Estimating Mutual Information from Average Classification Error

Charles Zheng and Yuval Benjamini

September 9, 2016

Abstract

Multivariate pattern analyses approaches in neuroimaging are fundamentally concerned with investigating the quantity and type of information processed by various regions of the human brain; typically, estimates of classification accuracy are used to quantify information. While a extensive and powerful library of methods can be applied to train and assess classifiers, it is not always clear how to use the resulting measures of classification performance to draw scientific conclusions: e.g. for the purpose of evaluating redundancy between brain regions. An additional confound for interpreting classification performance is the dependence of the error rate on the number and choice of distinct classes obtained for the classification task. In contrast, mutual information is a quantity which is in principle defined independently of the experimental design, and has ideal properties for comparative analyses. One obstacle to wider use of mutual information is the difficulty of estimating mutual information in high-dimensional neuroimaging data. We propose a new estimator of mutual information based on the concept of "average Bayes error." Since the error of a classifier can be used to infer an upper bound on the average Bayes error, we develop a lower confidence bounds for mutual information based on a new theoretical lower bound on mutual information as a function of average Bayes error. Additionally, we develop an estimator of mutual information based on separate high-dimensional asymptotic theory. We demonstrate the utility of our approach in simulated and real data examples.

1 Introduction

A fundamental challenge of computational neuroscience is to understand how information about the external world is processed and represented in the brain. Each individual neuron aggregates the incoming information into a single sequence of spikes–an output which is too simplistic by itself to capture the full complexity of sensory input. Only by combining the signals from massive ensembles of neurons is it possible to reconstruct our complex representation of the world. Nevertheless, neurons form hierarchies of specialization within neural circuits, which are further organized in various specialized regions of the brain. At the lowest level of the hierarchy–individual neurons, it is possible to infer and interpret the functional relationship between a neuron and stimulus features of interest using single-cell recording technologies. Due to the inherent stochasticity of the neural output, it is natural to view the neuron as a noisy channel, and use mutual information to quantify how much of the stimulus information is encoded by the neuron. Moving up the hierarchy to the the macroscale level of organization in the brain requires both different experimental methodologies and new approaches for summarizing and inferring measures of information in the brain.

Shannon's mutual information $I(X;Y)$ is fundamentally a measure of dependence between random variables X and Y, and is defined as

$$I(X;Y) = \int p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dxdy.$$

Various properties of $I(X;Y)$ make it ideal for quantifying the information between a random stimulus X and the signaling behavior of an ensembles of neurons, Y (Borst 1999). A leading metaphor is that of a noisy communications channel; the mutual information describes the rate at which Y can communicate bits from X. This framework is well-suited for summarizing the properties of a single neuron coding external stimulus information; indeed, experiments studying the properties of a single or a small number of neurons often make use of the concept of mutual information in summarizing or interpreting their results (Quiroga 2009). However, estimating mutual information for multiple channels require large and over-parameterized generative models. For instance, one can tractably estimate mutual information by assuming a multivariate Gaussian model: however, this approach essentially assumes a linear relationship between the input and output, and hence fails to quantify nonlinear dependencies. As the complexity of stimuli and

the number of output channels increases, these models are hard or impossible to estimate without gross over-fitting. As new technologies for simultaneous measurement of multiple brain regions developed, such as functional MRI, it became increasingly difficult to quantify information at such scales under the classical approach.

Machine learning algorithms showed a way forward: a seminal work by Haxby (2001) proposed to quantify the information in multiple channels by measuring how well the stimulus can be identified from the brain responses, in what is known as "multivariate pattern analysis" (MVPA). To demonstrate that a particular brain region responds to a certain type of sensory information, one employs supervised learning to build a classifier that classifies the stimulus class from the brain activation in that region. Classifiers that achieve above-chance classification accuracy indicate that information from the stimulus is represented in the brain region. In principle, one could just as well test the statistical hypothesis that the Fisher information or mutual information between the stimulus and the activation patterns is nonzero. But in practice, the machine learning approach enjoys several advantages: First, it is invariant to the parametric representation of the stimulus space, and is opportunistic in the parameterization of the response space. This is an important quality for naturalistic stimulus-spaces, such as faces or natural images. Second, it scales better with the dimensionality of both the stimulus space and the responses space, because a slimmer discriminative model can be used rather than a fully generative model.

Nevertheless, classification error is problematic for quantifying the strength of the relation between stimulus and outputs due to its arbitrary scale and strong dependence on experimental choices. Classification accuracy depends on the particular choice of stimuli exemplars employed in the study and the number of partitions used to define the classes for the classification task. The difficulty of the classification task depends on the number of classes defined: high classification accuracy can be achieved relatively easily by using a coarse partition of stimuli exemplars into classes. In a meta-analysis on visual decoding, Coutanche et al (2016) quantified the strength of a classification study using the formula

$$\text{decoding strength} = \frac{\text{accuracy} - \text{chance}}{\text{chance}}.$$

Such an approach may compensate for the differences in accuracy due purely to choice of number of classes defined; however, no theory is provided to

justify the formula. In contrast, mutual information has ideal properties for quantitatively comparing information between different studies, or between different brain regions, subjects, featurization models, or modalities. Not only is the mutual information defined independently of the arbitrary definition of stimulus classes (albeit still dependent on an implied distribution over stimuli), it is even meaningful to discuss the difference between the mutual information measured for one system and the mutual information for a second system.

