How many faces can be recognized? Performance extrapolation for multi-class classification

Charles Y. Zheng
Department of Statistics
Stanford University
Stanford, CA 94305
snarles@stanford.edu

Rakesh Achanta
Department of Statistics
Stanford University
Stanford, CA 94305
rakesha@stanford.edu

Yuval Benjamini
Department of Statistics
Hebrew University
Jerusalem, Israel
yuval.benjamini@mail.huji.ac.il

Abstract

The difficulty of multi-class classification generally increases with the number of classes. *Recognition systems* employ such classifiers in order to recognize people, spoken words, or chemicals: it is often of interest to know how many species the system can be trained to recognize before dropping below a minimum accuracy threshold. However, before such systems are deployed, typically only a small number of species are available for testing the system. Can we predict how well the recognition system will scale with an increased number of classes? We distinguish between two types of multi-class classifiers: *separable* classifiers, which include k-nearest neighbors, multinomial regression, one-vs-one and one-vs-all classifiers, *non-separable* classifiers, such as deep neural networks and decision trees. For recognition systems based on separable classifiers, the problem of predicting scalability reduces to the problem of estimating the higher-order moments of a *conditional accuracy distribution*, which in turn can be estimated from data.

1 Introduction

Object recognition, face recognition (or more generally person recognition) and language are a few of the cognitive building blocks which are fundamental to human cognition, and which can be understood as examples of generalized classification tasks. Machine classification can be employed to mimic this power of recognition. A robot equipped with a camera can algorithmically segment its input image into objects, and to learn to recognize unique objects and people which regularly appear in its environment. A general approach to implement such a recognition ability starts by employing some parametric featurization of the object to be identified. For example, for the task of face recognition, one might define features such as the proportions between the eyes and the relative position and size of the nose. The full domain of the recognition task is a collection of *measurements* (e.g. photographs of faces) which are associated to different *species*. The recognition system can be implemented by training a multi-class classifier to assign measurements to their corresponding species. While the system is in deployment, new species may be added to the system: when this happens, the classifier is retrained (or updated) using training data from the species to be added.

A limitation to such recognition systems, whether they be natural or artificial, is that the performance of the system (in terms of correct classification) can degrade if there are too many species. A face

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

recognition algorithm can have very high success rate if it only needs to distinguish between 100 different faces, but its identifications may be less reliable when it needs to distinguish between 10000 different faces. The consequences of such errors may be severe: Cole (2005) lists 22 cases of fingerprint misindentification in criminal trials.

Therefore, in the case of engineering recognition systems, it is of much practical interest to be able to evaluate the reliability of such systems before they are deployed. Yet, during the development phase, typically data from only a fraction of the species in the domain are available. From the empirical performance of the classifier on this initial subset of species, can we predict the performance of the system on a larger subset of species (or the entire domain)?

A related problem arises in studying *naturally occurring* recognition systems: for instance, human memory. Neuroscientists may be directly interested in the number of different cues which can be recalled by a subject. A similar dilemma arises, where data from only small number of species can be obtained, due to experimental constraints. From this data, can we predict the number of species which can be distinguished by the recognition system, above a minimum accuracy threshold?

We address this problem of *performance extrapolation* under the assumption that the initial subset of species is an i.i.d. sample from a larger population of species. Section 2 formalizes the recognition problem, and section 3 formalizes the problem of performance extrapolation. For a restricted family of classifiers–*separable* classifiers, we show that the problem of performance extrapolation reduces to a problem of nonparametric moment estimation. In section 4, we explore methods for nonparametric estimation. We propose a method called *moment-constrained maximum pseudolikelihood*, and demonstrate that it has reasonable performance in simulated and real data examples in section 5; section 6 concludes.

