29. Has upperparts color::rufous?
30. Has upperparts color::grey?
31. Has upperparts color::yellow?
32. Has upperparts color::olive?
33. Has upperparts color::green?
34. Has upperparts color::pink?
35. Has upperparts color::orange?
36. Has upperparts color::black?
37. Has upperparts color::white?
38. Has upperparts color::red?
39. Has upperparts color::buff?
40. Has underparts color::blue?
41. Has underparts color::brown?
42. Has underparts color::iridescent?
43. Has underparts color::purple?
44. Has underparts color::rufous?
45. Has underparts color::grey?
46. Has underparts color::yellow?
47. Has underparts color::olive?
48. Has underparts color::green?
49. Has underparts color::pink?
50. Has underparts color::orange?
51. Has underparts color::black?
52. Has underparts color::white?
53. Has underparts color::red?
54. Has underparts color::buff?
55. Has breast pattern::solid?
56. Has breast pattern::spotted?
57. Has breast pattern::striped?
58. Has breast pattern::multi-colored?
59. Has back color::blue?
60. Has back color::brown?
61. Has back color::iridescent?
62. Has back color::purple?
63. Has back color::rufous?
64. Has back color::grey?
65. Has back color::yellow?
66. Has back color::olive?
67. Has back color::green?
68. Has back color::pink?
69. Has back color::orange?
70. Has back color::black?
71. Has back color::white?

72. Has back color::red?
73. Has back color::buff?
74. Has tail shape::forked tail?
75. Has tail shape::rounded tail?
76. Has tail shape::notched tail?
77. Has tail shape::fan-shaped tail?
78. Has tail shape::pointed tail?
79. Has tail shape::squared tail?
80. Has upper tail color::blue?
81. Has upper tail color::brown?
82. Has upper tail color::iridescent?
83. Has upper tail color::purple?
84. Has upper tail color::rufous?
85. Has upper tail color::grey?
86. Has upper tail color::yellow?
87. Has upper tail color::olive?
88. Has upper tail color::green?
89. Has upper tail color::pink?
90. Has upper tail color::orange?
91. Has upper tail color::black?
92. Has upper tail color::white?
93. Has upper tail color::red?
94. Has upper tail color::buff?
95. Has head pattern::spotted?
96. Has head pattern::malar?
97. Has head pattern::crested?
98. Has head pattern::masked?
99. Has head pattern::unique pattern?
100. Has head pattern::eyebrow?
101. Has head pattern::eyering?
102. Has head pattern::plain?
103. Has head pattern::eyeline?
104. Has head pattern::striped?
105. Has head pattern::capped?
106. Has breast color::blue?
107. Has breast color::brown?
108. Has breast color::iridescent?
109. Has breast color::purple?
110. Has breast color::rufous?
111. Has breast color::grey?
112. Has breast color::yellow?
113. Has breast color::olive?
114. Has breast color::green?
115. Has breast color::pink?
116. Has breast color::orange?
117. Has breast color::black?
118. Has breast color::white?
119. Has breast color::red?
120. Has breast color::buff?
121. Has throat color::blue?
122. Has throat color::brown?
123. Has throat color::iridescent?
124. Has throat color::purple?
125. Has throat color::rufous?
126. Has throat color::grey?
127. Has throat color::yellow?
128. Has throat color::olive?