Hence, a popular approach which combines the strengths of the machine learning approach and the advantages of the information theoretic approach is to obtain a lower bound on the mutual information by using the confusion matrix of a classifier. This is the most popular approach for estimating mutual information in neuroimaging studies, but suffers from known shortcomings (Gastpar 2010, Quiroga 2009). The idea of linking classification performance to mutual information dates back to the beginnings of information theory: Shannon's original motivation was to characterize the minimum achievable error probability of a noisy communication channel. More explicitly, Fano's inequality provides a lower bound on mutual information in relation to the optimal prediction error, or Bayes error. Fano's inequality can be further refined to obtain a tighter lower bound on mutual information (Tebbe and Dwyer 1968.) However, all approaches to date can only obtain a lower bound on the mutual information from classification error. In practice, the bound obtained may be a vast underestimate.

In this paper, we propose a new way to link classification performance to the implied mutual information. To create this link we need to overcome the arbitrary choice of exemplars, and the arbitrary number of classes K. Towards this end, we define a notion of K-class *average Bayes error* which is uniquely defined for any given stimulus distribution and stochastic mapping from stimulus to response. The K-class average Bayes error is the expectation of the Bayes error (the classification error of the optimal classifier) when K stimuli exemplars are drawn i.i.d. from the stimulus distribution, and treated as distinct classes. Hence the average Bayes error can in principle be estimated if the appropriate randomization is employed for designing the experiment.

Our main theoretical contributions are (i) the derivation of a tight lower bound on mutual information as a function of k-class average Bayes error, and (ii) the derivation of an asymptotic relationship between the K-class average Bayes error and the mutual information. Although the K-class average

Bayes error is defined independently of the particular choice of stimuli, the quantity still depends on the choice of number of classes, K. Mutual information provides a quantification of information that does not depend on K, allowing more flexible comparisons and easier interpretation. Our method allows estimates of the K-class Bayes error to be translated into estimate of the mutual information, and this resulting estimator of mutual information will be asymptotically independent of the choice of number of classes K.

2 Framework

2.1 Notation

Functionals of distributions, such as expectation $\mathbf{E}[\cdot]$ or mutual information $\mathrm{I}[\cdot]$, can be written as functionals of density functions:

$$\mathbf{E}[g] \stackrel{D}{=} \mathbf{E}[g(x)] \stackrel{D}{=} \int g(x)dx$$

$$\mathrm{H}[g] \stackrel{D}{=} \int g(x) \log g(x)dx$$

$$\mathrm{I}[p] \stackrel{D}{=} \mathrm{I}[p(x,y)] = \int p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dxdy$$

or as functionals of random variables: $\mathbf{E}(X)$, $\mathrm{H}(X)$, $\mathrm{I}(X,Y)$. Square brackets are used when the arguments are density functions, and parentheses are used when the arguments are functions of random variables.

2.2 Model

The following simplified model captures the essence of many neuroimaging studies. Let $\mathcal{X}$ be a space of stimuli, represented by q-dimensional vectors. In the design stage of the experiment, a set of k stimuli exemplars $\{x^{(1)}, \ldots, x^{(k)}\}$ are selected, and assigned into a T-length sequence of stimuli $(x^{(i_1)}, \ldots, x^{(i_T)})$ to be presented to the subject. In the execution of the experiment, an activation pattern or *response* y^t is obtained for each of the stimuli presentations $x^{(i_t)}$. Generally y^t is a p-dimensional vector, representing activity levels of p disjoint brain regions. To simplify, assume that each of the k stimuli is presented a total of r times in the sequence; further assume that

the responses to each stimulus presentation are conditionally independent, hence the ordering of the sequence does not matter. Henceforth we can let $y^{(i),j}$ denote the response to the jth presentation of the stimulus $x^{(i)}$.

While in many studies, the stimuli exemplars $x^{(1)}, \ldots, x^{(k)}$ are chosen somewhat arbitrarily, more types of inferences can be drawn if randomization is employed to choose the stimuli. Based on this distinction, we discuss two types of experimental designs:

1. *Fixed stimuli design.* The exemplars $x^{(1)}, \ldots, x^{(k)}$ are treated as fixed.

2. *Randomized stimuli design.* The experimenter first specifies a space of stimuli $\mathcal{X}$ and a probability distribution $p(x)$ over that space. Then, k exemplars are drawn i.i.d. from that space,

$$X^{(1)}, \ldots, X^{(k)} \overset{iid}{\sim} p(x).$$

As we will discuss, the randomized stimuli design allows the possibility of *generalizing* beyond the chosen exemplars; i.e. drawing inferences about the entire joint distribution $p(x, y)$.

And while we draw a clear-cut distinction between the fixed design setting and randomized design setting, in practice, the boundaries are frequently blurred. Indeed, few researchers in the field actually employ an explicit randomization mechanism to choose the stimuli, yet they may think that their stimuli are "effectively random". For stimulus categories such as faces or natural images, it is not clear how to specify a probability distribution $p(x)$ in the first place. Yet, a practical approach is to choose stimuli in an *ad-hoc* manner with the intent of obtaining a representative set, and to treat the stimuli as having been randomly sampled from some implicitly defined $p(x)$ *after the fact.* In such studies, the convenient fiction of a randomized design is employed to formalize the process of generalizing experimental results. Such a cavalier attitude towards randomization may seem concerning by the standards of clinical trials, but it is appropriate for the goals of typical imaging studies, where the ultimate objective to discover some qualitative properties of human perception and cognition.

