Convergence rates of the Voting Gibbs classifier, with application to Bayesian feature selection

Andrew Y. Ng ANG@CS.BERKELEY.EDU
Computer Science Division, University of California, Berkeley, CA 94720

Michael I. Jordan JORDAN@CS.BERKELEY.EDU
Computer Science Division & Department of Statistics, University of California, Berkeley, CA 94720

Abstract

The Gibbs classifier is a simple approximation to the Bayesian optimal classifier in which one samples from the posterior for the parameter θ, and then classifies using the single classifier indexed by that parameter vector. In this paper, we study the *Voting Gibbs classifier*, which is the extension of this scheme to the full Monte Carlo setting, in which N samples are drawn from the posterior and new inputs are classified by voting the N resulting classifiers. We show that the error of Voting Gibbs converges rapidly to the Bayes optimal rate; in particular the relative error decays at a rapid $O(1/N)$ rate. We also discuss the feature selection problem in the Voting Gibbs context. We show that there is a choice of prior for Voting Gibbs such that the algorithm has high tolerance to the presence of irrelevant features. In particular, the algorithm has sample complexity that is *logarithmic* in the number of irrelevant features.

1. Introduction

Bayesian methods for reasoning about uncertainty have a natural appeal, and the increasing availability of approximation algorithms has played an important role in making these methods practical. Some of these approximation methods are, however, poorly understood. In this paper, we consider an elementary, yet foundational, question regarding the performance of sampling-based approximation methods in the setting of Bayesian classification.

Consider a setting in which we have a family of discriminative classifiers parameterized by θ. After observing some number of training examples, we obtain a posterior distribution on θ. When asked to classify a new example, exact Bayesian inference demands that we integrate over θ to determine the posterior distribution of the class label y. But this integral is often difficult to perform. A simple approximation is provided by the Gibbs classifier, which draws a single sample y from the posterior distribution of the class label, and uses that y as its prediction. It is well known that the Gibbs classifier has error at most twice that of the Bayesian optimal classifier.

In this paper, we consider the generalization of the Gibbs classifier to the full Monte Carlo setting, in which we instead draw N samples $y^1, \ldots, y^N$ from the posterior distribution, and take a majority vote of these samples to obtain the final prediction. We refer to this as the Voting Gibbs algorithm (cf. Green, 1995; Sykacek, 2000; Denison and Mallick, 2000). We ask the elementary yet important question of how it performs relative to the Bayesian optimal classifier. We show that (under mild assumptions) the relative error of Voting Gibbs compared to the Bayesian optimal classifier decays at the rapid rate of $O(1/N)$.

We also address the case in which our learning algorithm may use a prior over θ that is different from the "true" prior. There are several reasons that we believe that this case of "misspecified priors" is an important aspect of our analysis: (1) it can be costly to elicit a prior from experts, and simplified "textbook" priors are often substituted; (2) even if a realistic prior is available, it can be computationally intractable to implement this prior; (3) simplified priors can be easier to understand. Moreover, there is an interesting and somewhat surprising application of our results on misspecified priors to the problem of feature selection. In particular, we show that a Voting Gibbs algorithm that uses a particular misspecified prior has sample complexity that is *logarithmic* in the number of irrelevant features, a result that matches the best known results for feature selection problems in a frequentist

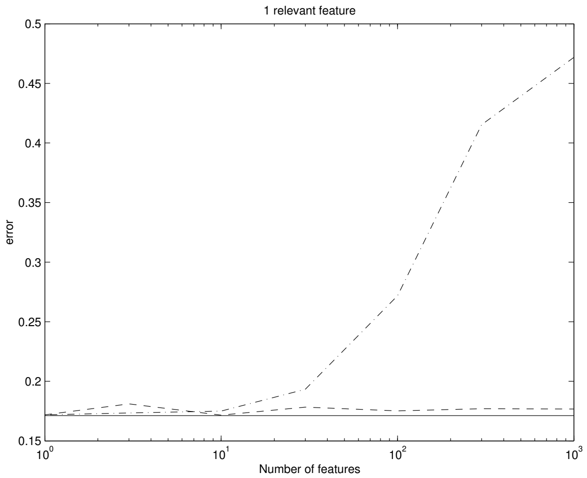


Figure 1. Plot of error vs. total number of features for an optimal classifier that knows which feature is relevant to a classification decision (solid), and Voting Gibbs algorithms using different priors (dash and dash-dot). Details are provided in Section 5.

setting (Ng, 1998; Littlestone, 1988; Kivinen and Warmuth, 1994). That this result is not merely of theoretical interest is demonstrated by the empirical results shown in Figure 1. These results, which are described in more detail in Section 5, show classification error rates in an experiment in which one feature is relevant to a classification decision. The solid curve plots the error rate for an algorithm that is told in advance which feature is relevant. The other two curves show the error rates for Voting Gibbs algorithms which are not told which feature is relevant and which make use of different priors. The high degree of insensitivity to irrelevant features exhibited by the lower (dashed) curve is surprising and noteworthy.

