

---

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

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

1       The difficulty of multi-class classification generally increases with the number of  
2       classes. Using data from a subset of the classes, can we predict how well a classifier  
3       will scale with an increased number of classes? Under the assumption that the  
4       classes are sampled exchangeably, and under the assumption that the classifier is  
5       generative (e.g. QDA or Naive Bayes), we show that the expected accuracy when  
6       the classifier is trained on  $k$  classes is the  $k - 1$ st moment of a *conditional accuracy*  
7       *distribution*, which can be estimated from data. This provides the theoretical  
8       foundation for performance extrapolation based on pseudolikelihood, unbiased  
9       estimation, and high-dimensional asymptotics. We investigate the robustness of  
10      our methods to non-generative classifiers in simulations and one optical character  
11      recognition example.

## 12   1   Introduction

13   In multi-class classification, one observes pairs  $(z, y)$  where  $y \in \mathcal{Y} \subset \mathbb{R}^p$  are feature vectors, and  $z$   
14   are unknown labels, which lie in a countable label set  $\mathcal{Z}$ . The goal is to construct a classification rule  
15   for predicting the label of a new data point; generally, the classification rule  $h : \mathcal{Y} \rightarrow \mathcal{Z}$  is learned  
16   from previously observed data points. In many applications of multi-class classification, such as face  
17   recognition or image recognition, the space of potential labels is practically infinite. In such a setting,  
18   one might consider a sequence of classification problems on finite label subsets  $\mathcal{Z}_1 \subset \dots \subset \mathcal{Z}_K$ ,  
19   where in the  $i$ -th problem, one constructs the classification rule  $h^{(i)} : \mathcal{Y} \rightarrow \mathcal{Z}_i$ . Supposing that  $(Z, Y)$   
20   have a joint distribution, define the accuracy for the  $i$ -th problem as

$$\text{acc}^{(i)} = \Pr[h^{(i)}(Y) = Z | Z \in \mathcal{Z}_i].$$

21   Using data from only  $\mathcal{Z}_k$ , can one predict the accuracy achieved on the larger label set  $\mathcal{Z}_K$ , with  
22    $K > k$ ? This is the problem of *performance extrapolation*.

23   A practical instance of performance extrapolation occurs in neuroimaging studies, where the number  
24   of classes  $k$  is limited by experimental considerations. Kay et al. [1] obtained fMRI brain scans which  
25   record how a single subject's visual cortex responds to natural images. The label set  $\mathcal{Z}$  corresponds  
26   to the space of all grayscale photographs of natural images, and the set  $\mathcal{Z}_1$  is a subset of 1750  
27   photographs used in the experiment. They construct a classifier which achieves over 0.75 accuracy  
28   for classifying the 1750 photographs; based on exponential extrapolation, they estimate that it would  
29   take on the order of  $10^{9.5}$  photographs before the accuracy of the model drops below 0.10! Directly  
30   validating this estimate would take immense resources, so it would be useful to develop the theory  
31   needed to understand how to compute such extrapolations in a principled way.

32   However, in the fully general setting, it is impossible to construct non-trivial bounds on the accuracy  
33   achieved on the new classes  $\mathcal{Z}_K \setminus \mathcal{Z}_k$  based only on knowledge of  $\mathcal{Z}_k$ : after all,  $\mathcal{Z}_k$  could consist

entirely of well-separated classes while the new classes  $\mathcal{Z}_K \setminus \mathcal{Z}_k$  consist entirely of highly inseparable classes, or vice-versa. Thus, the most important assumption for our theory is that of *exchangeable sampling*. The labels in  $\mathcal{Z}_i$  are assumed to be an exchangeable sample from  $\mathcal{Z}$ . The condition of exchangeability ensures that the separability of random subsets of  $\mathcal{Z}$  can be inferred by looking at the empirical distributions in  $\mathcal{Z}_k$ , and therefore that some estimate of the achievable accuracy on  $\mathcal{Z}_K$  can be obtained.

The assumption of exchangeability greatly limits the scope of application for our methods. Many multi-class classification problems have a hierarchical structure [2], or have class labels distributed according to non-uniform discrete distributions, e.g. power laws [3]; in either case, exchangeability is violated. It would be interesting to extend our theory to the hierarchical setting, or to handle non-hierarchical settings with non-uniform prior class probabilities, but again we leave the subject for future work.