2.3 Fixed stimuli design

We begin by describing a fixed stimuli design, and the types of inferences that can be made using machine learning tools. The general approach we describe

was first introduced to the neuroimaging community by Haxby (1999), but its foundational concepts belong to the area of statistical learning (Hastie et al. 2008).

Let $\mathcal{X} \subset \mathbb{R}^q$ and $\mathcal{Y} \subset \mathbb{R}^p$; For every $x \in \mathcal{X}$, let $p_x(y)$ be a probability density on $\mathcal{Y}$. Take a subset $\{x^{(1)}, \ldots, x^{(k)}\} \subset \mathcal{X}$. Let Y_i^j be a random vector distributed according to density $p_{x^{(i)}}(y)$; and let $Y^{(i),j}$ be independent for $i = 1, \ldots, k$ and $j = 1, \ldots, r$.

In MVPA, one carries out a classification task to assess whether y contains information about x. Formally, a classification rule is any (possibly stochastic) mapping $f : \mathcal{Y} \to \{1, \ldots, k\}$. The *generalization accuracy* of the classification rule for classes $x^{(1)}, \ldots, x^{(k)}$ is

$$\mathrm{GA}(f) = \frac{1}{k} \sum_{i=1}^{k} \Pr[f(Y) = i | X = x^{(i)}].$$

A trivial classification rule which outputs the result of a k-sided die roll for all inputs y would achieve a generalization accuracy of $\mathrm{GA} = \frac{1}{k}$. Conversely, even a single counterexample with $\mathrm{GA} > \frac{1}{k}$ is indicative that y contains nonzero information about x. Hence, in order to demonstrate that y is informative of x, one tests the null hypothesis

$$H_0 : \mathrm{GA}(f) = \frac{1}{k}$$

versus the alternative

$$H_1 : \mathrm{GA}(f) > \frac{1}{k}.$$

Rejecting the null hypothesis for a given classification rule f can be taken as evidence that y is informative of x.

We have not yet specified how any classification rule f is to be obtained. Unless one has strong prior knowledge about the nature of the brain encoding, it is necessary to choose the function f in a data-dependent way in order to obtain a reasonable classification rule. A wide variety of machine learning algorithms exist for "learning" good classification rules f from data. We use the terminology *classifier* to refer to any algorithm which takes data as input, and produces a classification rule f as output. The following discussion makes it necessary for us to make a precise distinction between the *classifier* and the *classification rule* it produces, and our usage of the terms may differ

from the standard in the literature. Mathematically speaking, the classifier is a functional which maps a set of observations to a classification rule,

$$\mathcal{F} : \{(x^1, y^1), \ldots, (x^m, y^m)\} \mapsto f(\cdot).$$

The data $(x^1, y^1), \ldots, (x^m, y^m)$ used to obtain the classification rule is called *training data.* When the objective is to obtain the best possible classification rule, as is the case in diagnostic settings, it is optimal to use all of the availible data to train the classifier. However, when the goal is to obtain *inference* about the performance of the classification rule, it beomes necessary to split the data into two independent sets: one set to train the classifier, and one to evaluate the performance. The reason that such a splitting is necessary is because using the same data to test and train a classifier introduces significant bias into the empirical classification error.

In *data-splitting*, one creates a *training set* consisting of r_1 repeats per class,

$$\{(x^{(1)}, y^{(1),1}), \ldots, (x^1, y^{(1),r_1}), \ldots, (x^{(k)}, y^{(k),1}), \ldots, (x^{(m)}, y^{(m),r_1})\}$$

and a *test set* consisting of the remaining $r_2 = r - r_1$ repeats.

$$\{(x^{(1)}, y^{(1),r_1+1}), \ldots, (x^{(1)}, y^{(1),r}), \ldots, (x^{(k)}, y^{(k),r_1+1}), \ldots, (x^{(k)}, y^{(k),r_1})\}.$$

One inputs the training data into the classifier to obtain the classification rule f,

$$f = \mathcal{F}(\{(x^{(1)}, y^{(1),1}), \ldots, (x^{(1)}, y^{(1),r_1}), \ldots, (x^{(k)}, y^{(k),1}, \ldots, (x^{(k)}, y^{(k),r_1})\}).$$

The test statistic of interest is the test error, defined as

$$\widehat{\mathrm{GA}} = \frac{1}{kr_2} \sum_{i=1}^{k} \sum_{j=r_1+1}^{r} \mathrm{I}(f(y^{(i),j}) \neq i).$$

Due to the conditional independence of the training set and test set, $\widehat{\mathrm{GA}}$ is an unbiased estimate of GA. Hence various approaches can be used to obtain a threshold c_α such that $\Pr[\widehat{\mathrm{GA}} > c_\alpha] \leq \alpha$ holds (approximately) under the null hypothesis. These approaches include permutation tests, tests based on a universal variance bound, or the generalized likelihood ratio test. The hypothesis H_0 is then rejected at level α if $\widehat{\mathrm{GA}} > c_\alpha$.

There is also a way to improve on data-splitting, by using *cross-validation*. Essentially, one applies the data-splitting approach multiple times, with different training and test partitions, and aggregates the results. Cross-validation can be used to reduce the variance of the estimated GA and hence improve the power of the test, and it can also be used to obtain *confidence intervals* for the estimate. We discuss such a cross-validation in greater detail in section 5.

