An Empirical Study of the Naïve Bayes Classifier

Article · January 2001

CITATIONS
738

READS
12,120

1 author:

Irina Rish
IBM
136 PUBLICATIONS **2,802** CITATIONS

SEE PROFILE

An empirical study of the naive Bayes classifier

I. Rish*
T.J. Watson Research Center
rish@us.ibm.com

Abstract

The naive Bayes classifier greatly simplify learning by assuming that features are independent given class. Although independence is generally a poor assumption, in practice naive Bayes often competes well with more sophisticated classifiers.

Our broad goal is to understand the data characteristics which affect the performance of naive Bayes. Our approach uses Monte Carlo simulations that allow a systematic study of classification accuracy for several classes of randomly generated problems. We analyze the impact of the distribution entropy on the classification error, showing that low-entropy feature distributions yield good performance of naive Bayes. We also demonstrate that naive Bayes works well for certain nearly-functional feature dependencies, thus reaching its best performance in two opposite cases: completely independent features (as expected) and functionally dependent features (which is surprising). Another surprising result is that the accuracy of naive Bayes is not directly correlated with the degree of feature dependencies measured as the class-conditional mutual information between the features. Instead, a better predictor of naive Bayes accuracy is the amount of information about the class that is lost because of the independence assumption.

1 Introduction

Bayesian classifiers assign the most likely class to a given example described by its feature vector. Learning such classifiers can be greatly simplified by assuming that features are independent given class, that is, $P(\mathbf{X}|C) = \prod_{i=1}^{n} P(X_i|C)$, where $\mathbf{X} = (X_1, \cdots, X_n)$ is a feature vector and C is a class. Despite this unrealistic assumption, the resulting classifier known as *naive Bayes* is remarkably successful in practice, often competing with much more sophisticated techniques [6; 8; 4; 2]. Naive Bayes has proven effective in many practical applications, including text classification, medical diagnosis, and systems performance management [2; 9; 5].

*T.J. Watson Research Center, 30 Saw Mill River Road, Hawthorne, NY 10532. Phone +1 (914) 784-7431

The success of naive Bayes in the presence of feature dependencies can be explained as follows: optimality in terms of zero-one loss (classification error) is not necessarily related to the quality of the fit to a probability distribution (i.e., the appropriateness of the independence assumption). Rather, an optimal classifier is obtained as long as both the actual and estimated distributions agree on the most-probable class [2]. For example, [2] prove naive Bayes optimality for some problems classes that have a high degree of feature dependencies, such as disjunctive and conjunctive concepts.

However, this explanation is too general and therefore not very informative. Ultimately, we would like to understand the data characteristics which affect the performance of naive Bayes. While most of the work on naive Bayes compares its performance to other classifiers on particular benchmark problems (e.g., UCI benchmarks), our approach uses Monte Carlo simulations that allow a more systematic study of classification accuracy on parametric families of randomly generated problems. Also, our current analysis is focused only on the *bias* of naive Bayes classifier, not on its *variance*. Namely, we assume an infinite amount of data (i.e., a perfect knowledge of data distribution) which allows us to separate the approximation error (bias) of naive Bayes from the error induced by training sample set size (variance).

We analyze the impact of the distribution entropy on the classification error, showing that certain almost-deterministic, or low-entropy, dependencies yield good performance of naive Bayes. (Note that almost-deterministic dependencies are a common characteristic in many practical problem domains, such as, for example, computer system management and error-correcting codes.) We show that the error of naive Bayes vanishes as the entropy $H(P(\mathbf{X}|C))$ approaches zero. Another class of almost-deterministic dependencies generalizes functional dependencies between the features. Particularly, we show that naive Bayes works best in two cases: completely independent features (as expected) and functionally dependent features (which is less obvious), while reaching its worst performance between these extremes.

We also show that, surprisingly, the accuracy of naive Bayes is not directly correlated with the degree of feature dependencies measured as the class-conditional mutual information between the features, $I(X_i; X_j|C)$ (X_i and X_i are features and C is the class). Instead, our experiments reveal that a better predictor of naive Bayes accuracy can be

the loss of information that features contain about the class when assuming naive Bayes model, namely $I(C;(X_i,X_j))-I_{NB}(C;(X_i,X_j))$, where I_{NB} is the mutual information between features and class under naive Bayes assumption.