The remainder of this paper is structured as follows. Section 2 provides a formal introduction to the problem and the algorithms. Section 3 then presents our main results on the quality of the Voting Gibbs classifier in the case of a correctly specified prior, and Section 4 goes on to discuss Voting Gibbs in the context of misspecified priors, with application to feature selection. Lastly, Section 5 presents experimental results, and Section 6 closes with our conclusions.

2. Problem definition and notation

2.1 Bayesian classification

We are concerned with the problem of Bayesian classification in the discriminative setting, where given an input $x \in X$, we wish to predict the corresponding label $y \in \{0, 1\}$. Formally, we assume a family of probabilistic binary predictors $\{f_\theta : X \mapsto [0,1] \mid \theta \in \Theta\}$ parameterized by $\theta \in \Theta$, where $f_\theta(x)$ is interpreted as the probability that y is 1 given x. For example, for Bayesian logistic regression, we would use $f_\theta(x) = 1/(1 + \exp(-w^T x - \beta))$, parameterized by $\theta = (w, \beta)$. Since we are in a Bayesian setting, we also have a prior distribution $p_\Theta(\cdot)$ over θ.

We also assume a fixed distribution D over X from which training examples are drawn iid. We are given a training set $S = \{(x_i, y_i)\}_{i=1}^m$ of m examples, generated by first sampling θ^* according to the prior p_Θ, then sampling x_i iid according to D, and finally setting each y_i independently to 1 or 0 according to the probabilities $\Pr[y_i = 1|x_i, \theta^*] = f_{\theta^*}(x_i)$. Finally, let $p_\Theta(\theta|S)$ denote the posterior distribution of θ given the dataset S. We explicitly allow the case of $m = 0$ training examples, in which case the posterior reduces to the prior.

A classifier is any (possibly stochastic) map $h : X \mapsto \{0, 1\}$. For example, the familiar Bayesian optimal classifier h_B is obtained by calculating

$$\Pr[y = 1|x, S] = \int_\theta f_\theta(x) p_\Theta(\theta|S) d\theta \qquad (1)$$

and then predicting $h_B(x|S) = h_B(x) = 1$ if $\Pr[y = 1|x, S] \geq 0.5$, and predicting 0 otherwise.

2.2 The Voting Gibbs classifier

We let $\hat{p}_\Theta$ denote a prior used by a learning procedure. When $\hat{p}_\Theta \neq p_\Theta$, we say $\hat{p}_\Theta$ is a misspecified prior. Note that when we refer to the "Bayesian optimal classifier," we always mean the classifier that uses p_Θ.

When a Gibbs classifier h_G using $\hat{p}_\Theta$ is required to classify x, it first samples $\hat{\theta}$ according to the (possibly misspecified) posterior distribution $\hat{p}_\Theta(\cdot|S)$, then further samples $\hat{y}$ so that $\hat{y} = 1$ with probability $f_{\hat{\theta}}(x)$ and $\hat{y} = 0$ with probability $1 - f_{\hat{\theta}}(x)$. Finally, its prediction is $h_G(x|S) = h_G(x) = \hat{y}$.

We are interested in the performance of the extension of Gibbs classifiers to the full Monte Carlo setting, in which multiple samples are taken from the posterior. We call this the Voting Gibbs (VG) classifier. When asked to predict a label for an input x, the Voting Gibbs classifier VG(N) first draws N (a parameter) iid samples $\theta^1, \ldots, \theta^N$ according to the posterior distribution $\hat{p}_\Theta(\cdot|S)$. Then, it further samples $y^1, \ldots, y^N$ independently, setting $y^i = 1$ with probability $f_{\theta^i}(x)$, and $y^i = 0$ with probability $1 - f_{\theta^i}(x)$. Finally it picks its output by taking a majority-vote of the y^i, predicting 1 if $\hat{y} = (1/N)\sum_{i=1}^N y^i \geq 0.5$, and 0 otherwise. (Alternatively, one may also skip the second stage of sampling and predict $h_{VG(N)}(x|S) = h_{VG(N)}(x) = 1$

when $(1/N)\sum f_{\theta^i}(x) \geq 0.5$, and 0 otherwise. This algorithm, which skips one step of randomization, is probably more appealing to many, and is also be used in some of our experiments. All of our analyses and results also apply to it.)

Voting Gibbs uses a Monte Carlo approximation to the Bayesian optimal classifier, and VG(1) is the Gibbs classifier. Voting Gibbs should be thought of as using samples to obtain a Monte Carlo estimate of $\Pr[y = 1|S, x]$, and then thresholding its estimate at 0.5 to make its prediction. Folk wisdom suggests that only a small number of Monte Carlo samples are needed in order to do well in the Bayesian classification setting. We seek to investigate the degree to which this is true.

An important feature of the Voting Gibbs classifier is that it can draw its samples $\theta^1, \ldots, \theta^N$ off-line before the new input vector is presented. When expensive methods such as Markov chain Monte Carlo (Gilks et al., 1996) or rejection sampling (Ripley, 1987) are required to draw the samples, this enables us to perform the expensive sampling offline, making the algorithm subsequently able to classify individual inputs quickly.