2 The recognition problem

In the recognition problem, there exists a large or infinite population of *species*, e.g. individuals, and an infinite population of *measurements*, and each measurement is associated with a single species. A recognition system is an algorithm which contains a database of measurements from a number of species: each measurement is represented within the recognition system by a real vector $y \in \mathbb{R}^p$ which lies in the space $\mathcal{Y}$, and labeled with an integer-valued ID key. The ID key is the recognition system's internal representation of the species associated to the measurements: we assume that the database has the property of *integrity*, where any two measurements in the database share the same ID key if and only if they are associated with the same species. Given a query measurement, the system answers with an ID number: this answer is correct if and only if the query measurement and the database measurements labeled with the same ID are associated to the same species. Without loss of generality, take the ID keys to be $\{1, 2, \ldots, k\}$, and let $y^{(i),j}$ denote the jth measurement associated with ID key i in the database; let n_i be the number of measurements associated to the ith ID key, and $n = \sum_{i=1}^{k} n_i$ be the total number of measurements.

The recognition system maps queries y to an ID key; let us refer to this mapping as $f : \mathcal{Y} \to \{1, \ldots, k\}$. New data can be added to the recognition system over time; we assume it is done in a manner so that the integrity of the database is preserved. Let $t = 1, 2, \ldots$ refer to discrete time stages, and let k_t denote the number of classes in the database at time t, and f_t denote the mapping from queries to answers at time t; also define $n_{i,t}$ analogously.

2.1 Multi-class classification

A natural approach for implementing a recognition system is through *supervised learning*: the data is input into a supervised learning algorithm in order to *learn* a classification rule $f(y)$. Examples of multi-class classifiers include k-nearest neighbors, multinomial logistic regression, linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), decision trees, and random forests, as well as the two 'divide and conquer' approaches, one-vs-one (OVO) and one-vs-all (OVA) (Friedman et al, 2008.) A unified definition of a classifier is as follows: a k-class classifier $\mathcal{C}$ is a function which takes $k + 2$ arguments: the first k arguments are *probability distributions* $F_1, \ldots, F_k$, the $(k + 1)$st argument is a vector of prior probabilities $\vec{\pi}$, and the $(k + 2)$nd argument is a *query* y [1]. The level

[1] We treat randomized classifiers (such as random forests) as a probability distribution over deterministic classifiers.

sets $\{y : \mathcal{C}(F_1, \ldots, F_k, y) = k\}$ are called *decision regions.* We say the classifier is *continuous* if and only if the *decision regions* of $\mathcal{C}$ are continuous in the first k arguments with respect to the topology of weak convergence[2]. Most of the commonly used classifiers satisfy this definition of continuity: an exception is k-nearest neighbors with fixed k, but then again, k-nearest neighbors with $\lim_{n\to\infty} k/n \in (0, 1)$ is continuous. *Training a classifier* refers to defining a classification rule

$$f(y) = \mathcal{C}(\hat{F}_1, \ldots, \hat{F}_k, \vec{\pi}, y)$$

where $\hat{F}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \delta_{y^{(i),j}}$ is the empirical distribution, and where $\vec{\pi}$ is commonly set to either be the uniform distribution on k elements, or the empirical proportions of the classes, but can be adjusted in order to favor certain classes over others.

For theoretical purposes, let us assume that each species can be *uniquely* parameterized by a parameter vector x, and let $\mathcal{X}$ denote the space of parameter vectors. Note that the parameter x may or may not observed by the recognition system: in the latter case, the system only has access to ID keys which are non-informative of $p(y|x)$. Let $p(x)$ be the population distribution of species, and assume that a conditional feature vector distribution $p(y|x)$ be defined for every $x \in \mathcal{X}$. We require that $p(x)$ be a density with respect to Lesbegue measure, but $p(y|x)$ is allowed to include Dirac delta components. The database of the classifier therefore consists of pairs $\{(x^{(i)}), y^{(i),j})\}$ where $x^{(i)}$ is the parameter of the ith species in the database.

Whenever new data is added, the classifier is retrained. Depending on which multi-class classifier is used, we can categorize the recognition system as either a *separable* system, or a *non-separable* system.