This paper is structured as follows. In the next section we provide necessary background and definitions. Section 3 discusses naive Bayes performance for nearly-deterministic dependencies, while Section 4 demonstrates that the "information loss" criterion can be a better error predictor than the strength of feature dependencies. A summary and conclusions are given in Section 5.

2 Definitions and Background

Let $\mathbf{X}=(X_1,...,X_n)$ be a vector of observed random variables, called *features*, where each feature takes values from its *domain* D_i. The set of all feature vectors (*examples*, or *states*), is denoted $\Omega = D_1 \times ... \times D_n$. Let C be an unobserved random variable denoting the *class* of an example, where C can take one of m values $c \in \{0,...,m-1\}$. Capital letters, such as X_i, will denote variables, while lower-case letters, such as x_i, will denote their values; boldface letters will denote vectors.

A function $g: \Omega \rightarrow \{0,...,m-1\}$, where $g(\mathbf{x})=C$, denotes a *concept* to be learned. Deterministic $g(x)$ corresponds to a concept without noise, which always assigns the same class to a given example (e.g., disjunctive and conjunctive concepts are deterministic). In general, however, a concept can be *noisy*, yielding a random function $g(x)$.

A classifier is defined by a (deterministic) function $h: \Omega \rightarrow \{0,...,m-1\}$ (a *hypothesis*) that assigns a class to any given example. A common approach is to associate each class i with a discriminant function $f_i(\mathbf{x})$, $i=0,...,m-1$, and let the classifier select the class with maximum discriminant function on a given example: $h(\mathbf{x}) = \arg\max_{i\in\{0,...,m-1\}} f_i(\mathbf{x})$.

The *Bayes* classifier $h^*(\mathbf{x})$ (that we also call *Bayes-optimal* classifier and denote $BO(\mathbf{x})$), uses as discriminant functions the class posterior probabilities given a feature vector, i.e. $f_i^*(\mathbf{x}) = P(C=i|\mathbf{X}=\mathbf{x})$. Applying Bayes rule gives $P(C=i|\mathbf{X}=\mathbf{x}) = \frac{P(\mathbf{X}=\mathbf{x}|C=i)P(C=i)}{P(\mathbf{X}=\mathbf{x})}$, where $P(\mathbf{X}=\mathbf{x})$ is identical for all classes, and therefore can be ignored. This yields Bayes discriminant functions

$$f_i^*(\mathbf{x}) = P(\mathbf{X}=\mathbf{x}|C=i)P(C=i), \qquad (1)$$

where $P(\mathbf{X}=\mathbf{x}|C=i)$ is called the *class-conditional probability distribution (CPD)*. Thus, the Bayes classifier

$$h^*(\mathbf{x}) = \arg\max_i P(\mathbf{X}=\mathbf{x}|C=i)P(C=i) \qquad (2)$$

finds the *maximum a posteriori probability* (MAP) hypothesis given example $\mathbf{x}$. However, direct estimation of $P(\mathbf{X}=\mathbf{x}|C=i)$ from a given set of training examples is hard when the feature space is high-dimensional. Therefore, approximations are commonly used, such as using the simplifying assumption that features are independent given the class. This yields the *naive Bayes* classifier $NB(\mathbf{x})$ defined by discriminant functions

$$f_i^{NB}(\mathbf{x}) = \Pi_{j=1}^n P(X_j=x_j|C=i)P(C=i). \qquad (3)$$

The probability of a classification error, or $risk$ of a classifier h is defined as

$$R(h) = P(h(\mathbf{X}) \neq g(\mathbf{X})) =$$
$$= \sum_{\mathbf{x}\in\Omega} P(h(\mathbf{x}) \neq g(\mathbf{x}))P(\mathbf{X}=\mathbf{x}) = E_{\mathbf{x}}\{P(h(\mathbf{x}) \neq g(\mathbf{x}))\},$$

where $E_{\mathbf{x}}$ is the expectation over $\mathbf{x}$. $R^*=R(h^*)$ denotes the Bayes error (Bayes risk).

We say that classifier h is *optimal* on a given problem if its risk coincides with the Bayes risk. Assuming there is no noise (i.e. zero Bayes risk), a concept is called *separable* by a set of functions $S=\{f_c(x)|c=0,...,m-1\}$ if every example $\mathbf{x}$ is classified correctly when using each $f_c(x)$ as discriminant functions.