2.3 Error metrics

Given a training set S, the expected generalization error of hypothesis h on a particular input $x \in X$ is $\varepsilon_x(h) = \Pr[h(x) \neq y|x, S] = \int_\theta \Pr[h(x) \neq y|x, \theta]p_\Theta(\theta|S)d\theta$, where the probability is over any randomization in h and in the uncertainty in y given S. Note that this is the *Bayesian expected error* and is averaged over θ, which differs from the PAC notion of error in which misclassification is measured with respect to a single "true" θ (Valiant, 1984).

In some cases it may be appropriate to view the test set as generated from a distribution that is different from the training distribution. We assume a testing distribution D' over the input space X. We then define the generalization error of a classifier h to be

$$\varepsilon(h) = \varepsilon_{D',S}(h) = \mathrm{E}_{x\sim D'}\left[\Pr[h(x) \neq y|x, S]\right] \qquad (2)$$

where the subscript $x \sim D'$ means the expectation is with respect to x distributed according to D', and we have again used a Bayesian notion of error.

In this paper, we are concerned with how well the Voting Gibbs classifier approximates the Bayesian optimal classifier h_B. There are two standard ways to quantify this. Given a classifier h, we define its *additional absolute error* (compared to the Bayesian optimal classifier) to be

$$\varepsilon_{D',S}(h) - \varepsilon_{D',S}(h_B). \qquad (3)$$

We also define its *additional relative error* to be

$$\frac{\varepsilon_{D',S}(h) - \varepsilon_{D',S}(h_B)}{\varepsilon_{D',S}(h_B)}. \qquad (4)$$

Note that these two measures of error are closely related. For example, an upper bound on additional relative error immediately implies a bound on additional absolute error,[1] and an upper bound on additional absolute error with a lower bound on the Bayes error similarly implies a bound on additional relative error.

It seems likely that, at least in the case of correct priors, the performance of VG(N) will improve as N becomes large, approaching the Bayes error in the limit of $N \to \infty$. Since the running time of VG(N) is linear in N, it is important for practical applications to quantify exactly how quickly this happens. The next section will study the rate at which the performance of VG(N) approaches the Bayes error, for several learning and non-learning scenarios.

3. Voting Gibbs with correct priors

This section presents our results on the rate at which the error of Voting Gibbs approaches the Bayesian error. For now, we treat only the case of "correct" priors, $\hat{p}_\Theta = p_\Theta$. The case of misspecified priors is left to Section 4.

When $\hat{p}_\Theta = p_\Theta$, training does not play a significant role, since identical priors give identical posteriors, and so if we can prove a bound for an arbitrary prior (when there is no training data), then we may define that prior to be the "posterior" $\hat{p}_\Theta(\cdot|S)$, and thereby also obtain a bound for the case of learning from data. This will turn out to be more complicated when we begin to consider misspecified priors in the next section.

For the case of correctly specified priors, we have the theorem below, given in two parts. The first part states that even in the worst case, the additional relative error of Voting Gibbs is at most $O(1/\sqrt{N})$. (The full paper (Ng & Jordan, 2001) shows that this is tight if we make no additional assumptions.) The intuition behind this result is that, since VG(N) is averaging over N random samples to estimate $\Pr[y = 1|x]$, the standard deviation of these estimates is at most $O(1/\sqrt{N})$, and hence so is the additional error. The second part of the theorem shows that, under an additional (and fairly mild) technical assumption, the additional relative error of Voting Gibbs can be shown to decay at the much faster rate of $O(1/N)$.

[1] To see this, note that $\varepsilon(h) - \varepsilon(h_B) \leq (\varepsilon(h) - \varepsilon(h_B))/2\varepsilon(h_B)$, since $\varepsilon(h_B) \leq 0.5$.

Theorem 1 *Let X, D', S be fixed, and suppose $\hat{p}_\Theta = p_\Theta$. Then the additional relative error of Voting Gibbs (compared to the Bayes optimal classifier) is upper-bounded by*

$$\frac{\varepsilon_{D',S}(h_{VG(N)}) - \varepsilon_{D',S}(h_B)}{\varepsilon_{D',S}(h_B)} = O\left(\frac{1}{\sqrt{N}}\right) \quad (5)$$

where the big-O does not hide any terms that depend on X, D', S, or p_Θ. Now suppose we further assume that D', S, and p_Θ are such that the random variable $\overline{y}(x) = \Pr[y = 1|x, S]$ (whose distribution is induced by $x \sim D'$) has a density $p(\overline{y})$, so that within some small interval $[0.5-\delta, 0.5+\delta]$, $p(\overline{y})$ does not vary too much: That there is some constant $B > 0$ so that $\sup_{\overline{y}\in[0.5-\delta,0.5+\delta]} p(\overline{y}) \leq B \inf_{\overline{y}\in[0.5-\delta,0.5+\delta]} p(\overline{y})$. Then the additional relative error of Voting Gibbs is upper-bounded by

$$\frac{\varepsilon_{D',S}(h_{VG(N)}) - \varepsilon_{D',S}(h_B)}{\varepsilon_{D',S}(h_B)} = O\left(\frac{1}{N}\right), \quad (6)$$

where the big-O notation hides constants only depending on B and δ.