While tests of the generalization error suffice to establish the presence of information, the generalization error is less satisfactory as a measure of the information between X and Y, because GA depends on the classification rule f obtained–but since the performance of the classifier may vary depending on the choice of model (k-nearest neighbors, SVM, etc.) and the choice of tuning parameters, the quantity GA is therefore not uniquely defined. To resolve the ill-definedness of the generalization accuracy, we define the Bayes accuracy, which is simply the *optimal* generalization accuracy

$$\mathrm{BA} = \sup_f \mathrm{GA}(f).$$

Due to Bayes' theorem, the optimal classification rule f^* which achieves the Bayes error can be given explicitly: it is the maximum a posteriori (MAP) rule

$$f^*(y) = \mathrm{argmax}_{i=1}^k \; p(y|x^{(i)}).$$

Of course, it is not possible to construct this rule in practice since the joint distribution is unknown. Instead, a reasonable approach is to try a variety of classifiers, producing rules $f_1, \ldots, f_m$, and taking the best generalization accuracy as an estimate of the Bayes accuracy. We give more details of this approach in the Discussion.

This way one can draw inferences about information between the fixed ensemble $x_1, ..., x_k$ and the brain regions being examined. But in many cases it is desired to generalize the results to other stimuli that were not included in the experiment. In order to do this, we need to be able to assume a randomized design.

2.4 Randomized stimuli design

Under the assumption of randomized design, we treat the stimuli $x_1, \ldots, x_k$ as independent draws from some distribution $p(x)$. Defining the joint distri-

bution of the stimulus X and the response Y as

$$p(x, y) = p(x)p(y|x),$$

one can also define the mutual information

$$I(X;Y) = \int p(x, y) \log \frac{p(x, y)}{p(x)p(y)} dxy.$$

In contrast the case of fixed design, the randomized design framework provides a principled way of making inferences about the population of stimuli exemplars, beyond the particular exemplars that were chosen for the study. This is particularly relevant for complex stimuli, for which the number of distinct stimuli species (e.g. distinct faces) could be astronomically large (i.e. the number of faces that a human could distinguish.) The concept of information $I(X;Y)$ captures the notion of the "complexity" of the stimuli representation. By inferring the information $I(X;Y)$, we make inferences about the complexity of the representation in the given brain region Y.

Methodologies for estimating mutual information under the assumption of randomization have been studied in (Borst 1999), (Nelken 2005) and most notably (Gastpar 2010); the latter studies the problem under an identical setup to ours. In addition to these approaches, one can obtain a lower bound on the information using the mutual information of the confusion matrix (Treves et al), (Quiroga 2009). In Section 3, we present our methodology for inferring mutual information under the randomized stimuli design setting.

Before presenting our approach, we first define the notion of average Bayes accuracy, which is fundamental to our approach. The motivation for defining average Bayes accuracy is the fact that the quantity BA is not a parameter of the joint distribution $p(x, y)$, but rather depends on the specific exemplars $x_1, \ldots, x_k$ selected; hence, one may write $\mathrm{BA}(x_1, \ldots, x_k)$ to emphasize this dependence. The k-class *average* Bayes accuracy, on the other hand, is defined uniquely for any joint distribution,

$$\mathrm{ABA}_k[p(x, y)] = \mathbf{E}[\mathrm{BA}(X_1, \ldots, X_k)], \tag{1}$$

where $X_1, \ldots, X_K$ are drawn i.i.d. from $p(x)$. Since we know the Bayes classification rule, we can write the k-class average Bayes accuracy explicitly:

$$\mathrm{ABA}_k[p(x, y)] = \frac{1}{k} \int \left[\prod_{i=1}^{k} p(x_i) dx_i \right] \int dy \max_i p(y|x_i).$$

Our theoretical contributions will be to obtain (i) a lower bound on mutual information $\mathrm{I}[p(x,y)]$ as a function of average Bayes accuracy $\mathrm{ABA}_k[p(x,y)]$ for any continuous joint density function $p(x,y)$, (ii) an asymptotic functional relationship

$$\mathrm{ABA}_k[p] = \pi_k(\sqrt{2\mathrm{I}[p]})$$

in a particular limit where the dimensionality of X and Y grow while $\mathrm{I}(X,Y)$ converges to a finite limit, and (iii) derivation of data-based lower confidence bounds based on these results.

3 Lower bound

[NOTE: See notes info 2.]

4 Asymptotic Theory

[NOTE: Adapt from NIPS paper.]

5 Applications

Having established the theoretical basis of our method, the current section is dedicated to discussing applications and practical issues. In contrast to the previous section, our focus in this section will be methodological rather than theoretical.

Within the framework of randomized designs, a number of machine-learning-based modeling approaches can be applied to assess informativity. The most common approaches are based on *classification* and *regression*, but a third approach, *identification*, combines the strengths of both classification and regression.

The different machine-learning-based modeling approaches excel in different experimental settings. Classification-based approaches require large amounts of training data and test data for a small number of stimuli categories. Therefore, an investigator intending to use classification will design the experiment to have a large number of repeats relative to the number of stimulus categories.

In contrast, in the identification-based approach, it is not necessary to have multiple repeats for each stimulus exemplar. On the other hand, it *is*

necessary to have a parameterization of the stimulus exemplars. The success of the identification approach depends heavily on having a parameterization which captures the relevant features of the stimulus–this can be a difficult task when studying complex stimuli such as faces or natural images.