As a measure of dependence between two features X_k and X_j we use the class-conditional mutual information [1], which can be defined as

$$I(X_k;X_j|C) = H(X_k|C) + H(X_j|C) - H(X_k,X_j|C),$$

where $H(A|C)$ is the class-conditional entropy of A, defined as:

$$-\sum_i P(C=i)\sum_t P(A=t|C=i)\log P(A=t|C=i).$$

Mutual information is zero when X_k and X_j are mutually independent given class C, and increases with increasing level of dependence, reaching the maximum when one feature is a deterministic function of the other.

3 When does naive Bayes work well? Effects of some nearly-deterministic dependencies

In this section, we discuss known limitations of naive Bayes and then some conditions of its optimality and near-optimality, that include low-entropy feature distributions and nearly-functional feature dependencies.

3.1 Concepts without noise

We focus first on concepts with $P(C=i|\mathbf{x})=0$ or 1 for any $\mathbf{x}$ and i (i.e. no noise), which therefore have zero Bayes risk. The features are assumed to have finite domains (i-th feature has k_i values), and are often called *nominal*. (A nominal feature can be transformed into a numeric one by imposing an order on its domain.) Our attention will be restricted to binary classification problems where the class is either 0 or 1.

Some limitations of naive Bayes are well-known: in case of binary features ($k_i=2$ for all $i=1,...,n$), it can only learn linear discriminant functions [3], and thus it is always suboptimal for non-linearly separable concepts (the classical example is XOR function; another one is m-of-n concepts [7; 2]). When $k_i>2$ for some features, naive Bayes is able to learn (some) polynomial discriminant functions [3]; thus, polynomial separability is a necessary, although not sufficient, condition of naive Bayes optimality for concepts with finite-domain features.

Despite its limitations, naive Bayes was shown to be optimal for some important classes of concepts that have a high degree of feature dependencies, such as disjunctive and conjunctive concepts [2]. These results can be generalized to concepts with any nominal features (see [10] for details):

Theorem 1 *[10] The naive Bayes classifier is optimal for any two-class concept with nominal features that assigns class 0 to exactly one example, and class 1 to the other examples, with probability 1.* [1]

The performance of naive Bayes degrades with increasing number of class-0 examples (i.e., with increasing prior $P(C = 0)$, also denoted $P(0)$), as demonstrated in Figure 1a. This figure plots average naive Bayes error computed over 1000 problem instances generated randomly for each value of $P(C = 0)$. The problem generator, called **ZeroBayesRisk**, assumes m features (here we only consider two features), each having k values, and varies the number l of class-0 examples from 1 to $k^2/2$ (so that $P(C = 0)$ varies from $1/N$ to 0.5; the results for $P(C = 0) > 0.5$ are symmetric)[2]. As expected, larger $P(C = 0)$ (equivalently, larger l), yield a wider range of problems with various dependencies among features, which result into increased errors of Bayes; a closer look at the data shows no other cases of optimality besides $P(C = 0) = 1/N$.

Surprisingly, the strength of inter-feature dependencies, measured as the class-conditional mutual information $I(X_1; X_2|C)$ (also denoted MI), is not a good predictor of naive Bayes performance: while average naive Bayes error increases monotonically with $P(0)$, the mutual information is non-monotone, reaching its maximum around $P(0) = 0.1$. This observation is consistent with previous empirical results on UCI benchmarks [2]) that also revealed low correlation between the degree of feature dependence and relative performance of naive Bayes with respect to other classifiers, such as C4.5, CN2, and PEBLS.

It turns out that the entropy of class-conditional marginal distributions, $P(X_i|C)$, is a better predictor of naive Bayes performance. Intuitively, low entropy of $P(X_i|0)$ means that most of 0s are "concentrated around " one state (in the limit, this yields the optimality condition stated by Theorem 1). Indeed, plotting average $H(P(X_1|0)$ in Figure 1a demonstrates that both average error and average entropy increase monotonically in $P(0)$. Further discussion of low-entropy distributions is given next in the more general context of noisy (non-zero Bayes risk) classification problems.