Proof (Sketch). Due to space constraints, we only prove here that the additional absolute (rather than relative) error $\varepsilon_{D',S}(h_{VG(N)}) - \varepsilon_{D',S}(h_B)$ is bounded by these $O(1/\sqrt{N})$ and $O(1/N)$ quantities. The proofs for additional relative error are given in the full version of this paper (Ng & Jordan, 2001).

To prove the $1/\sqrt{N}$ bound, we show that the additional absolute error on any particular input x is bounded by

$$\varepsilon_x(h_{VG(N)}) - \varepsilon_x(h_B) \leq \frac{1}{\sqrt{eN}}. \quad (7)$$

By relabeling outputs if necessary, we may assume without loss of generality that $\overline{y} = \overline{y}(x) = \Pr[y = 1|x, S] \leq 0.5$ for all x. Let any $x \in X$ be fixed, and assume $h_B(x) = 0$. Note that the Bayesian expected error on x is just $\varepsilon_x(h_B) = \overline{y}$ (since h_B predicts 0, and there is a $\overline{y} = \overline{y}(x)$ chance the label is 1). The expected error of $h_{VG(N)}$ is

$$\begin{aligned}
\varepsilon_x(h_{VG(N)}) &= (1-\overline{y})\Pr[h_{VG(N)}(x) = 1] \\
&\quad + \overline{y}(1 - \Pr[h_{VG(N)}(x) = 1]) \\
&= \overline{y} + (1-2\overline{y})\Pr[h_{VG(N)}(x) = 1] \\
&\leq \overline{y} + (1-2\overline{y})\Pr[\hat{y} \geq 0.5]
\end{aligned}$$

where $\hat{y} = (1/N)\sum_{i=1}^N y^i$ is the average of the N samples drawn by $VG(N)$.[2] Note $\hat{y}$ has expectation $\overline{y}$.

[2] $\hat{y} = (1/N)\sum f_{\theta^i}(x)$ also works.

Thus,

$$\begin{aligned}
\varepsilon_x(h_{VG(N)}) - \varepsilon_x(h_B) &\leq \overline{y} + (1-2\overline{y})\Pr[\hat{y} \geq 0.5] - \overline{y} \\
&= (1-2\overline{y})\Pr[\hat{y} \geq (0.5-\overline{y}) + \overline{y}] \\
&\leq (1-2\overline{y})\exp(-2(0.5-\overline{y})^2 N) \\
&= (1-2\overline{y})\exp(-(1-2\overline{y})^2 N/2) \\
&\leq \sup_{\gamma\in\mathbb{R}_0^+} \gamma\exp(-\gamma^2 N/2) \\
&= \frac{e^{-1/2}}{\sqrt{N}}
\end{aligned}$$

where for the second inequality, we used the Hoeffding inequality (also referred to as the additive form of the Chernoff bound; see, e.g., Kearns and Vazirani, 1994), which bounds the chance of the mean of N iid random variables being far from the expected value. This proves Equation (7). Taking expectations on both sides with respect to $x \sim D'$ gives $\varepsilon_{D',S}(h_{VG(N)}) - \varepsilon_{D',S}(h_B) \leq 1/\sqrt{eN}$, which completes the first part of the proof.

For the $O(1/N)$ bound, assume as before that $\overline{y}(x) = \Pr[y = 1|x, S] \leq 0.5$ for all x. Also assume without loss of generality that $\delta \leq 0.25$. Showing an $O(1/N)$ additional absolute (rather than relative) error bound actually requires weaker assumptions on $\overline{y}$'s density than stated in the theorem; we require only that there exists a constant B' so that $\sup_{\overline{y}\in[0.5-\delta,0.5+\delta]} p(\overline{y}) \leq B'$. (It is easily verified that this is satisfied by picking $B' = 2B/\delta$, since otherwise the density $p(\overline{y})$ would integrate to greater than 1, a contradiction.) We can write the additional absolute error as

$$\begin{aligned}
&\int_0^{0.5} (\Pr[h_{VG(N)}(x) \neq y|\overline{y}] - \overline{y}) p(\overline{y}) d\overline{y} \\
&\quad \leq \int_0^{0.5} (\overline{y} + (1-2\overline{y})\Pr[\hat{y} \geq 0.5|\overline{y}] - \overline{y}) p(\overline{y}) d\overline{y} \\
&\quad = \int_0^{0.5} (1-2\overline{y})\Pr[\hat{y} \geq 0.5|\overline{y}] p(\overline{y}) d\overline{y} \\
&\quad \leq \int_0^{0.5-\delta} \Pr[\hat{y} \geq 0.5|\overline{y}] p(\overline{y}) d\overline{y} \\
&\qquad + \int_{0.5-\delta}^{0.5} (1-2\overline{y})\Pr[\hat{y} \geq 0.5|\overline{y}] p(\overline{y}) d\overline{y}
\end{aligned}$$