In addition to the assumption of exchangeability, we consider a restricted set of classifiers. We focus on *generative classifiers*, which are classifiers that work by training a model separately on each class. This convenient property allows us to characterize the accuracy of the classifier by selectively conditioning on one class at a time. In section 3, we use this technique to reveal an equivalence between the expected accuracies of  $\mathcal{Z}_k$  to moments of a common distribution. This moment equivalence result allows standard approaches in statistics, such as U-statistics and nonparametric pseudolikelihood, to be directly applied to the extrapolation problem, as we discuss in section 4. In non-generative classifiers, the classification rule has a joint dependence on the entire set of classes, and cannot be analyzed by conditioning on individual classes. In section 5, we empirically study the performance of our classifiers. Since generative classifiers only comprise a minority of the classifiers used in practice, we applied our methods to a variety of generative and non-generative classifiers in simulations and in one OCR dataset. Our methods have varying success on generative and non-generative classifiers, but seem to work badly for neural networks.

*Our contribution.*

To our knowledge, we are the first to formalize the problem of prediction extrapolation. We introduce three methods for prediction extrapolation: the method of extended unbiased estimation and the constrained pseudolikelihood method are novel. The third method, based on asymptotics, is a new application of a recently proposed method for estimating mutual information [4].

## 2 Setting

Having motivated the problem of performance extrapolation, we now reformulate the problem for notational and theoretical convenience. Instead of requiring  $\mathcal{Z}_k$  to be a random subset of  $\mathcal{Z}$  as we did in section 1, take  $\mathcal{Z} = \mathbb{N}$  and  $\mathcal{Z}_k = \{1, \dots, k\}$ . We fix the size of  $\mathcal{Z}_k$  without losing generality, since any monotonic sequence of finite subsets can be embedded in a sequence with  $|\mathcal{Z}_k| = k$ . In addition, rather than randomizing the labels, we will randomize the marginal distribution  $p(y|z)$  of each label; Towards that end, let  $\mathcal{Y} \subset \mathbb{R}^p$  be a space of feature vectors, and let  $\mathcal{P}(\mathcal{Y})$  be a measurable space of probability distributions on  $\mathcal{Y}$ . Let  $\mathbb{F}$  be a probability measure on  $\mathcal{P}$ , and let  $F_1, F_2, \dots$  be an infinite sequence of i.i.d. draws from  $\mathbb{F}$ . We refer to  $\mathbb{F}$ , a probability measure on probability measures, as a *meta-distribution*. The distributions  $F_1, \dots, F_k$  are the marginal distributions of the first  $k$  classes. Further assuming that the labels are equiprobable, we rewrite the accuracy as

$$\text{acc}^{(t)} = \frac{1}{t} \sum_{i=1}^t \mathbb{P}_{F_i}[h^{(t)}(Y) = i].$$

where the probabilities are taken over  $Y \sim F_i$ .

In order to construct the classification rule  $h^{(t)}$ , we need data from the classes  $F_1, \dots, F_t$ . In most instances of multi-class classification, one observes independent observations from each  $F_i$  which are used to construct the classifier. Since the order of the observations does not generally matter, a sufficient statistic for the training data for the  $t$ -th classification problem is the collection of empirical distributions  $\hat{F}_1^{(t)}, \dots, \hat{F}_t^{(t)}$  for each class. Henceforth, we make the simplifying assumption that the training data for the  $i$ -th class remains fixed from  $t = i, i + 1, \dots$ , so we drop the superscript on  $\hat{F}_i^{(t)}$ . Write  $\hat{\mathbb{F}}(F)$  for the conditional distribution of  $\hat{F}_i$  given  $F_i = F$ ; also write  $\hat{\mathbb{F}}$  for the marginal distribution of  $\hat{F}$  when  $F \sim \mathbb{F}$ . As an example, suppose every class has the number of training

84 examples  $r \in \mathbb{N}$ ; then  $\hat{F}$  is the empirical distribution of  $r$  i.i.d. observations from  $F$ , and  $\hat{\mathbb{F}}(F)$  is  
 85 the *empirical meta-distribution* of  $\hat{F}$ . Meanwhile,  $\hat{\mathbb{F}}$  is the true meta-distribution of the empirical  
 86 distribution of  $r$  i.i.d. draws from a random  $F \sim \mathbb{F}$ .

## 87 2.1 Multiclass classification

88 Extending the formalism of Tewari and Bartlett [5]<sup>1</sup>, we define a classifier as a collection of mappings  
 89  $\mathcal{M}_i : \mathcal{P}(\mathcal{Y})^k \times \mathcal{Y} \rightarrow \mathbb{R}$  called *classification functions*. Intuitively speaking, each classification  
 90 function *learns a model* from the first  $k$  arguments, which are the empirical marginals of the  $k$  classes,  
 91  $\hat{F}_1, \dots, \hat{F}_k$ . For each class, the classifier assigns a real-valued *classification score* to the *query point*  
 92  $y \in \mathcal{Y}$ . A higher score  $\mathcal{M}_i(\hat{F}_1, \dots, \hat{F}_k, y)$  indicates a higher estimated probability that  $y$  belongs to  
 93 the  $k$ -th class. Therefore, the classification rule corresponding to a classifier  $\mathcal{M}_i$  assigns a class with  
 94 maximum classification score to  $y$ :