3.2 Noisy concepts

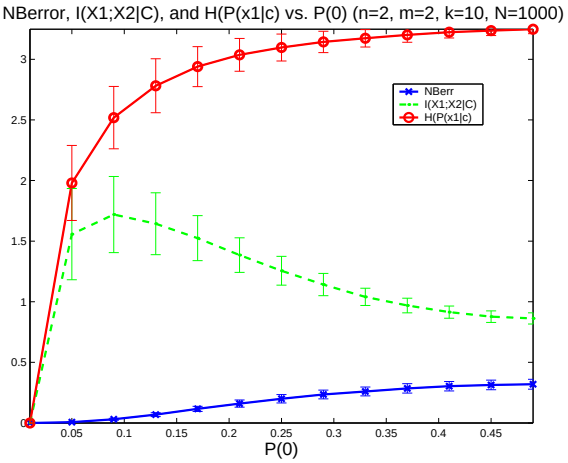
Low-entropy feature distributions

Generally, concepts can be noisy, i.e. can have non-deterministic $P(C = i|\mathbf{x})$ and thus a non-zero Bayes risk. A natural extension of the conditions of Theorem 1 to noisy concepts yields low-entropy, or "extreme", probability distributions, having almost all the probability mass concentrated in one state. Indeed, as shown in [10], the independence assumption becomes more accurate with decreasing entropy which yields an asymptotically optimal performance of naive Bayes. Namely,

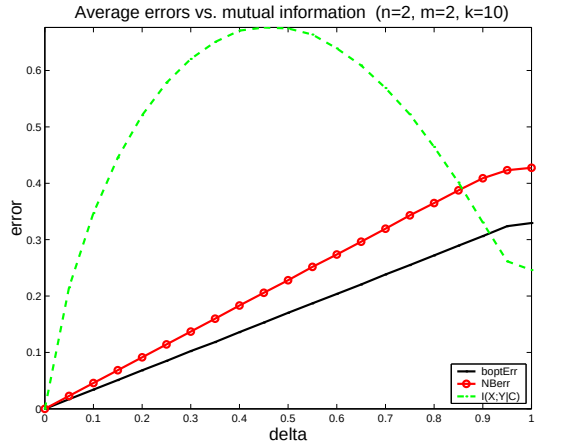
Theorem 2 *[10] Given that one of the following conditions hold:*

[1] Clearly, this also holds in case of a single example of class 1.

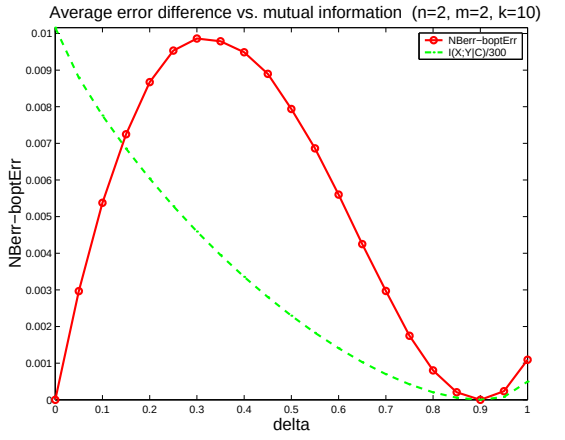
[2] Note that in all experiments perfect knowledge of data distribution (i.e., infinite amount of data)is assumed in order to avoid the effect of finite sample size.



(a)



(b)



(c)

Figure 1: (a) results for the generator **ZeroBayesRisk** (k=10, 1000 instances): average naive Bayes error (NBerr), class-conditional mutual information between features ($I(X1; X2|C)$), and entropy of marginal distribution, $H(P(X_1|0))$; the error bars correspond to the standard deviation of each measurement across 1000 problem instances; (b) Results for the generator **EXTREME**: average Bayes and naive Bayes errors and average $I(X1; X2|C)$; (c) results for the generator **FUNC1**: average difference between naive Bayes error and Bayes error (≈ 0.336 - constant for all δ), and scaled I(X1;X2—C) (divided by 300).

1. *a joint probability distribution* $P(X_1, ..., X_n)$ *is such that* $P(x_1^*, ..., x_n^*) \geq 1 - \delta$ *for some state* $\mathbf{x}^* = (x_1^*, ..., x_n^*)$, *or*
2. *a set of marginal probability distributions* $P(X_1), ..., P(X_n)$ *is such that for each* i, $P(X_i = x_i^*) \geq 1 - \delta$ *for some* x_i^*,

then $|P(x_1, ..., x_n) - \prod_{i=1}^{n} P(X_i = x_i)| \leq n\delta$.