If we can show that each of the above two integrals is $O(1/N)$, then we are done. The first is easy. For $\overline{y} \leq 0.5-\delta$, $\Pr[\hat{y} \geq 0.5|\overline{y}] \leq \exp(-2\delta^2 N)$ (by the Hoeffding inequality again) so $\int_0^{0.5-\delta} \Pr[\hat{y} \geq 0.5|\overline{y}] p(\overline{y}) d\overline{y} \leq \exp(-2\delta^2 N) = O(1/N)$. For the second integral, we can again apply the Hoeffding inequality, to get:

$$\begin{aligned}
&\int_{0.5-\delta}^{0.5} (1-2\overline{y})\Pr[\hat{y} \geq 0.5|\overline{y}] p(\overline{y}) d\overline{y} \\
&\quad \leq \int_{0.5-\delta}^{0.5} 2(0.5-\overline{y})\exp(-2(0.5-\overline{y})^2 N) B' d\overline{y}
\end{aligned}$$

$$
\begin{aligned}
&\leq \int_{-\infty}^{0.5} 2(0.5-\overline{y})\exp(-2(0.5-\overline{y})^2 N)B' d\overline{y} \\
&= \int_0^{\infty} 2t\exp(-2t^2 N)B' dt \\
&= \frac{B'}{2N} \\
&= O\left(\frac{1}{N}\right).
\end{aligned}
$$

This completes the proof. □

Remark (non-triviality conditions for relative error). Additional relative error is just additional absolute error divided by $\varepsilon(h_B)$. So once X, D', S and p_Θ are fixed, a $O(\cdot)$-statement on additional relative error would seem no more interesting than one on additional absolute error. However, the notes in the theorem on the big-O notation make it clear that we are showing something stronger than this, and in particular that we are *not* absorbing a $1/\varepsilon(h_B)$ term into the big-O notation; in this sense, these are indeed "honest" bounds on relative and not just absolute error.

Note also that these bounds on the number of samples N needed have *no dependence* on quantities such as the dimension of the parameter vector θ or the input space X.

4. Learning with misspecified priors, with application to feature/model selection

In this section, we study the case of misspecified priors, $\hat{p}_\Theta \neq p_\Theta$. As a motivating example of our results, we give our first theorem in terms of a result on feature selection.

Let there be a classification problem where the inputs X have f features, of which an unknown subset is relevant. More specifically, let $R \in \{0,1\}^f$ be a random variable that is a string of f bits that indicates whether each of the f features is relevant. Our prior $\hat{p}_\Theta$ assumes that the subset of relevant features is picked randomly according to the following procedure:

1. First, the number r of relevant features is chosen uniformly from $\{0, 1, \ldots, f\}$.
2. Second, one of the $\binom{f}{r}$ subsets of r out of the f features is chosen randomly.

Note therefore that a *particular* feature subset of size r has chance $1/\binom{f}{r}(f+1)$ of being chosen.

Next, we also assume that, conditioned on R, we have some prior $\hat{p}_\Theta(\theta|R)$ (so that, e.g., for all θ to which $\hat{p}_\Theta(\theta|R)$ assigns positive probability, the classifier $f_\theta(\cdot)$ examines the i-th feature of its inputs only if $R_i = 1$). For instance, for logistic regression where $f_\theta(x) = f_{w,\beta}(x) = 1/(1+\exp(-w^T x - \beta))$, we may have $\hat{p}(w, \beta|R)$ drawing w_i from a Normal$(0, \sigma_1^2)$ distribution if $R_i = 1$, and $w_i = 0$ otherwise.

We are interested in evaluating how well a VG algorithm can perform feature selection. Therefore, we want to compare its performance against that of a "Bayes optimal classifier" that knows in advance exactly which features are relevant. So, for some R^*—the "true" set of relevant features—let $p_\Theta(\theta) = \hat{p}_\Theta(\theta|R^*)$. How well does Voting Gibbs using the misspecified prior $\hat{p}$ do?

Theorem 2 *Let any m_0, N and $0 < \epsilon < 1$ be fixed, and assume the training and testing distributions D and D' are the same. Also let R^* be fixed, and let r^* be the number of relevant features ($r^* = \sum_i R_i^*$). Let a training set S of size m be given, where m is distributed uniformly in $\{\lceil(1-\epsilon)m_0\rceil, \lceil(1-\epsilon)m_0\rceil + 1, \ldots, m_0\}$. Then*

$$
\mathrm{E}[\varepsilon(h_{VG(N)})] \leq \left(1 + O\left(\frac{1}{\sqrt{N}}\right)\right)\left(\mathrm{E}[\varepsilon(h_B)] + O\left(\sqrt{\frac{r^*\log f}{m_0}}\right)\right) \quad (8)
$$

where the expectations are over the random training set.

Corollary 3 *To ensure that $\mathrm{E}[\varepsilon(h_{VG(N)})]$ is at most some constant $\epsilon_0 > 0$ more than $(1 + O(1/\sqrt{N}))\mathrm{E}[\varepsilon(h_B)]$, it suffices to choose $m_0 = \Omega(r^* \log f)$.*

Remark (random training set size). The theorem contains a minor technical assumption that m have some small amount of randomization around m_0. This is a condition that treats the training set size as random (usually not an unrealistic assumption), and is needed in the proof of theorem. (See the full paper (Ng & Jordan, 2001) for details.)

Note that to state the simplest possible result, we have given the theorem only in terms of a $1/\sqrt{N}$ convergence rate. Note also that that by letting $N = \infty$, this result also gives a bound for the setting of exact Bayesian inference using misspecified priors.