While we described the classification approach in section 2, we describe the application of our method using classifiers in section 4.1. Then, in section 4.2, we will introduce the identification task, and how our method can also be combined with the identification-based approach.

5.1 Classification setting

Recall the notation used in section 2.1: the k stimuli exemplars are denoted $\{x^{(1)}, \ldots, x^{(k)}\}$ and the r responses for the ith class are given by $y^{(i),1}, \ldots, y^{(i),r}$.

For a given classifier $\mathcal{F}$, the classification rule f varies randomly depending on the input. One can define the average generalization error

$$e_{AGE} = \mathbf{E}[e_{gen}(\mathcal{F}(\{(x^{(1)}, Y^{(1),1}), \ldots, (x^{(1)}, Y^{(1),r_1}), \ldots, (x^{(k)}, Y^{(k),1}, \ldots, (x^{(k)}, Y^{(k),r_1})\}))]$$

where the expectation is taken over the sampling of the training data *conditional* on fixing the stimuli $x^{(1)}, \ldots, x^{(r)}$. The procedure we suggest for obtaining a confidence interval for $\hat{I}(X;Y)$ is to first obtain a confidence interval for the average generalization error, $[\underline{e}, \bar{e}]$, then apply the inversion formula to construct the interval[1]

$$C = [\frac{1}{2}(\pi_k^{-1}(\bar{e}))^2, \frac{1}{2}(\pi_k^{-1}(\underline{e}))^2].$$

We can use cross-validation to construct such a confidence interval C. For a large number of resampling trials B, let $S_1, \ldots, S_B$ be random subsets of the data such that each S_i consists of r_1 repeats for each stimulus[2]. Let $\bar{S}_i$

[1]Note that the theory established in section 3.2 does not apply to the coverage properties of C, since we used a confidence interval for e_{AGE} rather than e_{Bayes} to construct the interval. Despite this limitation, the resulting interval C still serves the useful purpose of conveying the uncertainty about the inferred information.

[2]That is, let $\rho^{i,j}(\ell)$ be a random injective mapping from $\{1, \ldots, r_1\}$ to $\{1, \ldots, r\}$ for each $i = 1, \ldots, B$ and $j = 1, \ldots, k$. The $\rho^{i,j}$ are i.i.d. from the uniform distribution on such mappings. Define $S_i = \{(x^{(j)}, y^{(j),\rho^{i,j}(\ell)}) | 1 \leq \ell \leq r \text{ and } 1 \leq j \leq k\}$. Meanwhile, define $\bar{S}_i = \{(x^{(j)}, y^{(j),\ell}) | 1 \leq j \leq k, 1 \leq \ell \leq r\} \setminus S_i$.

denote the complement of S_i in the data. For each $i = 1, \ldots, B$, construct a classification rule $f^{(i)}$ by inputting the set S_i as training data for the classifier,

$$f^{(i)} = \mathcal{F}(\{(x, y) : (x, y) \in S_i\})$$

Define the ith estimate of test error as

$$e_i = \frac{1}{kr_2} \sum_{j=1}^{k} \sum_{(x^{(j)}, y) \in \bar{S}_i} I(f(y) \neq j).$$

Order the estimates so that $e_{(1)} \leq e_{(2)} \leq \cdots \leq e_{(B)}$.

5.2 Identification setting

6 Results

6.1 Simulation

Multiple-response logistic regression model

$$X \sim N(0, I_p)$$

$$Y \in \{0, 1\}^q$$

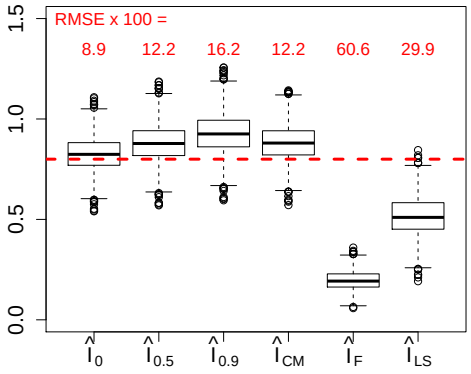
$$Y_i | X = x \sim \text{Bernoulli}(x^T B_i)$$

where B is a $p \times q$ matrix.

Methods.

- Nonparametric: $\hat{I}_0$ naive estimator, $\hat{I}_\alpha$ anthropic correction.
- ML-based: $\hat{I}_{CM}$ confusion matrix, $\hat{I}_F$ Fano, $\hat{I}_{LS}$ low-SNR method.

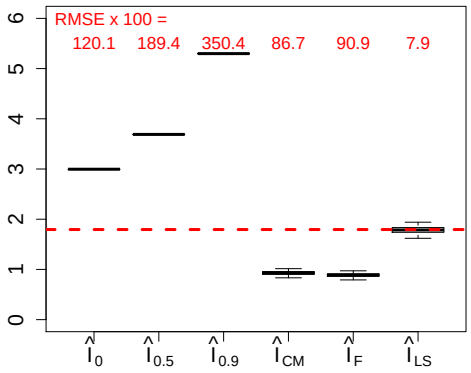
Sampling distribution of $\hat{I}$ for $\{p = 3,\ B = \frac{4}{\sqrt{3}} I_3,\ K = 20,\ r = 40\}$.
True parameter $I(X; Y) = 0.800$ *(dotted line.)*



Naïve estimator performs best! $\hat{I}_{LS}$ not effective.

Sampling distribution of $\hat{I}$ for $\{p = 50,\ B = \frac{4}{\sqrt{50}} I_{50},\ K = 20,\ r = 8000\}$.