$$h(y) = \operatorname{argmax}_{i \in \{1, \dots, k\}} \mathcal{M}_i(y).$$

95 For some classifiers, the classification functions  $\mathcal{M}_i$  are especially simple in that  $\mathcal{M}_i$  is only a  
 96 function of  $\hat{F}_i$  and  $y$ . Furthermore, due to symmetry, in such cases one can write

$$\mathcal{M}_i(\hat{F}_1, \dots, \hat{F}_k, y) = \mathcal{Q}(\hat{F}_i, y),$$

97 where  $\mathcal{Q}$  is called a *single-class classification function* (or simply *classification function*), and we  
 98 say that  $\mathcal{M}$  is a *generative classifier*. Quadratic discriminant analysis and Naive Bayes [6] are two  
 99 examples of generative classifiers<sup>2</sup>. The *generative* property allows us to prove strong results about  
 100 the accuracy of the classifier under the exchangeable sampling assumption, as we see in Section 3.

## 101 3 Performance extrapolation for generative classifiers

102 Let us specialize to the case of a generative classifier, with classification function  $\mathcal{Q}$ . Consider  
 103 estimating the expected accuracy for the  $k$ -th classification problem,

$$p_k \stackrel{\text{def}}{=} \mathbf{E}[\operatorname{acc}^{(k)}]. \quad (1)$$

104 In the case of a generative classifier, we have

$$p_k = \mathbf{E}[\operatorname{acc}^{(k)}] = \mathbf{E} \left[ \frac{1}{k} \sum_{i=1}^k \Pr_{Y \sim F_i} [\mathcal{Q}(\hat{F}_i, Y) > \max_{j \neq i} \mathcal{Q}(\hat{F}_j, Y)] \right].$$

105 Define the *conditional accuracy* function  $u(\hat{F}, y)$  which maps a distribution  $\hat{F}$  on  $\mathcal{Y}$  and a *test*  
 106 observation  $y$  to a real number in  $[0, 1]$ . The conditional accuracy gives the probability that for  
 107 independently drawn  $\hat{F}'$  from  $\hat{\mathbb{F}}$ , that  $\mathcal{Q}(\hat{F}, y)$  will be greater than  $\mathcal{Q}(\hat{F}', y)$ :

$$u(\hat{F}, y) = \Pr_{\hat{F}' \sim \hat{\mathbb{F}}} [\mathcal{Q}(\hat{F}, y) > \mathcal{Q}(\hat{F}', y)].$$

108 Define the *conditional accuracy* distribution  $\nu$  as the law of  $u(\hat{F}, Y)$  where  $\hat{F}$  and  $Y$  are generated  
 109 as follows: (i) a true distribution  $F$  is drawn from  $\mathbb{F}$ ; (ii) the empirical distribution  $\hat{F}$  is drawn from

<sup>1</sup>As in their framework, we define a classifier as a vector-valued function. However, we introduce the notion of a classifier as a multiple-argument functional on empirical distributions, which echoes the functional formulation of estimators common in the statistical literature.

<sup>2</sup>For QDA, the classification function is given by

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

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

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

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

110  $\hat{\mathbb{F}}(F)$  (i.e., the training data for the class), (iii) the query  $Y$  is drawn from  $F$ , with  $Y$  independent  
 111 of  $\hat{F}$  (i.e. a single test data point from the same class.) The significance of the conditional accuracy  
 112 distribution is that the expected accuracy  $p_t$  can be written in terms of its moments.

113 **Theorem 3.1.** *Let  $\mathcal{Q}$  be a single-distribution classification function, and let  $\mathbb{F}, \hat{\mathbb{F}}(F)$  be a distribution  
 114 on  $\mathcal{P}(\mathcal{Y})$ . Further assume that  $\hat{\mathbb{F}}$  and  $\mathcal{Q}$  jointly satisfy the tie-breaking property:*

$$\Pr[\mathcal{Q}(\hat{F}, y) = \mathcal{Q}(\hat{F}', y)] = 0 \quad (2)$$

115 *for all  $y \in \mathcal{Y}$ , where  $\hat{F}, \hat{F}' \stackrel{iid}{\sim} \hat{\mathbb{F}}$ . Let  $U$  be defined as the random variable  $U = u(\hat{F}, Y)$  for  $F \sim \mathbb{F}$ ,  
 116  $Y \sim F$ , and  $\hat{F} \sim \hat{\mathbb{F}}(F)$  with  $Y \perp \hat{F}$ . Then*

$$p_k = \mathbf{E}[U^{k-1}],$$

117 *where  $p_k$  is the expected accuracy as defined by (1).*

118 (Proof is included in the supplement.)