The corollary, which re-states the error bound in the Theorem in terms of a sample complexity result, shows that if the (approximate) training set size m_0 is $\Omega(r^* \log f)$ (and if N is not unreasonably small), then we will do nearly as well as if we had known exactly which features are relevant. This is the sample complexity of Bayesian feature selection, and since it is only *logarithmic* in f, the total number of features, it means that Bayesian feature selection using the particular prior described earlier is very insensitive to the presence of irrelevant features. This result also recovers the best known such rates (Littlestone, 1988; Kivinen & Warmuth, 1994; Ng, 1998), and has sample complexity that beats that of the common "wrapper" model (Kohavi & John, 1997) feature selection algorithm (see the analysis in Ng, 1998). Indeed, the logarithmic dependence suggests that we can, for instance, *square* the total number of features, and need only *twice* as much training data as a result. Alternatively, we can also view this as saying that Bayesian feature selection can handle *exponentially* many irrelevant features as we have training examples. We believe this result has important implications for feature design in practical supervised learning tasks.

Theorem 2 is proved by showing a more general result (the proof of which is deferred to the full paper, Ng & Jordan, 2001) that is given in terms of $\mathrm{KL}(p_\Theta||\hat{p}_\Theta)$. More specifically, if p_Θ were the "correct" prior used by h_B and $\hat{p}_\Theta$ the misspecified prior used by $h_{VG(N)}$, then under the conditions given in the Theorem above, we have

$$\mathrm{E}[\varepsilon(h_{VG(N)})] \leq \left(1 + O\left(\frac{1}{\sqrt{N}}\right)\right) \cdot \left(\mathrm{E}[\varepsilon(h_B)] + O\left(\sqrt{\frac{\mathrm{KL}(p_\Theta||\hat{p}_\Theta)}{m_0}}\right)\right). \quad (9)$$

These results can also be stated in terms of worst-case error bounds for online learning and indeed such bounds were the inspiration for the theorem. For a closely related result in the worst-case setting for exact Bayesian inference (corresponding to $N = \infty$), see (Barron et al., 1993).

Proof of Theorem 2. The result is easily shown using Equation (9), by observing that

$$\begin{aligned} \hat{p}_\Theta(\theta) &= \sum_R \hat{p}(R)\hat{p}_\Theta(\theta|R) && (10)\\ &\geq \hat{p}(R^*)\hat{p}_\Theta(\theta|R^*) && (11)\\ &= \frac{1}{(f+1)\binom{f}{r^*}}\hat{p}_\Theta(\theta|R^*) && (12)\\ &= \frac{1}{(f+1)\binom{f}{r^*}}p_\Theta(\theta). && (13)\end{aligned}$$

This implies that

$$\begin{aligned} \mathrm{KL}(p_\Theta||\hat{p}_\Theta) &= \int_\theta p_\Theta(\theta)\log\frac{p_\Theta(\theta)}{\hat{p}_\Theta(\theta)}d\theta && (14)\\ &\leq \int_\theta p_\Theta(\theta)\log\left((f+1)\binom{f}{r^*}\right)d\theta && (15)\\ &= \log\left((f+1)\binom{f}{r^*}\right) && (16)\\ &\leq (r^*+1)\log(f+1), && (17)\end{aligned}$$

which when substituted back into Equation (9), gives the theorem. □

It is also interesting to note that if we had used a more "naive" choice of prior, for instance if we have a prior which posits that each feature independently has a some fixed probability of being relevant (so R is a sequence of f independent coin tosses), then an argument similar to the one above would give $\mathrm{KL}(p_\Theta||\hat{p}_\Theta) = O(f)$. This gives an upper-bound on the sample complexity of feature selection of $O(f)$, which is vastly inferior to $O(r^* \log f)$ when $r^* \ll f$. Our experiments in the next section will also empirically compare these two types of priors for feature selection.