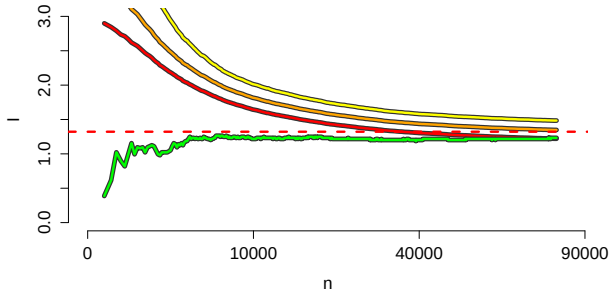
True parameter $I(X;Y) = 1.794$ *(dashed line.)*



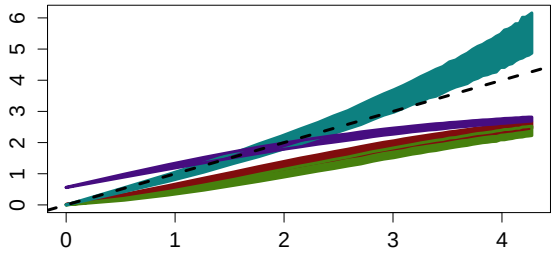
Non-parametric methods extremely biased.

Estimation path of $\hat{I}_{LS}$ and $\hat{I}_\alpha$ as n ranges from 10 to 8000.

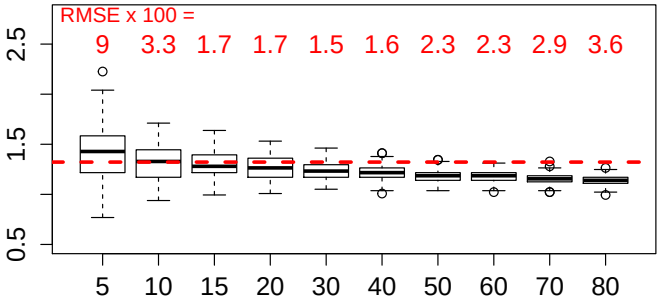
$\{p = 10,\ B = \frac{4}{\sqrt{10}} I_{10},\ K = 20\}$. True parameter $I(X;Y) = 1.322$ *(dashed line.)*



Estimated $\hat{I}$ vs true I.

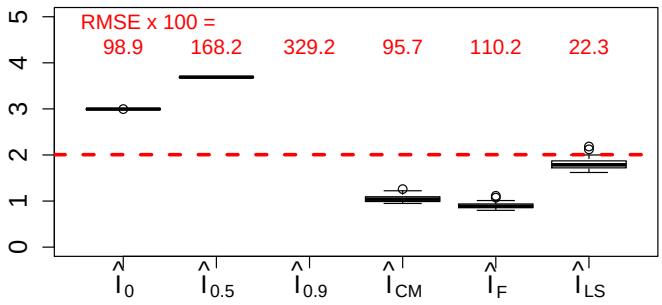


Sampling distribution of $\hat{I}_{LS}$ for $\{p = 10,\ B = \frac{4}{\sqrt{10}} I_{10},\ N = 80000\}$, and $K = \{5, 10, 15, 20, \ldots, 80\}$, $r = N/k$.
True parameter $I(X;Y) = 1.322$ *(dashed line.)*



Decreasing variance as K increases. Bias at large and small K.
$p = 20$ and $q = 40$, entries of B are iid $N(0, 0.025)$.
$K = 20$, $r = 8000$, true $I(X;Y) = 1.86$ *(dashed line.)*

Sampling distribution of $\hat{I}$.



6.2 Application to data

Kay et al. employed a randomized stimuli design in their 2008 paper, "Identifying Natural Images from Human Brain Activity." The experiment was designed in order to investigate how visual information from natural images is encoded in the V1 through V3 brain regions. The stimulus space, $\mathcal{X}$, consists of 128×128-pixel grayscale photographs. The response data consists of BOLD response in regions V1, V2, and V3 from a single subject. The raw time series were processed to yield a single averaged response vector $y^{(i)}$ for each stimulus $x^{(i)}$, for $i = 1, \ldots, 1870$. The dimensionality of $y^{(i)}$ varies depending on which regions of interest we are discussing, and whether we consider a subset of the voxels in those regions. Let v denote the dimensionality (number of voxels) of y.

Let $x^{(i)}$ denote the native 128×128-pixel representation of the image (i.e. a 16384-dimensional vector with entries between 0 and 1.) One of the goals of the Kay et al. paper is to evaluate competing encoding models. In the context of the study, an *encoding model* is a vector-valued function from the stimulus to a p-dimensional space,

$$\vec{g}(x) = (g_1(x), \ldots, g_p(x)).$$

One of the encoding models studied by Kay et al. is the Gabor wavelet pyramid, $\vec{g}_{Gabor}$, with $p = 10921$. Using a *training* subset of the stimulus-response pairs (x_i, y_i), $i = 1, \ldots, 1750$, Kay et al. fit a regularized linear model

$$y^{(i)} \approx B^T \vec{g}(x^{(i)})$$

where B is a $10921 \times v$-dimensional matrix, which is to be fit to the data. The resulting model is used to obtain predictions $\hat{y}^{(i)}$ for the remaining validation subset, $i = 1751, \ldots, 1870$, by