The performance of naive Bayes on low-entropy distributions is demonstrated using a random problem generator called **EXTREME**. This generator takes the number of classes, m, number of features, n, number of values per feature, k, and the parameter δ, and creates m class-conditional feature distributions, each satisfying the condition $P(\mathbf{x}|C = c) = 1 - \delta$ if $\mathbf{x} = \mathbf{x}^c$, where the $\mathbf{x}^c$ are m different states randomly selected from k^n possible states. For each class i, the remaining probability mass δ in $P(\mathbf{x}|C = i)$ is randomly distributed among the remaining $k^n - 1$ states. Class prior distributions are uniform. Once $P(\mathbf{X}|C)$ is generated, naive Bayes classifier (NB) is compared against the Bayes-optimal classifier (BO).

Figure 1b shows that, as expected, the naive Bayes error (both the average and the maximum) converges to zero with $\delta \to 0$ (simulation performed on a set of 500 problems with $n = 2$, $m = 2$, $k = 10$). Note that, similarly to the previous observations, the error of naive Bayes is not a monotone function of the strength of feature dependencies; namely, the average class-conditional mutual information plotted in Figure 1b is a concave function reaching its maximum between $\delta = 0.45$ and $\delta = 0.5$, while the decrease of average naive Bayes error is monotone in δ.

Almost-functional feature dependencies

Another "counterintuitive" example that demonstrates the non-monotonic relation between the feature dependence and the naive Bayes accuracy is the case of certain functional and nearly-functional dependencies among features. Formally,

Theorem 3 *[10] Given equal class priors, Naive Bayes is optimal if* $X_i = f_i(X_1)$ *for every feature* X_i, $i = 2, ..., n$, *where* $f_i(\cdot)$ *is a one-to-one mapping* [3].

Namely, naive Bayes can be optimal in situations just opposite to the class-conditional feature independence (when mutual information is at minimum) - namely, in cases of completely deterministic dependence among the features (when mutual information achieves its maximum). For example, Figure 1c plots the simulations results obtained using an "nearly-functional" feature distribution generator called **FUNC1**, which assumes uniform class priors, two features, each having k values, and "relaxes" functional dependencies between the features using the noise parameter δ. Namely, this generator selects a random permutation of k numbers, which corresponds to a one-to-one function f that binds the two features: $X_2 = f(X_1)$ $(1 - \delta)$. Then it generates randomly two class-conditional (marginal) distributions for the X_1 feature, $P_0(X_1)$ and $P_1(X_1)$, for class 0 and class 1, respectively. Finally, it creates class-conditional joint feature distributions satisfying the following conditions:

$$P_c(x_1, x_2 = f(x_1)) = P_c(x_1)(1 - \delta), \text{and}$$
$$P_c(x_1, x_2 \neq f(x_1)) = P_c(x_1)\frac{\delta}{k-1}, c = 0, 1.$$

This way the states satisfying functional dependence obtain $1 - \delta$ probability mass, so that by controlling δ we can get as close as we want to the functional dependence described before, i.e. the generator relaxes the conditions of Theorem 3. Note that, on the other hand, $\delta = \frac{k-1}{k}$ gives us uniform distributions over the second feature $P_c(x_2) = \sum_{x_1} P_c(x_1, x_2) = \frac{1}{k}$, which makes it independent of X_1 (given class c). Thus varying δ from 0 to 1 explores the whole range from deterministic dependence to complete independence between the features given class.

The results for 500 problems with $k = 10$ are summarized in Figure 1c, which plots the difference between the average naive Bayes error and average Bayes risk (which turned out to be ≈ 0.336, a constant for all δ) is plotted against δ. We can see that naive Bayes is optimal when $\delta = 0$ (functional dependence) and when $\delta = 0.9$ (complete independence), while its maximum error is reached between the two extremes. On the other hand, the class-conditional mutual information decreases monotonically in δ, from its maximum at δ (functional dependencies) to its minimum at $\delta = 0.9$ (complete independence)[4].

