C5.1.5 Bayesian Classification

Nir Friedman
nirf@cs.huji.ac.il

Computer Science Department
Jerusalem University
Jerusalem
Israel

Ron Kohavi
ronnyk@CS.Stanford.EDU

Data Mining
Blue Martini software
2600 Campus Dr. Suite 175
San Mateo, CA 94403

Abstract

Bayesian classification addresses the classification problem by learning the distribution of instances given different class values. We review the basic notion of Bayesian classification, describe in some detail the *naive Bayesian classifier*, and briefly discuss some extensions.

C5.1.5.1 Introduction

The goal of classification [link to section C5.1.1] is to classify an instance to a *class* based on the value of several *attributes*. Many approaches to classification attempt to explicitly construct a function from the joint set of values of the attributes to class labels. Example of such classifiers include decision trees [link to section C5.1.3], decision rules [link to section C5.1.4] and neural networks [link to section C5.1.4].

Bayesian classification takes a somewhat different approach to this problem. In this approach, we approximate the joint probability distribution of the class and the attributes: $\Pr(C, A_1, \ldots, A_k)$, where C is a random variable describing the class, and $A_1, \ldots A_k$ are random variables describing the attributes. Thus, learning in Bayesian classification amounts to estimation of this joint probability distribution. After we construct such an estimate, we classify a new instances by examining conditional probability of C given the particular attribute values, and returning the class that is most probable.

The standard approach to Bayesian classification uses the *chain rule* to decompose the joint distribution:

$$\Pr(C, A_1, \ldots, A_k) = \Pr(C) \Pr(A_1, \ldots, A_k | C) \tag{1}$$

The first term on the right hand side of (1) is the *prior* probability of the class labels. These can be directly estimated from the training data, or from a larger sample of the population. For example, we can often get statistics on the number of, say, breast cancer occurrences in the general population. The second term on the right-hand side of (1) is the distribution of attribute values given the class label. The estimation of this term is usually more complex, and we elaborate on it below.

Once we have an estimate of $\Pr(C)$ and $\Pr(A_1, \ldots, A_k|C)$ we can use *Bayes rule* to get the conditional probability of the class given the attributes:

$$\Pr(C|A_1, \ldots, A_k) = \alpha \Pr(C) \Pr(A_1, \ldots, A_k | C), \tag{2}$$

where α is a normalization factor that ensures that the conditional probability of all possible class labels sums up to 1. (In practice, we do not need to explicitly evaluate this factor because it is constant for a given instance.) Using (2) we can classify new instances by combining the prior probability of each class with the probability of the given attribute values given that class.

C5.1.5.2 Properties of Bayesian Classifiers

Bayesian classification does not attempt learn an explicit decision rule. Instead, learning reduces to estimating probabilities. A consequence there are some differences with other approaches to classification. In this section, we briefly touch on the main ones.

A basic property that we often require is asymptotic correctness; the classification system should learn the best possible classifier if we provide it with a sufficient number of training instances, ignoring computational limitation.

It can be shown that induction of a Bayesian classifier can be asymptotically optimal (i.e., reaches the smallest possible classification error given a sufficiently large training set) if the method of estimating $\Pr(A_1, \ldots, A_k|C)$ is *consistent*, that is will converge to the true underlying conditional distribution given a sufficiently large sample. Thus, the asymptotic properties depend on our choice of methods for estimating $\Pr(A_1, \ldots, A_k|C)$. Note that in contrast to some learning methods, in Bayesian classification it is possible that the class of hypotheses we consider contains an optimal classifier, and yet we would not learn it even with infinite amount of data. This can happen if the probabilistic model that correspond to this optimal classification rule does not provide the best approximation to the observed probability distribution.

This asymptotic guarantee suggests that if our knowledge about the domain leads us to believe that a particular model (i.e., class of hypotheses) for $\Pr(A_1, \ldots, A_k|C)$ allow for a good approximation of the true distribution, then we would expect the Bayesian classifier to perform well. On the other hand, this does *not* imply that an "unrealistic" model, that does not give good approximation to the distribution, is necessarily a bad classifier. For example, the model used in the naive Bayesian classifier of the next section, makes unrealistic assumptions, yet often leads to competitive classification performance (Domingos & Pazzani 1997).

Probabilistic semantics of Bayesian classification yield the following advantages over other methods.

First, Bayesian classification can be combined with principled methods for dealing with asymmetric loss functions. For example, in cancer screening, a misdiagnosis of a malignant tumor is more costly than a misdiagnosis of a benign tumor, since the detection of cancer in early stage can dramatically improve the chances of curing the cancer. To deal with such situations, we can rely on *decision theory* to provide a principled methods for combine probability estimates with the utility (or cost) of different decisions. See, for example, Duda & Hart (1973) and Bishop (1995).