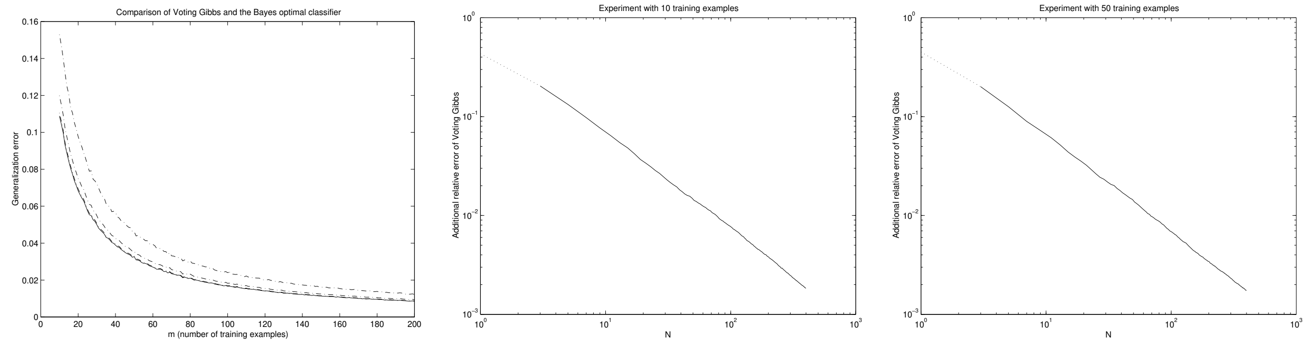
5. Experiments

5.1 The case of correct priors

Our first experiment compares VG(N) and the Bayes optimal classifier in a simple setting that was chosen so that exact Bayesian inference is feasible, which allows repeated comparison between the two methods. Consider a parameter θ uniformly distributed in $[0, 1]$, and let the target output on input x (also uniformly distributed in $[0, 1]$) be 1 if $x \geq \theta$, and 0 otherwise. Using correct priors $\hat{p}_\Theta = p_\Theta$, each classifier (both Bayes optimal and VG) was trained with m training examples $\{(x_i, y_i)\}$ with noisy labels that were corrupted at the (known) noise rate of 0.2, so that $y_i = 1$ with probability 0.8 when $x_i \geq \theta$, and $y_i = 1$ with probability 0.2 when $x_i < \theta$. On each trial, both classifiers observed exactly the same data sample.

Figure 2(a) presents a plot of the generalization errors of the Bayes optimal classifier and of Voting Gibbs with $N = 1, 7$ and 51, plotted as a function of training set size. VG(1)'s error seems somewhat larger than the Bayes optimal classifier's, VG(7) appears to be tracking it quite well, and VG(51)'s performance is virtually indistinguishable from that of the Bayes optimal classifier.

Since our bounds are on the additional relative error, we also plot the additional error as a function of N, for a training set of size $m = 10$. (See Figure 2(b).) As expected, the additional relative error does decrease quite rapidly with N. If the additional relative error decays as $O(1/N)$, then we would also expect on the log-log scale of the plot to see a line with slope approximately -1. Ignoring the single point corresponding to $N = 1$ (see caption), Figure 2(b) seems to almost ex-



(a) (b) (c)

Figure 2. (a) Plot of error vs. number of training examples m for the Bayes optimal classifier (solid) and for VG(1), VG(7) and VG(51) (dash-dot, with higher N corresponding to lower curves). The curve for VG(51) almost completely overlaps that of the Bayes optimal classifier. (The results reported here are averages over 5000 trials.) (b) Plot of additional relative error for VG(N) as a function of (odd) N, for $m = 10$. The dotted part of the line corresponds to only one point on the graph that had $N = 1$. If we ignore this "very small sample" case, the slope of the rest of the line is approximately -1. (c) Same as the previous figure, but with $m = 50$.

actly match the asymptotic slope predicted by our theory. Repeating this with a training set size of $m = 50$, Figure 2(c) shows nearly identical behavior in which the additional relative error also decays as $O(1/N)$.

5.2 Feature selection: The case of misspecified priors

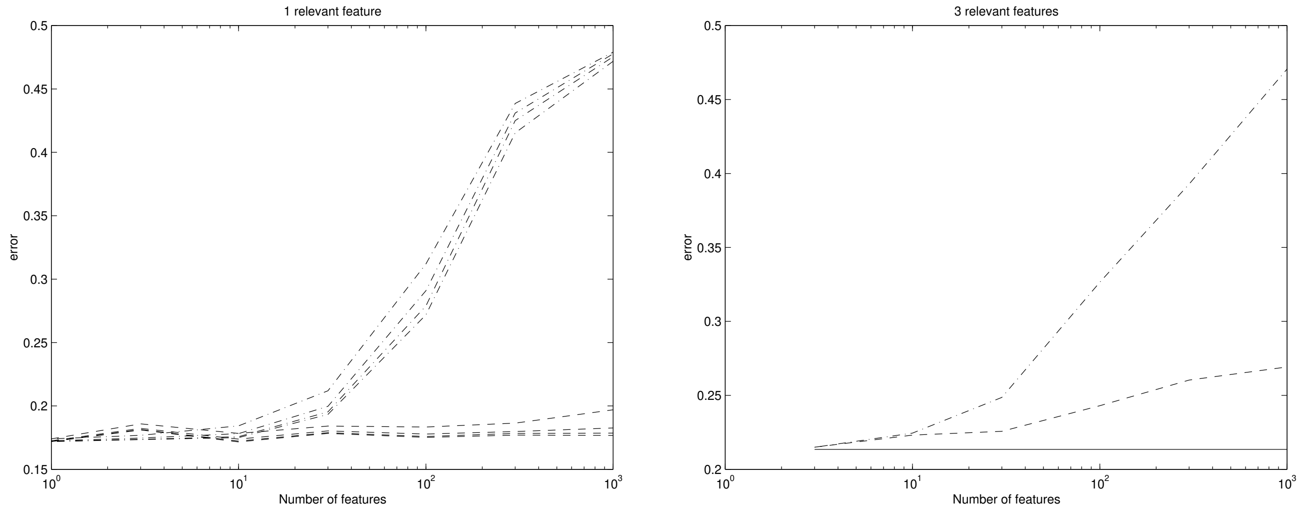
Our second set of experiments studied feature selection. Our learning problem was Bayesian logistic regression (as described in Section 4), and the Bayes optimal classifier which serves as our baseline knows exactly which r^* of the f features are relevant. We tested VG(N) using the "good" prior and the "naive" prior (which posits that R is a sequence of independent coin tosses) described in the previous section. For our experiments, we used 100 training and 10000 test examples, and reversible jump Markov chain Monte Carlo (Green, 1995) to draw N classifiers for VG(N). We let the total number of features vary and let just a single feature be relevant. Our results using $N = 15$ are shown in Figure 1. The results shown are averages of 50 independent trials. The solid line near the bottom shows the error of h_B, which knows exactly which feature was relevant. The dashed line shows VG(15) using the "good" prior, and the dash-dot line VG(15) using the "naive" prior.[3] The results are dramatically different: As predicted by theory, the "good" prior is very insensitive to the presence of large numbers of irrelevant features, and does only slightly worse than if we had been told exactly which features were relevant. In contrast, as the number of irrelevant features becomes large, the error using the "naive" prior approaches that of random guessing (0.5). Note also the scale of the x-axis—even when learning with only 100 training examples and 1000 features (999 of which are irrelevant), the algorithm still performs well.