4 Information loss: a better error predictor than feature dependencies?

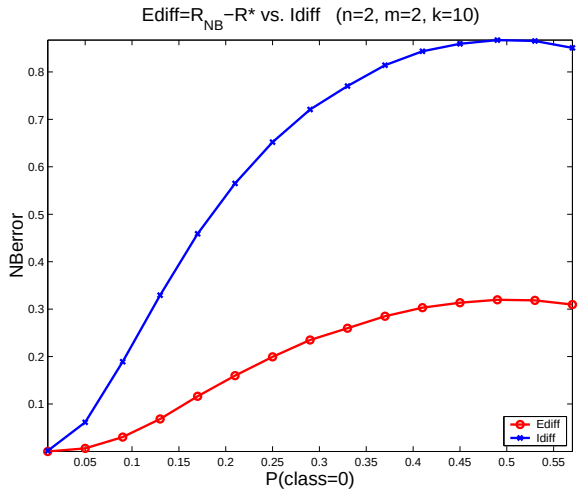
As we observed before, the strength of feature dependencies (i.e. the class-conditional mutual information between the features) 'ignored' by naive Bayes is not a good predictor of its classification error. This makes us look for a better parameter that estimates the impact of independence assumption on classification.

We start with a basic question: which dependencies between features can be ignored when solving a classification task? Clearly, the dependencies which do not help distinguishing between different classes, i.e. do not provide any information about the class. Formally, let $I(C; (X_1, X_2))$ be the mutual information between the features and the class (note the difference from class-conditional mutual information) given the "true" distribution $P(X_1, X_2, C)$, while $I_{NB}(C; (X_1, X_2))$ is the same quantity computed for $P_{NB}(X_1, X_2, C) = P(X1|C)P(X2|C)P(C)$, the naive Bayes approximation of $P(X_1, X_2, C)$. Then the parameter $Idiff = I(C; (X_1, X_2)) - I_{NB}(C; (X_1, X_2))$ measures the amount of information about the class which is "lost" due to naive Bayes assumption. Figure 2a shows that average $Idiff$ ("information loss") increases monotonically with $P(0)$, just as the average error of naive Bayes. More interestingly, Figure 2b plots average naive Bayes error versus average $Idiff$ for three different values of k ($k = 5, 10, 15$), which all yield

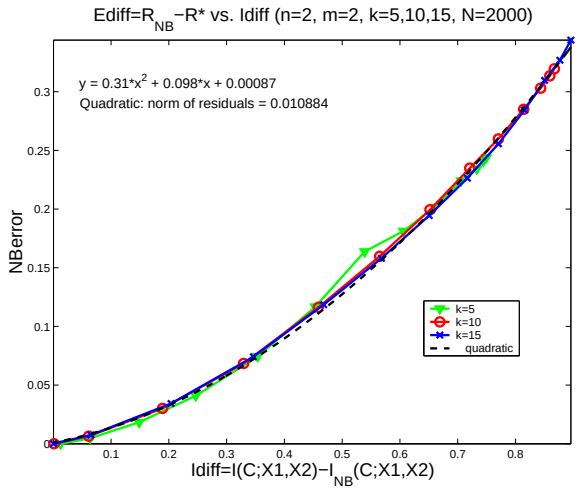
[3] A similar observation was made in [11], but the important "one-to-one" condition on functional dependencies was not mentioned there. However, it easy to construct an example of a non-one-to-one functional dependence between the features that yields non-zero error of naive Bayes.

[4] Note that the mutual information in Figure 1c is scaled (divided by 300) to fit the error range.

almost same curve, closely approximated by a quadratic function $y = 0.3x^2 + 0.1x + 0.001$. Our results, not shown here due to space restrictions, also demonstrate that variance of the error increases with $Idiff$ for each fixed k; however, maximum variance decreases with k. While the dependence between the error and the information loss requires further study, it is clear that the for zero-Bayes-risk problems information loss is a much better predictor of the error than the mutual dependence between the features (compare to Figure 1a).