$$\hat{y}^{(i)} = B^T \vec{g}(x^{(i)}).$$

Kay et al. introduced the novel idea of assessing the accuracy of these predictions using a *classification*-based metric, rather than a usual metric for prediction error such as mean-squared error. One treats the 120 validation stimuli $x_{1751}, \ldots, x_{1870}$ as distinct classes. Next, for each $i = 1751, \ldots, 1870$, one predicts the class of $y^{(i)}$, by correlating the observed $y^{(i)}$ with the predictions $\hat{y}^{(j)}$. In other words, let f be the mapping from the observed response to the predicted class: f is given by

$$f(y) = \text{argmax}_{j=1751}^{1870} \text{Cor}(y, \hat{y}^{(j)}).$$

The test error is

$$e_{test} = \frac{1}{120} \sum_{i=1751}^{1870} I(i \neq f(y^{(i)})).$$

This test error can be computed using different encoding models $\vec{g}$ and different regions of the brain, or different-sized subsets of voxels. One can therefore get a sense of the differences between encoding models or between brain regions by comparing the resulting test errors.

While differing from a conventional classification task, the Kay et al. *identification* task can also be treated using our framework. An *identification rule* (as opposed to a classification rule) is defined by a function f which takes arguments $\{y, x^{(1)}, \ldots, x^{(k)}\}$ where k can be any positive integer. The function f returns an integer from 1 to k,

$$f(y, x^{(1)}, \ldots, x^{(k)}) \in \{1, \ldots, k\}.$$

One can study competing encoding models by constraining identification rules to using features provided by the encoding, i.e. only considering functions

$$f(y, \vec{g}(x^{(1)}), \ldots, \vec{g}(x^{(k)})) \in \{1, \ldots, k\}.$$

The generalization error for the identification task is defined as

$$e_{gen}(f) = \Pr[f(Y, \vec{g}(x^{(1)}), \ldots, \vec{g}(x^{(k)})) \neq i | Y \sim p(y|x^{(i)})],$$

Now, the Bayes rule will be specific to the encoding model

$$e_{Bayes}(\vec{g}(x^{(1)}), \ldots, \vec{g}(x^{(k)})) = \inf_f \Pr[f(Y, \vec{g}(x^{(1)}), \ldots, \vec{g}(x^{(k)})) \neq i | Y \sim p(y|x^{(i)})].$$

As before, the Bayes identification rule which attains the minimal error can be given explicitly,

$$f_{Bayes}(y, \vec{g}(x^{(1)}), \ldots, \vec{g}(x^{(k)})) = \text{argmin}_{i=1}^{k} p(y|\vec{g}(x^{(i)})).$$

The average Bayes error can be defined as before, but is specific to the encoding model:

$$e_{ABE}(\vec{g}) = \mathbf{E}[e_{Bayes}(\vec{g}(X^{(1)}), \ldots, \vec{g}(X^{(k)}))]$$

Our asymptotic theory links average Bayes error to information and is *not specific* to the whether the task is based on identification or classification. Hence, the approximation

$$e_{ABE,k}(\vec{g}) \approx \pi_k(\sqrt{2I(\vec{g}(X);Y)})$$

still holds in the appropriate regime.

However, one adjustment that needs to be made is how to construct point estimates or confidence intervals for e_{ABE}. Here we will limit our discussion to confidence intervals. The approach taken is to use cross-validation to construct the confidence interval.

Let N be the total number of stimuli exemplars. Choose $T < N$ and $k < N-T$; T denotes the number of *training* exemplars and k controls the number of *test* exemplars.

Set B, the number of Monte Carlo simulations, to be a large number. Let $\beta \in [0,1]$ be a Bayesian smoothing parameter (typically β is close to 0). For each $b = 1, \ldots, B$, define the bth sampled test error e_b by:

1. Sample T stimuli exemplars without replacement from the library of N exemplars.

2. Train an identification rule f_b using the training set pairs $(\vec{g}(x^{(i)}), y^{(i)})$.

3. Let $S_1, \ldots, S_M$ denote all possible subsets of size k from the remaining $N-T$ exemplars.

4. For each subset $S_j = \{x^{(i_{j,1})}, \ldots, x^{(i_{j,k})}\}$, compute the test error $e_{test,b,j}$ via

$$e_{test,b,j} = \frac{1}{k} \sum_{\ell=1}^{k} I(i_{j,\ell} \neq f(y^{(i_{j,\ell})}, \vec{g}(x^{(i_{j,1})}), \ldots, \vec{g}(x^{(i_{j,k})})))$$

5. Set $e_b = \beta \frac{k-1}{k} + (1-\beta) \frac{1}{M} \sum_{j=1}^{M} e_{test,b,j}$.

For most identification rules, e_b can be computed in $O(N^2)$ time by using the hypergeometric sampling formula; e.g., see Nishimoto (2011) for details. Setting the parameter $\beta > 0$ is typically recommended in order to reduce variance in the procedure.

Now, choosing a level α, let $\bar{e}(\vec{g})$ be the $(1-\alpha/2)$th quantile and $\underline{e}(\vec{g})$ be the $(\alpha/2)$th quantile of $\{e_b\}_{b=1}^{B}$. Then one constructs the confidence interval

$$[\frac{1}{2}(\pi_k^{-1}(\bar{e}))^2, \frac{1}{2}(\pi_k^{-1}(\underline{e}))^2].$$

for $I(\vec{g}(X);Y)$.