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

122 While  $U = u(\hat{F}, Y)$  is not directly observed, we can obtain unbiased estimates of  $u(\hat{F}_i, y)$  by using  
 123 test data. For any  $\hat{F}_1, \dots, \hat{F}_k$ , and independent test point  $Y \sim F_i$ , define

$$\hat{u}(\hat{F}_i, Y) = \frac{1}{k-1} \sum_{j \neq i} I(\mathcal{Q}(\hat{F}_i, Y) > \mathcal{Q}(\hat{F}_j, Y)). \quad (3)$$

124 Then  $\hat{u}(\hat{F}_i, Y)$  is an unbiased estimate of  $u(\hat{F}_i, Y)$ , as stated in the following theorem.

125 **Theorem 3.2.** *Assume the conditions of theorem 3.1. Then defining*

$$V = (k-1)\hat{u}(\hat{F}_i, y), \quad (4)$$

126 *we have*

$$V \sim \text{Binomial}(k-1, u(\hat{F}_i, y)).$$

127 *Hence,*

$$\mathbf{E}[\hat{u}(\hat{F}_i, y)] = u(\hat{F}_i, y).$$

128

129 In section 4, we will use this result to estimate the moments of  $U$ . Meanwhile, since  $U$  is a random  
 130 variable on  $[0, 1]$ , we also conclude that  $p_t$  follows a *mixed exponential decay*. Let  $\alpha$  be the law of  
 131  $-\log(U)$ . Then from change-of-variables  $\kappa = -\log(u)$ , we get

$$p_t = \mathbf{E}[U^{t-1}] = \int_0^1 u^{t-1} d\nu(u) = \int_0^1 e^{t \log(u)} \frac{1}{u} d\nu(u) = \int_{\mathbb{R}^+} e^{-\kappa t} d\alpha(\kappa).$$

132 This fact immediately suggests the technique of fitting a mixture of exponentials to the test accuracy  
 133 at  $t = 2, 3, \dots, k$ : we explore this idea further in Section 4.1.

### 134 3.1 Properties of the conditional accuracy distribution

135 The conditional accuracy distribution  $\nu$  is determined by  $\mathbb{F}$  and  $\mathcal{Q}$ . What can we say about the the  
 136 conditional accuracy distribution without making any assumptions on either  $\mathbb{F}$  or  $\mathcal{Q}$ ? The answer is:  
 137 not much. For an arbitrary probability measure  $\nu'$  on  $[0, 1]$ , one can construct  $\mathbb{F}$  and  $\mathcal{Q}$  such that the  
 138 conditional accuracy  $U$  has the distribution  $\nu'$ , even if one makes the *perfect sampling assumption*  
 139 that  $\hat{F} = F$ .

140 **Theorem 3.3.** *Let  $U$  be defined as in Theorem 3.1, and let  $\nu$  denote the law of  $U$ . Then, for any  
 141 probability distribution  $\nu'$  on  $[0, 1]$ , one can construct a meta-distribution  $\mathbb{F}$  and a classification  
 142 function  $\mathcal{Q}$  such that the conditional accuracy  $U$  has distribution  $\nu'$  under perfect sampling (that is,  
 143  $\hat{F} = F$ .)*

144 **Proof.** Let  $G$  be the cdf of  $\nu$ ,  $G(x) = \int_0^x d\nu(x)$ , and let  $H(u) = \sup_x \{G(x) \leq u\}$ . Define  $\mathcal{Q}$  by

$$\mathcal{Q}(\hat{F}, y) = \begin{cases} 0 & \text{if } \mu(\hat{F}) > y + H(y) \\ 0 & \text{if } y + H(y) > 1 \text{ and } \mu(\hat{F}) \in [H(y) - y, y] \\ 1 + \mu(\hat{F}) - y & \text{if } \mu(\hat{F}) \in [y, y + H(y)] \\ 1 + y + \mu(\hat{F}) & \text{if } \mu(\hat{F}) + H(y) > 1 \text{ and } \mu(\hat{F}) \in [0, H(y) - y]. \end{cases}$$

145 Let  $\theta \sim \text{Uniform}[0, 1]$ , and define  $F \sim \mathbb{F}$  by  $F = \delta_\theta$ , and also  $\hat{F} = F$ . A straightforward calculation  
146 yields that  $\nu = \nu'$ .  $\square$

147 On the other hand, we can obtain a positive result if we assume that the classifier approximates a  
148 *Bayes classifier*. Assuming that  $F$  is absolutely continuous with respect to Lebesgue measure  $\Lambda$   
149 with probability one, a Bayes classifier results from assuming perfect sampling ( $\hat{F} = F$ ) and taking  
150  $\mathcal{Q}(\hat{F}, y) = \frac{dF}{d\Lambda}(y)$ . Theorem 3.4. states that for a Bayes classifier, the measure  $\nu$  has a density  
151  $\eta(u)$  which is monotonically increasing. Since a ‘good’ classifier approximates the Bayes classifier,  
152 we intuitively expect that a monotonically increasing density  $\eta$  is a good model for the conditional  
153 accuracy distribution of a ‘good’ classifier.