2.2 Separability

The property of separability captures the intuitive notion that *information is not shared between classes*. To formalize the notion of separability, we begin with the class of *scoring rule*-based classifier. A scoring rule $\mathcal{Q}$ is a real-valued function which takes three arguments: a distribution F on $\mathcal{Y}$, a prior probability π, and a query measurement $y \in \mathcal{Y}$. A scoring-based classifier defines

$$\mathcal{C}(F_1, \ldots, F_k, \vec{\pi}, y) = \text{argmax}_i \mathcal{Q}(F_i, \vec{\pi}_i, y).$$

For notational convenience, we assume that ties occur with probability zero: that is, $p(x, y)$ and $\mathcal{Q}$ jointly satisfy the *tie-breaking* property: for any x and $\pi \in [0, 1]$, letting $F_n(x)$ be the empirical distribution of n points drawn from $p(y|x)$, and $Y, Y' \overset{iid}{\sim} p(y|x)$, we have

$$\Pr[\mathcal{Q}(F_n(x), \pi, Y) = \mathcal{Q}(F_n(x), \pi, Y')] = 0. \tag{1}$$

Quadratic discriminant analysis and Naive Bayes are two examples of scoring-based classifiers. For QDA, the scoring rule is given by

$$\mathcal{Q}_{QDA}(F, \pi, y) = -(y - \mu(F))^T \Sigma(F)^{-1} (y - \mu(F)) - \log\det(\Sigma(F)) + 2\log\pi$$

where $\mu(F) = \int y dF(y)$ and $\Sigma(F) = \int (y - \mu(F))(y - \mu(F))^T dF(y)$. In Naive Bayes, the scoring rule is

$$\mathcal{Q}_{NB}(\hat{F}, \pi, y) = \log\pi + \sum_{i=1}^{n} \log \hat{f}_i(y_i)$$

where $\hat{f}_i$ is a density estimate for the ith component of F.

A *separable* classifier is a classifier which can be *approximated* by a scoring-based classifier in a large-sample limit. Let $\hat{F}_{i,t}$ denote the empirical distribution of the measurements of species i at time t, and $\hat{F}_t$ denote the empirical distribution of all measurements at time t.

Definition 2.1. (Separability) Suppose that for time stages $t = 1, 2, \ldots$, $k_t = O(t)$ and $n_{i,t} = O(t)$, with $x^{(1)}, x^{(2)}, \ldots$ drawn i.i.d. from $p(x)$, and $y^{(i),1}, \ldots$ drawn i.i.d. from $p(y|x^{(i)})$ for each $i = 1, 2, \ldots$. A recognition system (characterized by mappings f_t) is considered *separable* if and only there exists a scoring rule $\mathcal{Q}$ and probability vector $\tilde{\pi}$ such that defining

$$\tilde{f}_t(y) = \text{argmax}_{i=1}^{k_t} \mathcal{Q}(\hat{F}_{i,t}, \tilde{\pi}_i, y)$$

[2]We say that a randomized classifier is continuous if and only if it is continuous with probability one.

we have

$$\lim_{t\to\infty} \frac{1}{k_t} \sum_{i=1}^{k_t} \Pr[f_t(Y) = \tilde{f}_t(Y) | Y \sim p(y|x^{(i)})] \to 1.$$

We will show that recognition systems based on certain implementations of k-nearest neighbors, LDA, one-vs-one, or one-vs-all classifiers satisfy this definition of separability.

Definition 2.2.(i) Define a binary classifier $\mathcal{B}$ as a binary-valued mapping with four arguments: distributions F_0, F_1, prior probability π_0 and a query y. A one-vs-one (OVO) recognition system is defined by

$$f_t(y) = \text{argmax}_{i=1}^{k_t} \sum_{j\neq i} I\left(\mathcal{B}(\hat{F}_{i,t}, \hat{F}_{j,t}, \frac{n_{i,t}}{n_{i,t}+n_{j,t}}, y) = 0\right),$$

resolving ties arbitrarily.