129. Has throat color::green?
130. Has throat color::pink?
131. Has throat color::orange?
132. Has throat color::black?
133. Has throat color::white?
134. Has throat color::red?
135. Has throat color::buff?
136. Has eye color::blue?
137. Has eye color::brown?
138. Has eye color::purple?
139. Has eye color::rufous?
140. Has eye color::grey?
141. Has eye color::yellow?
142. Has eye color::olive?
143. Has eye color::green?
144. Has eye color::pink?
145. Has eye color::orange?
146. Has eye color::black?
147. Has eye color::white?
148. Has eye color::red?
149. Has eye color::buff?
150. Has bill length::about the same as head?
151. Has bill length::longer than head?
152. Has bill length::shorter than head?
153. Has forehead color::blue?
154. Has forehead color::brown?
155. Has forehead color::iridescent?
156. Has forehead color::purple?
157. Has forehead color::rufous?
158. Has forehead color::grey?
159. Has forehead color::yellow?
160. Has forehead color::olive?
161. Has forehead color::green?
162. Has forehead color::pink?
163. Has forehead color::orange?
164. Has forehead color::black?
165. Has forehead color::white?
166. Has forehead color::red?
167. Has forehead color::buff?
168. Has under tail color::blue?
169. Has under tail color::brown?
170. Has under tail color::iridescent?
171. Has under tail color::purple?
172. Has under tail color::rufous?
173. Has under tail color::grey?
174. Has under tail color::yellow?
175. Has under tail color::olive?
176. Has under tail color::green?
177. Has under tail color::pink?
178. Has under tail color::orange?
179. Has under tail color::black?
180. Has under tail color::white?
181. Has under tail color::red?
182. Has under tail color::buff?
183. Has nape color::blue?
184. Has nape color::brown?
185. Has nape color::iridescent?

186. Has nape color::purple?
187. Has nape color::rufous?
188. Has nape color::grey?
189. Has nape color::yellow?
190. Has nape color::olive?
191. Has nape color::green?
192. Has nape color::pink?
193. Has nape color::orange?
194. Has nape color::black?
195. Has nape color::white?
196. Has nape color::red?
197. Has nape color::buff?
198. Has belly color::blue?
199. Has belly color::brown?
200. Has belly color::iridescent?
201. Has belly color::purple?
202. Has belly color::rufous?
203. Has belly color::grey?
204. Has belly color::yellow?
205. Has belly color::olive?
206. Has belly color::green?
207. Has belly color::pink?
208. Has belly color::orange?
209. Has belly color::black?
210. Has belly color::white?
211. Has belly color::red?
212. Has belly color::buff?
213. Has wing shape::rounded-wings?
214. Has wing shape::pointed-wings?
215. Has wing shape::broad-wings?
216. Has wing shape::tapered-wings?
217. Has wing shape::long-wings?
218. Has size::large (16 - 32 in)?
219. Has size::small (5 - 9 in)?
220. Has size::very large (32 - 72 in)?
221. Has size::medium (9 - 16 in)?
222. Has size::very small (3 - 5 in)?
223. Has shape::upright-perching water-like?
224. Has shape::chicken-like-marsh?
225. Has shape::long-legged-like?
226. Has shape::duck-like?
227. Has shape::owl-like?
228. Has shape::gull-like?
229. Has shape::hummingbird-like?
230. Has shape::pigeon-like?
231. Has shape::tree-clinging-like?
232. Has shape::hawk-like?
233. Has shape::sandpiper-like?
234. Has shape::upland-ground-like?
235. Has shape::swallow-like?
236. Has shape::perching-like?
237. Has back pattern::solid?
238. Has back pattern::spotted?
239. Has back pattern::striped?
240. Has back pattern::multi-colored?
241. Has tail pattern::solid?
242. Has tail pattern::spotted?