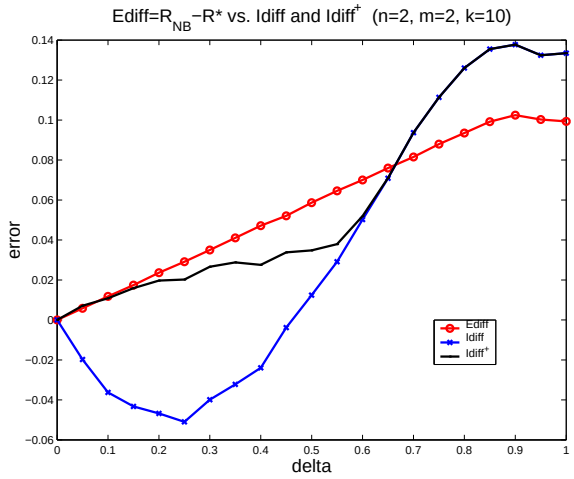
(a)



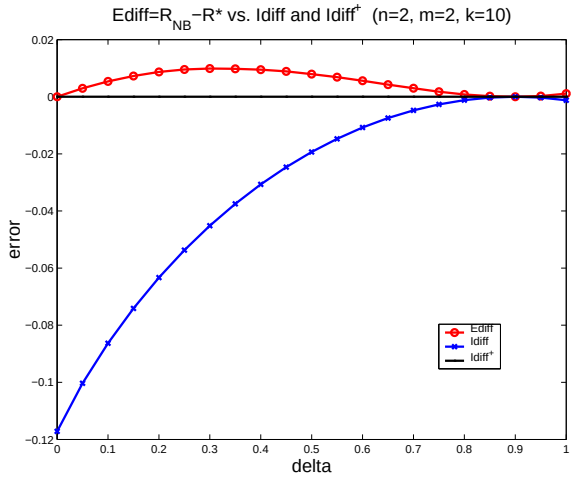
(b)

Figure 2: Results for generator **ZeroBayesRisk** (13 values of P(0) in $[0, 0.5]$ range, 2000 instances per each value of $P(0)$): (a) Average naive Bayes error and average information loss $Idiff$ versus $P(0)$; (b) Average naive Bayes error versus average "information loss" $Idiff$ for k=5,10, and 15.

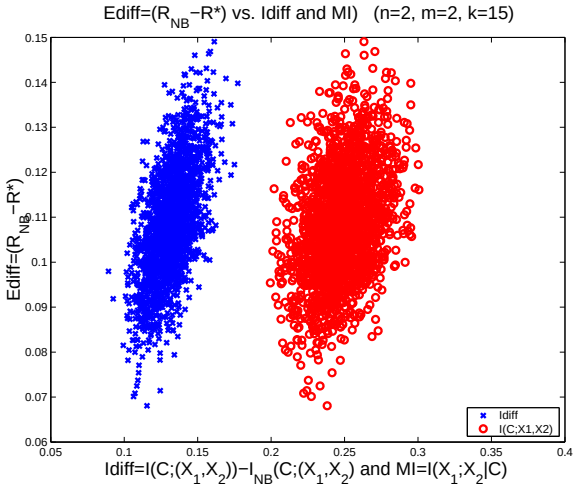
For non-zero Bayes risk, the picture is somewhat less clear. However, the information loss still seems to be a better error predictor than the class-conditional mutual information between the features. Figure 3a plots the average difference between naive Bayes error and the Bayes risk, called $Ediff$, and the information loss $Idiff$ versus the parame-



(a)



(b)



(c)

Figure 3: Information loss $Idiff$ on noisy concepts: average error difference between naive Bayes and optimal Bayes, $Ediff$, and average $Idiff$ for (a) generator **EXTREME** and (b) generator **FUNC1**; (c) scatter plot of $Ediff$ versus $Idiff$ and versus mutual information $MI = I(X_1; X_2|C)$ for generator **RANDOM**.

ter δ. At the first sight, it looks like $Idiff$ is non-monotone in δ while $Ediff$ is monotone; particularly, while the error increases with δ, information loss decreases in the interval $0 \leq \delta \leq 0.25$. Note, however, this interval yields $negative$(!) values of $Idiff$. It appears that naive Bayes overestimates the amount of information the features have about the class (possibly, by counting same information twice due to the independence assumption), which results in negative $Idiff$. If we assume that such overestimation is not harmful, just equivalent to not losing any information, and plot instead the average of $max(Idiff, 0)$ (denoted $Idiff^+$), we observe a monotone relationship between the average of $Idiff^+$ and the average naive Bayes error, as one would expect (i.e., both increase monotonically up to $\delta = 0.9$, and then decrease).

Similarly, in Figure 3b we plot the error difference $Ediff$ as well as $Idiff$ and $Idiff^+$ versus δ for our second generator of non-zero Bayes risk problems, **FUNC1**. In this cases, naive Bayes always overestimates the amount of information about the class, thus $Idiff$ is always non-positive, i.e. $Idiff^+ = 0$. Its relation to the naive error Bayes which reaches its maximum at some intermediate value of δ is thus not clear.