(ii) Define a binary scoring rule $\mathcal{D}$ as a real-valued mapping with four arguments: distributions F_0, F_1, prior probability π_0, and a query y. A one-vs-all (OVA) recognition system is defined by

$$f_t(y) = \text{argmax}_{i=1}^{k_t} \mathcal{D}\left(\hat{F}_{i,t}, \sum_{j\neq i} \frac{n_{j,t}}{n_t - n_{i,t}} \hat{F}_{j,t}, \frac{n_{i,t}}{n_t}, y\right).$$

(iii) Let d be a distance metric on $\mathcal{Y}$. Let $D_t(y)$ denote the induced distribution of $d(Y, y)$ when $Y \sim \hat{F}_t$, and let $d_{\alpha,t}$ denote the α-quantile of $D_t(y)$. A kNN recognition system with neighborhood size α is defined by

$$f_t(y) = \text{argmax}_{i=1}^{k_t} \Pr[d(y, Y) < d_{\alpha,t} | Y \sim \hat{F}_{i,t}].$$

(iv) Assume WLOG that $y_1 = 1$ for all $y \in \mathcal{Y}$, and let B^t be a $p \times k_t$ matrix which minimizes the log-likelihood

$$\sum_{j=1}^{k_t} n_{j,t} \mathbf{E}_{\hat{F}_j}\left[\langle Y, B_j^t\rangle - \log\left[\sum_{\ell=1}^{k_t} \exp[\langle Y, B_\ell^t\rangle]\right]\right].$$

A multinomial logistic regression recognition system is defined by

$$f_t(y) = \text{argmax}_{i=1}^{k_t} \langle y, B_i^t\rangle.$$

Theorem 2.1. *(i) an OVO recognition system equipped with a continuous binary classifier is separable; (ii) an OVA recognition system equipped with a continuous binary scoring rule is separable; (iii) a kNN recognition system with fixed neighborhood size $\alpha \in (0, 1)$ is separable; (iv) a multinomial logistic regression recognition system is separable.*

3 Prediction Extrapolation

3.1 Problem formulation

Recall the notation from section 2.1. Assume that $x^{(1)}, x^{(2)}, \ldots$ are sampled i.i.d. from the population distribution $p(x)$, and $y^{(i),1}, \ldots, y^{(i),r}$ from the conditional distribution $p(y|x^{(i)})$ for $i = 1, 2, \ldots$. Take $k_t = t$ and $n_{i,t} = r$.

Unlike in section 2.1., only the first $r_1 < r$ measurements in each species will be used to construct the classifier: redefine

$$\hat{F}_{i,t} = \frac{1}{r_1} \sum_{j=1}^{r_1} \delta_{y^{(i),j}}.$$

Since $\hat{F}_{i,t}$ no longer depends on t, we will write it as $\hat{F}_i$. The remaining $r_2 = r - r_1$ measurements of each species constitute the *test set*, used to evaluate the performance of the classifier.

The generalization accuracy at time t is defined

$$\text{acc}^{(t)} = \frac{1}{k} \sum_{i=1}^{k} \Pr[f_t(y) = i | y \sim p(y|x^{(i)})].$$

The extrapolation problem is the problem of predicting $\text{acc}^{(K)}$ using only information known at time $k < K$, namely, $\{y^{(i),j}\}_{i=1,j=1}^{k,r}$.

3.2 Conditional accuracy

Consider estimating the expected accuracy at time t,

$$p_t \stackrel{def}{=} \mathbf{E}[\text{acc}^{(t)}].$$

Assume that the classifier is based on a scoring rule $\mathcal{Q}$. Further assume that $\mathcal{Q}$ has a trivial dependence on the prior probability parameter: $\mathcal{Q}(F, a, y) = \mathcal{Q}(F, b, y)$ for all F, y, and $a, b \in [0, 1]$. This assumption is more mild than it appears, since most classifiers indeed have a trivial dependence on $\vec{\pi}$ in the case when $\vec{\pi}$ is set to the uniform distribution.