243. Has tail pattern::striped?
244. Has tail pattern::multi-colored?
245. Has belly pattern::solid?
246. Has belly pattern::spotted?
247. Has belly pattern::striped?
248. Has belly pattern::multi-colored?
249. Has primary color::blue?
250. Has primary color::brown?
251. Has primary color::iridescent?
252. Has primary color::purple?
253. Has primary color::rufous?
254. Has primary color::grey?
255. Has primary color::yellow?
256. Has primary color::olive?
257. Has primary color::green?
258. Has primary color::pink?
259. Has primary color::orange?
260. Has primary color::black?
261. Has primary color::white?
262. Has primary color::red?
263. Has primary color::buff?
264. Has leg color::blue?
265. Has leg color::brown?
266. Has leg color::iridescent?
267. Has leg color::purple?
268. Has leg color::rufous?
269. Has leg color::grey?
270. Has leg color::yellow?
271. Has leg color::olive?
272. Has leg color::green?
273. Has leg color::pink?
274. Has leg color::orange?
275. Has leg color::black?
276. Has leg color::white?
277. Has leg color::red?
278. Has leg color::buff?
279. Has bill color::blue?
280. Has bill color::brown?
281. Has bill color::iridescent?
282. Has bill color::purple?
283. Has bill color::rufous?
284. Has bill color::grey?
285. Has bill color::yellow?
286. Has bill color::olive?
287. Has bill color::green?
288. Has bill color::pink?
289. Has bill color::orange?
290. Has bill color::black?
291. Has bill color::white?
292. Has bill color::red?
293. Has bill color::buff?
294. Has crown color::blue?
295. Has crown color::brown?
296. Has crown color::iridescent?
297. Has crown color::purple?
298. Has crown color::rufous?
299. Has crown color::grey?

300. Has crown color::yellow?
301. Has crown color::olive?
302. Has crown color::green?
303. Has crown color::pink?
304. Has crown color::orange?
305. Has crown color::black?
306. Has crown color::white?
307. Has crown color::red?
308. Has crown color::buff?
309. Has wing pattern::solid?
310. Has wing pattern::spotted?
311. Has wing pattern::striped?
312. Has wing pattern::multi-colored?

B.2. Awa2

Following are the 50 binary questions used in the dataset.

1. Is it black?
2. Is it white?
3. Is it blue?
4. Is it brown?
5. Is it gray?
6. Is it orange?
7. Is it red?
8. Is it yellow?
9. Does it have patches?
10. Does it have spots?
11. Does it have stripes?
12. Is it furry?
13. Is it hairless?
14. Does it have a tough skin?
15. Is it big?
16. Is it small?
17. Is it bulbous?
18. Is it lean?
19. Does it have flippers?
20. Does it have hands?
21. Does it have hooves?
22. Does it have pads?
23. Does it have paws?
24. Does it have long legs?
25. Does it have a long neck?
26. Does it have a tail?
27. Does it have chewteeth?
28. Does it have meatteeth?
29. Does it have buckteeth?
30. Does it have strainteeth?
31. Does it have horns?
32. Does it have claws?
33. Does it have tusks?
34. Is it smelly?
35. Does it fly?
36. Does it hop?
37. Does it swim?
38. Does it burrow tunnels?
39. Does it walks?

40. Is it fast?
41. Is it slow?
42. Is it strong?
43. Is it weak?
44. Is it muscular?
45. Is it bipedal?
46. Is it quadrapedal?
47. Is it active?
48. Is it inactive?
49. Is it nocturnal?
50. Does it hibernate?
51. Is it agile?
52. Does it eat fish?
53. Does it eat meat?
54. Does it eat plankton?
55. Does it eat vegetation?
56. Does it eat insects?
57. Is it a forager?
58. Is it a grazer?
59. Is it a hunter?
60. Is it a scavenger?
61. Is it a skimmer?
62. Is it a stalker?
63. Is it a newworld animal?
64. Is it an oldworld animal?
65. Does it live in the arctic?
66. Is it a coastal animal?
67. Does it live in desert?
68. Does it live in bush?
69. Does it live in plains?
70. Does it live in forest?
71. Does it live in fields?
72. Does it live in jungle?
73. Does it live in mountains?
74. Does it live in ocean?
75. Does it live underground?
76. Does it live in water?
77. Does it live in a tree?
78. Does it live in a cave?
79. Is it fierce?
80. Is it timid?
81. Is it smart?
82. Does it live in a group?
83. Is it solitary?
84. Does it make nests?
85. Is it domestic?