154 **Theorem 3.4.** Assume the conditions of theorem 3.1, and further suppose that  $\hat{F} = F$ ,  $F$  is  
155 absolutely continuous with respect to  $\Lambda$  with probability one, that  $\mathcal{Q}(\hat{F}, y) = \frac{dF}{d\Lambda}(y)$ , and that  $F|Y$   
156 has a regular conditional probability distribution. Let  $\nu$  denote the law of  $U$ . Then  $\nu$  has a density  
157  $\eta(u)$  on  $[0, 1]$  which is monotonic in  $u$ .

158 **Proof.** It suffices to prove that

$$\nu([u, u + \delta]) < \nu([v, v + \delta])$$

159 for all  $0 < u < v < 1$  and  $0 < \delta < 1 - v$ . Let  $\mathcal{P}_{ac}(\mathcal{Y})$  denote the space of distributions supported  
160 on  $\mathcal{Y}$  which are absolutely continuous with respect to  $p$ -dimensional Lebesgue measure  $\Lambda$ . Let  $\mathbb{Y}$   
161 denote the marginal distribution of  $Y$  for  $Y \sim F$  with  $F \sim \mathbb{F}$ . Define the set

$$J_y(A) = \{F \in \mathcal{P}_{ac}(\mathcal{Y}) : u(F, y) \in A\}.$$

162 for all  $A \subset [0, 1]$ . One can verify that for all  $y \in \mathcal{Y}$ ,

$$\Pr_{\mathbb{F}}[J_y([u, u + \delta])|Y = y] \leq \Pr_{\mathbb{F}}[J_y([v, v + \delta])|Y = y],$$

163 using the fact that  $\mathbb{F}$  has no atoms. Hence, we obtain

$$\Pr[U \in [u - \delta, u + \delta]] = \mathbf{E}_{\mathbb{Y}}[\Pr[J_Y([u, u + \delta])|Y]] \leq \mathbf{E}_{\mathbb{Y}}[\Pr[J_Y([v, v + \delta])|Y]] = \Pr[U \in [v - \delta, v + \delta]].$$

164 Taking  $\delta \rightarrow 0$ , we conclude the theorem.  $\square$

165

## 166 4 Estimation

167 Suppose we have  $m$  independent test repeats per class,  $y^{(i),1}, \dots, y^{(i),m}$ . Let us define

$$V_{i,j} = \sum_{\ell \neq i} I(\mathcal{M}_i(\hat{F}_1, \dots, \hat{F}_k, y^{(i,j)}) > \mathcal{M}_\ell(\hat{F}_1, \dots, \hat{F}_k, y^{(i,j)})),$$

168 which coincides with the definition (4) in the special case that  $\mathcal{M}$  is generative.

169 At a high level, we have a hierarchical model where  $U$  is drawn from a distribution  $\nu$  on  $[0, 1]$  and  
170 then  $V_{i,j} \sim \text{Binomial}(k, U)$ . Let us assume that  $U$  has a density  $\eta(u)$ : then the marginal distribution  
171 of  $V_{i,j}$  can be written

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

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

173 We discuss the following three approaches for estimating  $p_t = \mathbf{E}[U^{t-1}]$  based on  $V_{i,j}$ . The first is  
174 an extension of *unbiased estimation* based on binomial U-statistics, which is discussed in Section

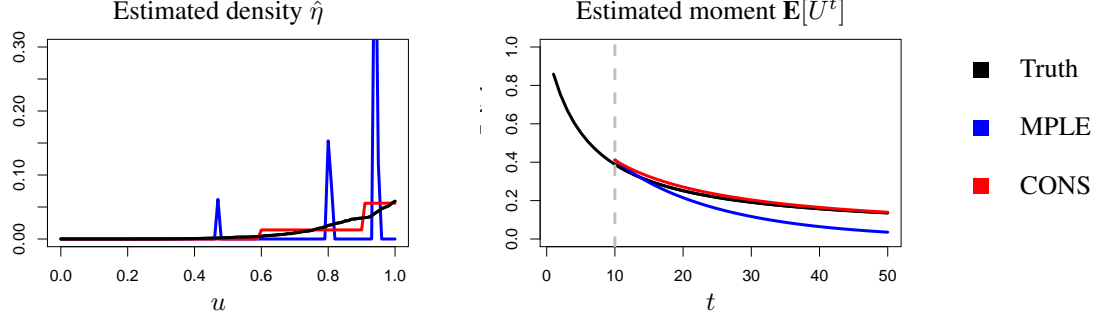


Figure 1: Maximum pseudolikelihood (MPLE) versus constrained pseudolikelihood (CONS). Adding constraints improves the estimation of the density  $\eta(u)$ , as well as moment estimation.