Define the *conditional accuracy* function $u(F, y)$ which maps a distribution F on $\mathcal{Y}$ and a *test* observation y to a real number in $[0, 1]$. The conditional accuracy gives the probability that for independently drawn X, letting $\hat{F}(X)$ be the empirical distribution of r_1 measurements drawn from $p(y|X)$, that the scoring function $\mathcal{Q}(F, 0, y)$ will give a higher score to y than the scoring function $\mathcal{Q}(\hat{F}(X), 0, y)$:

$$u(F, y) = \Pr[\mathcal{Q}(F, 0, y) > \mathcal{Q}(\hat{F}(X), 0, y)].$$

Define the *conditional accuracy* distribution μ as the law of $u(\hat{F}(X), Y)$ when $X \sim p(x)$, $Y \sim p(y|X)$, and $\hat{F}(X)$ is drawn as specified above. The significance of the conditional accuracy distribution is that the expected generalization error p_t can be written in terms of its moments.

Theorem 3.1. *Let U be defined as the random variable*

$$U = u(F, Y)$$

for X, Y drawn from $p(x, y) = p(x)p(y|x)$, and $\hat{F}(X) = \frac{1}{r_1}\sum_{j=1}^{r_1} \delta Y^j$ with $Y^i \overset{iid}{\sim} p(y|X)$ Then $p_k = \boldsymbol{E}[U^{k-1}]$.

Proof. Write $q^{(i)}(y) = \mathcal{Q}(\hat{F}_i, 0, y)$, and let $Y^{(i),*} \sim p(y|X^{(i)}$ for $i = 1, \ldots, k$. Note that by using conditioning and conditional independence, p_k can be written

$$\begin{aligned}
p_k &= \mathbf{E}\left[\frac{1}{k}\sum_{i=1}^{k} \Pr[q^{(i)}(Y^{(i),*}) > \max_{j\neq i} q^{(j)}(Y^{(i),*})]\right] \\
&= \mathbf{E}\left[\Pr[q^{(1)}(Y^{(1),*}) > \max_{j\neq 1} q^{(j)}(Y^{(1),*})]\right] \\
&= \mathbf{E}[\Pr[q^{(1)}(Y^{(1),*}) > \max_{j\neq 1} q^{(j)}(Y^{(1),*})|Y^{(1),*}, \hat{F}_1]] \\
&= \mathbf{E}[\Pr[\cap_{j>1} q^{(1)}(Y^{(1),*}) > q^{(j)}(Y^{(1),*})|Y^{(1),*}, \hat{F}_1]] \\
&= \mathbf{E}[\prod_{j>1} \Pr[q^{(1)}(Y^{(1),*}) > q^{(j)}(Y^{(1),*})|Y^{(1),*}, \hat{F}_1]] \\
&= \mathbf{E}[\Pr[q^{(1)}(Y^{(1),*}) > q^{(2)}(Y^{(1),*})|Y^{(1),*}, \hat{F}_1]^{k-1}] \\
&= \mathbf{E}[u(\hat{F}_1, Y^{(1),*})^{k-1}] = \mathbf{E}[U^{k-1}].
\end{aligned}$$

□

Theorem 3.1 tells us that the problem of extrapolation can be approached by attempting to estimate the conditional accuracy distribution. The $(t-1)$th moment of U gives us p_t, which will in turn be a good estimate of $\text{acc}^{(t)}$.

3.3 Properties of the conditional accuracy distribution

The conditional error distribution ν is determined by $p(x, y)$ and $\mathcal{Q}$. What can we say about the the conditional accuracy distribution without making any assumptions on either $p(x, y)$ or $\mathcal{Q}$? The

answer is: not much–for an arbitrary probability measure ν' on $[0,1]$, one can construct $p(x,y)$ and $\mathcal{Q}$ such that $\nu = \nu'$.