Finally, we used a "completely" random problem generator (called **RANDOM**) to compare the class-conditional mutual information between the features, $I(X_1; X_2|C)$, and the information loss $Idiff$, on arbitrary noisy concepts. For each class, this generator samples each $P(X_1 = x_1, X_2 = x_2|C = c)$ from a uniform distribution on the interval [0.0,1.0]; the resulting probability table is then normalized (divided by the total sum over all entries). Figure 3c shows a scatter-plot for $Ediff$, the error difference between naive Bayes and optimal Bayes classifiers, versus feature dependence $I(X_1; X_2|C)$ and versus information loss $Idiff$. In this cases, we can see that both parameters are correlated with the error, however, the variance is quite high, especially for $I(X_1; X_2|C)$. Further study of both parameters on different classes of noisy concepts is needed to gain a better understanding of their relevance to the classification error.

5 Conclusions

Despite its unrealistic independence assumption, the naive Bayes classifier is surprisingly effective in practice since its classification decision may often be correct even if its probability estimates are inaccurate. Although some optimality conditions of naive Bayes have been already identified in the past [2], a deeper understanding of data characteristics that affect the performance of naive Bayes is still required.

Our broad goal is to understand the data characteristics which affect the performance of naive Bayes. Our approach uses Monte Carlo simulations that allow a systematic study of classification accuracy for several classes of randomly generated problems. We analyze the impact of the distribution entropy on the classification error, showing that certain almost-deterministic, or low-entropy, dependencies yield good performance of naive Bayes. Particularly, we demonstrate that naive Bayes works best in two cases: completely independent features (as expected) and functionally dependent features (which is surprising). Naive Bayes has its worst performance between these extremes.

Surprisingly, the accuracy of naive Bayes is not directly correlated with the degree of feature dependencies measured as the class-conditional mutual information between the features. Instead, a better predictor of accuracy is the loss of information that features contain about the class when assuming naive Bayes model. However, further empirical and theoretical study is required to better understand the relation between those information-theoretic metrics and the behavior of naive Bayes. Further directions also include analysis of naive Bayes on practical application that have almost-deterministic dependencies, characterizing other regions of naive Bayes optimality and studying the effect of various data parameters on the naive Bayes error. Finally, a better understanding of the impact of independence assumption on classification can be used to devise better approximation techniques for learning efficient Bayesian net classifiers, and for probabilistic inference, e.g., for finding maximum-likelihood assignments.

Acknowledgements

We wish to thank Mark Brodie, Vittorio Castelli, Joseph Hellerstein, Jayram Thathachar, Daniel Oblinger, and Ricardo Vilalta for many insightful discussions that contributed to the ideas of this paper.

References

[1] T.M. Cover and J.A. Thomas. *Elements of information theory*. New York:John Wiley & Sons, 1991.

[2] P. Domingos and M. Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, 29:103–130, 1997.

[3] R.O. Duda and P.E. Hart. *Pattern classification and scene analysis*. New York: John Wiley and Sons, 1973.

[4] N. Friedman, D. Geiger, and Goldszmidt M. Bayesian network classifiers. *Machine Learning*, 29:131–163, 1997.

[5] J. Hellerstein, Jayram Thathachar, and I. Rish. Recognizing end-user transactions in performance management. In *Proceedings of AAAI-2000*, pages 596–602, Austin, Texas, 2000.

[6] J. Hilden. Statistical diagnosis based on conditional independence does not require it. *Comput. Biol. Med.*, 14(4):429–435, 1984.

[7] R. Kohavi. Wrappers for performance enhancement and oblivious decision graphs. Technical report, PhD thesis, Department of Computer Science, Stanford, CA, 1995.

[8] P. Langley, W. Iba, and K. Thompson. An analysis of Bayesian classifiers. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 399–406, San Jose, CA, 1992. AAAI Press.

[9] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[10] I. Rish, J. Hellerstein, and T. Jayram. An analysis of data characteristics that affect naive Bayes performance. Technical Report RC21993, IBM T.J. Watson Research Center, 2001.

[11] H. Schneiderman and T. Kanade. A statistical method for 3d detection applied to faces and cars. In *Proceedings of CVPR-2000*, 2000.