175 4.1. The second is the *pseudolikelihood* approach. In problems where the marginal distributions  
 176 are known, but the dependence structure between variables is unknown, the *pseudolikelihood* is  
 177 defined as the product of the marginal distributions. For certain problems in time series analysis and  
 178 spatial statistics, the maximum pseudolikelihood estimator (MPLE) is proved to be consistent [7]. We  
 179 discuss pseudolikelihood-based approaches in Section 4.2. Thirdly, we note that the high-dimensional  
 180 theory of Anon 2016 [4] can be applied for prediction accuracy, which we discuss in Section 4.3.

#### 181 4.1 Extensions of unbiased estimation

182 If  $V \sim \text{Binomial}(k, U)$ , then an unbiased estimator of  $U^t$  exists if and only if  $0 \leq t \leq k$ .

183 The theory of U-statistics [8] provides the minimal variance unbiased estimator for  $U^t$ :

$$U^t = \mathbf{E} \left[ \binom{V}{t} \binom{k}{t}^{-1} \right].$$

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

$$\hat{p}_t^{UN} = \frac{1}{km} \sum_{i=1}^k \sum_{j=1}^m \binom{V_{i,j}}{t-1} \binom{k}{t-1}^{-1}. \quad (5)$$

185 However, since  $\hat{p}_t^{UN}$  is undefined for  $k \geq t$ , we can use exponential extrapolation to define an  
 186 extended estimator  $\hat{p}_t^{EXP}$  for  $k > t$ . Let  $\hat{\alpha}$  be a measure defined by solving the optimization problem

$$\text{minimize}_{\alpha} \sum_{t=2}^k \left( \hat{p}_t^{UN} - \int_0^{\infty} \exp[-t\kappa] d\hat{\alpha}(\kappa) \right)^2.$$

187 After discretizing the measure  $\hat{\alpha}$ , we obtain a convex optimization problem which can be solved  
 188 using non-negative least squares [9]. Then define

$$\hat{p}_t^{EXP} = \begin{cases} \hat{p}_t^{UN} & \text{for } t \leq k, \\ \int_0^{\infty} \exp[-t\kappa] d\hat{\alpha}(\kappa) & \text{for } t > k. \end{cases}$$

#### 189 4.2 Maximum pseudolikelihood

190 The (log) pseudolikelihood is defined as

$$\ell(\eta) = \sum_{i=1}^k \sum_{j=1}^m \log \left( \int u^{V_{i,j}} (1-u)^{k-V_{i,j}} \eta(u) du \right), \quad (6)$$

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

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

192 The motivation for  $\hat{\eta}_{MPLE}$  is that it consistently estimates  $\eta$  in the limit where  $k \rightarrow \infty$ . However, in  
 193 finite samples,  $\hat{\eta}_{MPLE}$  is not uniquely defined, and if we define the plug-in estimator

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

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

196 Theorem 3.4. motivates the *monotonicity constraint* that  $\frac{d\hat{\eta}}{du} > 0$ . A second constraint is to restrict  
 197 the  $k$ -th moment of  $\hat{\eta}$  to match the unbiased estimate. The addition of these constraints yields the  
 198 constrained PMLE  $\hat{\eta}_{CON}$ , which is obtained by solving

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

199 By discretizing  $\eta$ , all of the above maximization problems can be solved using a general-purpose  
 200 convex solver<sup>3</sup>. While the added constraints do not guarantee a unique solution, they improve  
 201 estimation of  $\eta$  and thus improve moment estimation (Figure 1.)