Theorem 3.2. *Let U be defined as in Theorem 2.1, and let ν denote the law of U. Then, for any probability distribution ν' on $[0,1]$, one can construct a joint distribution $p(x,y)$ and a scoring rule $\mathcal{Q}$ such that $\nu = \nu'$.*

In practice, however, the scoring rule $\mathcal{Q}$ must approximate a monotonic function of the conditional density $p(y|x)$ in order to yield an effective classifier. It is therefore notable that in the case that (X,Y) have a density with respect to Lesbegue measure, taking an *optimal* scoring rule, with the property that $\mathcal{Q}(x,y,S(x)) = g(p(y|x))$ for monotonic g, the distribution of U has a monotonically increasing density.

Theorem 3.3. *Let U be defined as in Theorem 3.1, and let ν denote the law of U. Suppose (X,Y) has a density $p(x,y)$ with respect to Lebesgue measure on $\mathcal{X} \times \mathcal{Y}$, and with probability one, $\mathcal{Q}(y,0,\hat{F}(X))$ satisfies the property of monotonicity*

$$p(y|x) > p(y'|x) \text{ implies } \mathcal{Q}(\hat{F}(X),0,y) > \mathcal{Q}(\hat{F}(X),0,y')$$

and the property of tie-breaking (1)*, then μ has a density $\eta(u)$ on $[0,1]$ which is monotonic in u.*

4 Nonparametric Estimation

Let us assume that U has a density $\eta(u)$. While $U = u(\hat{F}(X),Y)$ cannot be directly observed, we can estimate $u(\hat{F}_i, y^{(i),r_1+j})$ for any $1 \le i \le k$, $1 \le j \le r_2$ from the data.

Theorem 4.1. *For given $p(x,y)$ and scoring rule $\mathcal{Q}$, assume that U as defined in Theorem 3.1 has a density $\eta(u)$ and that $\mathcal{Q}$ satisfies the tie-breaking property* (1)*. Define*

$$V_{i,j} = \sum_{i=1}^{k} I(q^{(i)}(y^{(i),j}) > q^{(j)}(y^{(i),j})).$$

Then

$$V_{i,j} \sim \textit{Binomial}(k, u(\hat{F}_i, y^{(i),j})).$$

At a high level, we have a hierarchical model where U is drawn from a density $\eta(u)$ on $[0,1]$ and then $V_{i,j} \sim \text{Binomial}(k,U)$; therefore the marginal distribution of $V_{i,j}$ can be written

$$\Pr[V_{i,j} = \ell] = \binom{k}{\ell} \int_0^1 u^\ell (1-u)^{k-\ell} \eta(u) du.$$

However, the observed $\{V_{i,j}\}$ do *not* comprise an i.i.d. sample.

We discuss the following three approaches for estimating $p_t = \mathbf{E}[U^{t-1}]$ based on $V_{i,j}$. The first is *unbiased estimation* based on binomial U-statistics, which is discussed in Section 4.1. The second is the *psuedolikelihood* approach. In problems where the marginal distributions are known, but the dependence structure between variables is unknown, the *psuedolikelihood* is defined as the product of the marginal distributions. For certain problems in time series analysis and spatial statistics, the maximum psuedolikelihood estimator (MPLE) is proved to be consistent (CITE). We discuss psuedolikelihood-based approaches in Sections 4.2 and 4.3. A third approach is an adaptation of the *mutual information estimator* developed by (Anon 2016). Anon 2016 develop an asymptotic theory which relates the Bayes error to the mutual information $I(X;Y)$ and vice versa. This allows us to estimate "information" from classification error p_k, and then predict the generalization error p_t from the estimated information: details are given in Section 4.4.

4.1 Unbiased estimation

If $V \sim \text{Binomial}(k,\eta)$, then an unbiased estimator $f_t(V)$ of $\eta^(t-1)$ exists if and only if $0 \le t \le k$.