Figure 1: Confidence intervals for $I(\vec{g}_{Gabor}(X); Y)$, where Y are subsamples of size $\{100, 200, \ldots, 600\}$ voxels from V1.

Figure 2: Confidence intervals for $I(\vec{g}_{reduced}(X); Y)$, where Y are subsamples of size $\{100, 200, \ldots, 600\}$ voxels from V1.

We computed such confidence intervals for $\vec{g}_{Gabor}$, taking Y to be random subsamples of size $\{100, 200, \ldots, 600\}$ voxels from V1, and also varying the parameter k within the procedure. The confidence intervals are shown in Figure ?.

Using the same voxel subsamples and parameters k, but now using a reduced encoding model $\vec{g}_{reduced}$ with $p = 681$, we computed confidence intervals for $I(\vec{g}_{reduced}(X); Y)$.

7 Discussion

Since the Bayes error is the large-sample limit of the achieved classification error, a promising approach is to perform classification using differently sized subsamples of the training data, producing a plot of classification error versus sample size–a "learning curve." One can then extrapolate the learning curve to estimate the Bayes error (Cortes et al. 1994.)However, much work remains to develop rigorous methodology for estimating Bayes error, and so we leave this first issue for future work.

We should keep in mind that the definition of mutual information depends on the specific class of stimulus that is of interest in the experiment. The Bayes error itself depends on the choices made by the experimenter in regards to the stimuli exemplar chosen in the experiment, and the decision of how to partition those exemplars in the classification task. For example, Nishimoto et al classified segments of a movie clips based on activation patterns, but the definition of the classification task, and the achievable classification performance, depends not only on the particular move clips used in the experiment, but also the choice of time interval used to define discrete classes: defining each class to be a 1sec segment of movie results in more distinct classes and lower classification accuracy than defining each class to be a 4sec segment of movie. The Bayes error, and any estimate of mutual information based on the Bayes error, would therefore be necessarily dependent on the experimental parameters.

Estimating e_{Bayes} is much more difficult in practice than estimating e_{gen}; however, nonparametric estimators of e_{Bayes} have been proposed in the literature. Cover (1969) first noted that the leave-one-out error of 1-nearest neighbor, e_{1nn},

asymptotically bounds the Bayes error in the sense that

$$\frac{1}{2}\mathbf{E}[e_{1nn}] \leq e_{Bayes} \leq \mathbf{E}[e_{1nn}].$$

Later, Fukunaga and Kessel (1971) and Fralick and Scott (1971) both proposed a consistent estimator of e_{Bayes} using the test error from kernel density estimation-based classifiers. Fukunaga and Hostetler 1975 improved on the density estimation approach by estimating the generalization error directly, rather than using empirical test error (hence reducing variance.)

Nevertheless, the convergence rates of nonparametric methods of Bayes risk estimation are too slow to provide much assurance for their use in high-dimensional problems. On the other hand, the empirical success of supervised learning approaches seems to justify a certain optimism that at least one of the popular black-box methods (random forests, neural networks, kernel SVM) can come close to the Bayes error rate within realistic sample sizes. Methods such as random forests and neural networks are known to have *universal approximation* properties that guarantee consistent recovery of the classification rule, given that the model complexity is scaled at an appropriate rate as the sample size increases; but methods such as kernel SVM have much more restrictive function classes and therefore have no guarantee of being able to recover the Bayes error. Yet, it is seen in many learning tasks that the test error is very similar for very different classification methods, some universal and some non-universal, suggesting that the true signal lies in a low-complexity function class which can be adequately approximated using a variety of methods. Empirical evidence for these phenomenon are collected in studies on the learning curves of classifers, beginning with Cortes 1994, and applied to specific applications by Figueroa et al. 2012, Beleites et al. 2013. We leave further discussion of Bayes error estimation for future work; in the theoretical portion of the current paper, we will simply leave unspecified the method used to estimate Bayes error, and in our simulation and real data examples, we will use an ad hoc approach for estimating Bayes error from test error for the purpose of demonstrating our method.

8 Conclusions

- We derive a relationship between average Bayes error (ABE) and mutual information (MI), motivating a novel estimator $\hat{I}_{LS}$.
- Theory based on high dimensional, low SNR limit, where

$$\text{ABE} \leftrightarrow \text{MI}.$$

- In ideal settings for supervised learning, ABE can be estimated effectively and $\hat{I}_{LS}$ can recover MI at much lower sample sizes than nonparametric methods.
- In simulations, $\hat{I}_{LS}$ works better than Fano's inequality or the confusion matrix approach.

9 References

- Gastpar, M. Gill, P. Huth, A. Theunissen, F. "Anthropic Correction of Information Estimates and Its Application to Neural Coding." *IEEE Trans. Info. Theory*, Vol 56 No 2, 2010.
- A. Borst and F. E. Theunissen, "Information theory and neural coding" Nature Neurosci., vol. 2, pp. 947?957, Nov. 1999.
- L. Paninski, "Estimation of entropy and mutual information," Neural Comput., vol. 15, no. 6, pp. 1191?1253, 2003.
- I. Nelken, G. Chechik, T. D. Mrsic-Flogel, A. J. King, and J. W. H. Schnupp, "Encoding stimulus information by spike numbers and mean response time in primary auditory cortex," J. Comput. Neurosci., vol. 19, pp. 199?221, 2005.
- Cover and Thomas. Elements of information theory.
- Muirhead. Aspects of multivariate statistical theory.
- van der Vaart. Asymptotic statistics.