### 202 4.3 High-dimensional asymptotics

203 Under a number of conditions on the distribution  $\mathbb{F}$ , including (but not limited to) having a large  
 204 dimension  $p$ , Anon [4] relate the accuracy  $p_t$  of the Bayes classifier to the mutual information between  
 205 the label  $z$  and the response  $y$ :

$$p_t = \bar{\pi}_t(\sqrt{2I(Z; Y)}).$$

206 where

$$\bar{\pi}_k(c) = \int_{\mathbb{R}} \phi(z - c) \Phi(z)^{k-1} dz.$$

207 While our goal is not to estimate the mutual information, we note that the results of Anon 2016 imply  
 208 a relationship between  $p_k$  and  $p_K$  for the Bayes accuracy under the high-dimensional regime:

$$p_K = \bar{\pi}_K(\bar{\pi}_k^{-1}(p_k)).$$

209 Therefore, under the high-dimensional conditions of [4] and assuming that the classifier approximates  
 210 the Bayes classifier, we naturally obtain the following estimator

$$\hat{p}_t^{HD} = \bar{\pi}_K(\bar{\pi}_k^{-1}(\hat{p}_k^{UN})).$$

## 211 5 Results

212 We applied the methods described in Section 4 on a simulated gaussian mixture (Figure 2) and on a  
 213 Telugu character classification task [12] (Table 1.)

214 For the simulated gaussian mixture, we vary the size of the initial subset from  $k = 3$  classes to  
 215  $k = K = 50$  classes, and extrapolate the performance for gaussian mixture model, multinomial  
 216 logistic, and one-layer neural network (with 10 units). Figure 3 shows how the predicted  $K$ -class  
 217 accuracy changes as  $k$  is varied. We see that the predicted accuracy curves for QDA and Logistic  
 218 have similar behavior, even though QDA is generative and multinomial logistic is not. All three  
 219 methods perform better on QDA and logistic classifiers than on the neural network: in fact, for the  
 220 neural network, the test accuracy of the initial set,  $\text{acc}^{(k)}$ , becomes a better estimator of  $\text{acc}^{(K)}$  than  
 221 the three proposed methods for most of the curve. We also see that the exponential extrapolation  
 222 method,  $\hat{p}^{EXP}$ , is more variable than constrained pseudolikelihood  $\hat{p}^{CONS}$  and high-dimensional  
 223 estimator  $\hat{p}^{HD}$ . Additional simulation results can be found in the supplement.

224 In the character classification task, we predict the 400-class accuracy of naive Bayes, multinomial  
 225 logistic regression, SVM [6],  $\epsilon$ -nearest neighbors<sup>4</sup>, and deep neural networks<sup>5</sup> using 20-class data

<sup>3</sup> We found that the disciplined convex programming language CVX, using the ECOS second-order cone programming solver, succeeds in optimizing the problems where the dimension of the discretized  $\eta$  is as large as 10,000 [10, 11].

<sup>4</sup>  $k$ -nearest neighbors with  $k = \epsilon n$  for fixed  $\epsilon > 0$

<sup>5</sup> The network architecture is as follows: 48 x 48 binary input image, (mc3) a 3x3 convolutional layer with m output maps, (MP2) a 2x2 max-pooling layer, and (400SM) a softmax output layer on 400 classes.

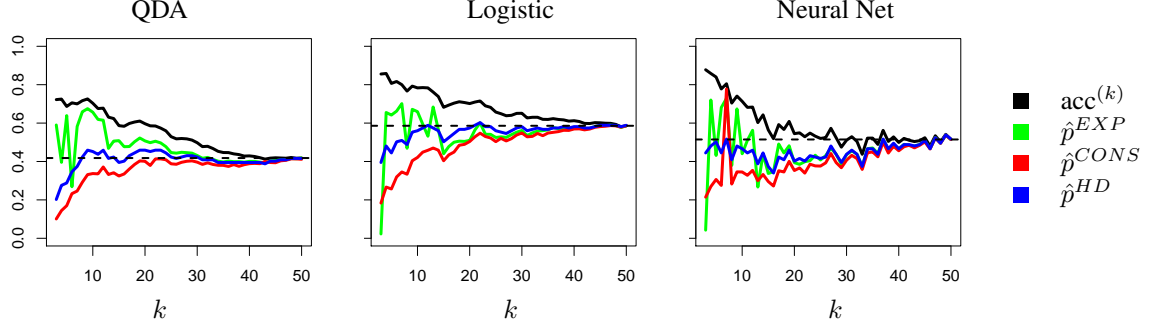


Figure 2: Predictions for  $\text{acc}^{(50)}$  as  $k$ , the size of the subset, is varied. Our methods work better for QDA and Logistic than Neural Net; overall,  $\hat{p}^{EXP}$  has higher variability than  $\hat{p}^{CONS}$  and  $\hat{p}^{HD}$ .