The theory of U-statistics provides the minimal variance unbiased estimator for $\eta^(t-1)$:

$$\eta^t = \mathbf{E}\left[\frac{\binom{V}{t}}{\binom{k}{t}}\right].$$

This result can be immediately applied to yield an unbiased estimator of p_t, when $t \leq k$:

$$\hat{p}_t^{UN} = \mathbf{E}\left[\frac{1}{kr_2}\sum_{i=1}^{k}\sum_{j=1}^{r_2}\frac{\binom{V_{i,j}}{t}}{\binom{k}{t}}\right]. \tag{2}$$

The problem of *extrapolation* concerns the case $t > k$, in which the expression (2) is undefined. Still, the estimator (2) is worthy of study, since it has close to optimal performance for the case $t \leq k$.

4.2 Maximum pseudo-likelihood

The psuedolikelihood is defined as

$$\ell_t(\eta) = \sum_{i=1}^{k}\sum_{j=1}^{r_1}\log\left(\int u^{V_{i,j}}(1-u)^{k-V_{i,j}}\eta(u)du\right), \tag{3}$$

and a maximum psuedolikelihood estimator (MPLE) is defined as any density $\hat{\eta}$ such that

$$\ell(\hat{\eta}_{MPLE}) = \sup_{\eta}\ell_t(\eta).$$

The motivation for $\hat{\eta}_{MPLE}$ is that it consistently estimates η in the limit where $k \to \infty$.

Theorem 4.2. *For given $p(x,y)$ and scoring rule $\mathcal{Q}$, assume that U as defined in Theorem 3.1 has a density $\eta(u)$ and that $\mathcal{Q}$ satisfies the tie-breaking property* (1)*, and also that $r_2 \geq 1$. For $t = 1, 2, \ldots,$ let $\hat{\eta}_t$ be any MPLE for ℓ_t. As $k_t \to \infty$, $\hat{\eta}_t$ weakly converges to η.*

However, in finite samples, $\hat{\eta}_{MPLE}$ is not uniquely defined, and if we define the plug-in estimator

$$\hat{p}_t^{MPLE} = \int u^{t-1}\hat{\eta}_{MPLE}(u)du,$$

$\hat{p}_t^{MPLE}$ can vary over a large range, depending on which $\hat{\eta} \in \text{argmax}_\eta\ell_t(\eta)$ is selected. These shortcomings motivate the adoption of additional constraints on the estimator $\hat{\eta}$.

4.3 Constrained pseudo-likelihood

Theorem 3.2. motivates the *monotonicity constraint* that $\frac{d\hat{\eta}}{du} > 0$, hence we define $\hat{\eta}_{INC}$ as a solution to

$$\text{maximize } \ell_t(\eta) \text{ subject to } \frac{d\hat{\eta}}{du} > 0.$$

An alternative strategy is to directly attack the variability is $\hat{p}_t$ due to non-uniqueness of $\hat{\eta}$. Therefore, we define $\hat{\eta}_{MC}$ (where MC stands for moment-constrained) as

$$\text{maximize } \ell_t(\eta) \text{ subject to } \int u^{k-1}\eta(u)du = \hat{p}_k^{UN}.$$

Thirdly, we can combine both the moment constraint and the monotonicity constraint, yielding $\hat{\eta}_{COM}$, which is obtained by solving

$$\text{maximize } \ell_t(\eta) \text{ subject to } \int u^{k-1}\eta(u)du = \hat{p}_k^{UN} \text{ and } \frac{d\hat{\eta}}{du} > 0.$$

Unfortunately, none of the three density estimators are uniquely defined. An easy way to see this is to transform the parameterization of $\eta(u)$, defining

$$\eta(u) = \int_0^u \xi(u)du;$$

the monotonicity constraint is equivalent to the condition that $\xi > 0$, and the moment condition translates into a linear equality constraint on ξ.

5 Results

6 Discussion

Acknowledgments

CZ is supported by an NSF graduate research fellowship.

References

[X] Naselaris, T., Kay, K. N., Nishimoto, S., & Gallant, J. L. (2011). Encoding and decoding in fMRI. *Neuroimage*, 56(2), 400-410.

[X] Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning.* Vol. 1. Springer, Berlin: Springer series in statistics, 2008.