Figure 3(a) presents the results of an extended experiment in which the errors of VG(N) where assessed, for $N = 1, 3, 7, 15$ and with both priors. In all cases, the lower lines correspond to larger values of N. We see that even with the smaller values of N, performance is still quite reasonable. Finally, Figure 3(b) shows the results when there are 3 relevant features. Once again, we see the "good" prior exhibits a very high tolerance to the presence of irrelevant features.

6. Summary

We have shown that, under mild assumptions, the relative error of Voting Gibbs converges to Bayes optimal performance at a rate of $O(1/N)$. When it is tractable to sample from the posterior distribution of the parameters, this indicates that Voting Gibbs can indeed

[3]Other experimental details: Inputs were drawn from a multivariate standard Normal distribution. For 1 relevant feature, we used for the priors $\sigma_1 = 5$, and $\beta \sim \text{Normal}(0, \sigma_2^2)$, where $\sigma_2 = 0.5$. (σ_1 was defined in Section 4.) For 3 relevant features, σ_1 was also rescaled to 5/3. In the "naive" prior, each feature was assumed to be equally likely to be relevant or irrelevant. Since exact Bayesian inference is not tractable, a long MCMC sequence (run using the "correct" prior $\hat{p}$) was used to approximate both h_B and the ground-truth posterior distributions. Lastly, these experiments were run using the alternative version of VG(N) described in Section 2.2, that skips the second stage of sampling (involving drawing y^i's from Bernoulli($f_{\theta^i}(x)$)), and predicts 1 whenever $(1/N) \sum f_{\theta^i}(x) \geq 0.5$.



(a) (b)

Figure 3. (a) Plot of errors of VG(1), VG(3), VG(7), VG(15) using the "good" (dash) and "naive" (dash-dot) priors. Higher lines correspond to lower values of N. (b) Same as Figure 1, but with 3 instead of 1 relevant features.

provide a good, practical way to approximate optimal Bayesian classification. In the context of feature selection, we also showed that Voting Gibbs has very high tolerance to the presence of irrelevant features, with bounds comparable to those of the best known feature selection algorithms.

Acknowledgements

We thank Nando de Freitas, Vassilis Papavassiliou and Hanna Pasula for helpful conversations about this work. This work was supported by ONR MURI N00014-00-1-0637 and NSF grant IIS-9988642.

References

Barron, A., Clarke, B., & Haussler, D. (1993). Information bounds for the risk of Bayesian predictions and the redundancy of Universal codes. *Proceedings of the International Symposium on Information Theory.*

Denison, D., & Mallick, B. (2000). Classification trees. In D. Dey, S. Ghosh and B. Mallick (Eds.), *Generalized Linear Models: A Bayesian perspective*, 365–372. Marcel-Dekker.

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (Eds.). (1996). *Markov Chain Monte Carlo in Practice.* Chapman and Hall.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, *82*, 711–732.

Kearns, M., & Vazirani, U. V. (1994). *An introduction to computational learning theory.* MIT Press.

Kivinen, J., & Warmuth, M. K. (1994). *Exponentiated gradient versus gradient descent for linear predictors* (Technical Report UCSC-CRL-94-16). Univ. of California Santa Cruz, Computer Research Laboratory.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, *97*, 273–324.

Littlestone, N. (1988). Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, *2*, 285–318.

Ng, A. Y. (1998). On Feature Selection: Learning with exponentially many irrelevant features as training examples. *Proceedings of the Fifteenth International Conference on Machine Learning* (pp. 404–412). Morgan Kaufmann.

Ng, A. Y., & Jordan, M. I. (2001). Convergence rates of the Voting Gibbs classifier, with application to Bayesian feature selection. www.cs.berkeley.edu/˜ang/papers/icml01-vg-long.ps.

Ripley, B. D. (1987). *Stochastic simulation.* John Wiley.

Sykacek, P. (2000). On input selection with reversible jump Markov chain Monte Carlo sampling. *Advances in Neural Information Processing Systems 12* (pp. 638–644).

Valiant, L. G. (1984). A theory of the learnable. *Communications of the ACM*, *25*, 1134–1142.