Classifier	Test $\text{acc}^{(20)}$	Test $\text{acc}^{(400)}$	$\hat{p}_{400}^{EXP}$	$\hat{p}_{400}^{CONS}$	$\hat{p}_{400}^{HD}$
Naive Bayes	0.947	0.601	0.884	<b>0.659</b>	0.769
Logistic	0.922	0.711	0.844	<b>0.682</b>	0.686
SVM	0.860	0.545	0.737	0.473	<b>0.546</b>
$\epsilon$ -NN	0.964	0.591	0.895	<b>0.395</b>	0.839
Deep neural net	<b>0.995</b>	0.986	0.973	(*)	0.957

Table 1: Performance extrapolation: predicting the accuracy on 400 classes using data from 20 classes on a Telugu character dataset. (\*) indicates failure to converge.  $\epsilon = 0.002$  for  $\epsilon$ -nearest neighbors.

with 103 training examples per class (Table 1). Taking the test accuracy on 400 classes (using 50 test examples per class) as a proxy for  $\text{acc}^{(400)}$ , we compare the performance of the three extrapolation methods; as a benchmark, also consider using the test accuracy on 20 classes as an estimate. The exponential extrapolation method performs well only for the deep neural network. Meanwhile, constrained PMLLE achieves accurate extrapolation for two out of four classifiers: logistic and SVM but failed to converge for the deep neural network (due to the high test accuracy). The high-dimensional estimator  $\hat{p}^{HD}$  performs well on the multinomial logistic, SVM, and deep neural network classifiers. All three methods beat the benchmark (taking the test accuracy at 20) for the first four classifiers; however, the benchmark is the best estimator for the deep neural network, similarly to what we observe in the simulation (albeit with a shallow network rather than a deep network.)

## 6 Discussion

Empirical results indicate that our methods generalize beyond generative classifiers. A possible explanation is that since the Bayes classifier is generative, any classifier which approximates the Bayes classifier is also ‘approximately generative.’ However, an important caveat is that the classifier must already attain close to the Bayes accuracy on the smaller subset of classes. If the classifier is initially far from the Bayes classifier, and then becomes more accurate as more classes are added, our theory could underestimate the accuracy on the larger subset. This is a non-issue for generative classifiers when the training data per class is fixed, since a generative classifier approximates the Bayes rule if and only if the single-class classification function approximates the Bayes optimal single-class classification function. On the other hand, for classifiers with built-in *model selection* or *representation learning*, it is expected that the classification functions become more accurate, in the sense that they better approximate a monotonic function of the Bayes classification functions, as data from more classes is added.

Our results are still too preliminary for us to recommend the use of any of these estimators in practice. Theoretically, it still remains to derive confidence bounds for the generative case; practically, additional experiments are needed to establish the reliability of these estimators in specific applications. There also remains plenty of room for new and improved estimators in this area: for instance, fixing the instability of the constrained pseudolikelihood estimator when the test accuracy is high.



## References

- [1] Kay, K. N., Naselaris, T., Prenger, R. J., & Gallant, J. L. (2008). "Identifying natural images from human brain activity." *Nature*, 452(March), 352-355.
- [2] Deng, J., Berg, A. C., Li, K., & Fei-Fei, L. (2010). "What does classifying more than 10,000 image categories tell us?" *Lecture Notes in Computer Science*, 6315 LNCS(PART 5), 71-84.
- [3] Garfield, S., Stefan W., & Devlin, S. (2005). "Spoken language classification using hybrid classifier combination." *International Journal of Hybrid Intelligent Systems* 2.1: 13-33.
- [4] Anonymous, A. (2016). "Estimating mutual information in high dimensions via classification error." Submitted to *NIPS 2016*.
- [5] Tewari, A., & Bartlett, P. L. (2007). "On the Consistency of Multiclass Classification Methods." *Journal of Machine Learning Research*, 8, 1007-1025.
- [6] Hastie, T., Tibshirani, R., & Friedman, J., (2008). *The elements of statistical learning*. Vol. 1. Springer, Berlin: Springer series in statistics.
- [7] Arnold, Barry C., & Strauss, D. (1991). "Pseudolikelihood estimation: some examples." *Sankhya: The Indian Journal of Statistics, Series B*: 233-243.
- [8] Cox, D.R., & Hinkley, D.V. (1974). *Theoretical statistics*. Chapman and Hall. ISBN 0-412-12420-3
- [9] Lawson, C. L., & Hanson, R. J. (1974). *Solving least squares problems*. Vol. 161. Englewood Cliffs, NJ: Prentice-hall.
- [10] Hong, J., Mohan, K. & Zeng, D. (2014). "CVX. jl: A Convex Modeling Environment in Julia."
- [11] Domahidi, A., Chu, E., & Boyd, S. (2013). "ECOS: An SOCP solver for embedded systems." *Control Conference (ECC), 2013 European. IEEE*.
- [12] Achanta, R., & Hastie, T. (2015) "Telugu OCR Framework using Deep Learning." arXiv preprint arXiv:1509.05962 .