Second, probabilistic methods provide principled method for dealing with missing values. Probability theory allows us to deal with missing values in classification by averaging over the possible values that the attribute might have taken. For example, if the value of A_1 is not provided, then the probability of $\Pr(A_2, \ldots, A_k|C)$ is $\sum_{x \in DOM(A_1)} \Pr(A_1 = x, A_2, \ldots, A_k|C)$. Using Bayes rule we can then compute the conditional probability $\Pr(C|A_2, \ldots, A_k)$ for classification. Similar considerations apply training with missing values as well, although these come at some computational cost; see Dempster, Laird & Rubin (1977) and Gelman, Carlin, Stern & Rubin (1995). We note that this approach assumes that the values are *missing at random*, that is, that the process by which these values were removed does not depend on the actual missing values, given the values we do observe (Rubin 1976). When this assumption is not reasonable, then we have to either include a model of this hiding process (i.e., the probability that the values are missing) or use other approaches (see below).

Finally, probabilistic methods allow for use of prior knowledge and for combining knowledge from other sources. The probabilistic semantics provides a clear way of using prior knowledge about the domain, and knowledge gathered from other sources (e.g., different training data) in the classification process. This knowledge can be used in various ways. For instance, prior knowledge my determine the type of model we use for estimating $\Pr(A_1, \ldots, A_k|C)$. In speech recognition, for example, the attributes are measurements of the speech signal, and the probabilistic model is a Hidden Markov Model (Rabiner 1990) that is usually composed from phoneme models. This highly structured model is motivated by our prior knowledge on speech. Note that the choice of model usually reflect our knowledge about the process that *generated* the observations. In contrast, choice of model class (e.g., decision trees vs. neural networks) in other classification methods usually depends on the type of decision surface we expect to learn and the amount of data we can learn with. Depending on the domain, either way of thinking of the choice of models can be more natural.

Prior knowledge can be also used in other ways. For example, it can be used to determine our prior estimate of probabilities. This leads to shifting our estimate toward specific values. If training data for a particular parameter of the model is sparse, then the final estimate is heavily dependent on the prior, and if there is sufficient training data, then the final estimate is usually not sensitive to the prior. Additionally, the probabilistic semantic, and the representation tools (such as probabilistic networks [link to section C5.6] (Pearl 1988)) allow to combine learning with modeling assumptions and knowledge about the domain. That is, we might fix in advance part of the model and learn the other parts.

C5.1.5.3 The Naive Bayesian Classifier

We now turn to the question of estimating $\Pr(A_1, \ldots, A_k|C)$. This is a *density estimation* problem, since we are attempting to learn the probability distribution of the attributes among all the instances with the same label. We first note that we cannot use counting to estimate this probability because most of the counts will be zero.

To see this, suppose that all the attributes are binary. Then there are 2^k possible assignments to the attributes, and even for a moderate number of attributes, we do not expect to see most of these assignments in the training data.

One way of addressing this problem, is to use the so called *Naive Bayesian classifier* (Duda & Hart 1973, Langley, Iba & Thompson 1992), sometimes called the *Simple Bayesian classifier* (Domingos & Pazzani 1997). We assume that each attribute is independent of the rest given the value of the class. We easily establish that, given this assumption, we can write

$$\Pr(A_1, \ldots, A_k|C) = \Pr(A_1|C) \cdot \Pr(A_2|C) \cdots \Pr(A_k \mid C) \tag{3}$$

Now the estimation problem is easier, since we need to estimate the probability of each attribute given the class independently of the rest. Combining (2) and (3), we get the Naive Bayesian classier classification rule:

$$\Pr(C|A_1, \ldots, A_k) = \alpha \Pr(C) \Pr(A_1|C) \cdots \Pr(A_k|C), \tag{4}$$

where, again, α is a normalization constant.

The probabilities above are estimated from the training set and the posterior probability for each class is computed. The prediction is made for the class with the largest posterior probability. The model works well in areas where the conditional independence assumption is likely to hold, such as medical domains (Kononenko 1993). In recent years, the model was found to be very robust and continues to perform well even in the face of obvious violations of this conditional independence assumption (Domingos & Pazzani 1997, Kohavi & Sommerfield 1995, Friedman 1997).

Estimating the probabilities can be done using simple frequency counts, but this creates problems if the counts of an attribute and a class is zero because assigning a probability of zero to one of the terms, $\Pr(A_i|C)$, causes the whole expression to evaluate to zero and rule out a class. This is especially problematic when attributes have many values and the distribution is sparse: several (or even all) classes get a probability of zero. Several methods have been proposed to overcome this issue. The zero probability can be replaced by a small constant, such as $0.5/n$ or $\Pr(C)/n$, where n is the number of instances in the training set (Clark & Niblett 1989, Kohavi, Sommerfield & Dougherty 1997). Another, more theoretically justified, approach is to apply a generalized Laplace correction (Cestnik 1990, Kohavi, Becker & Sommerfield 1997).

Unknown (missing, null) values are commonly handled in one of two ways. In evaluat-

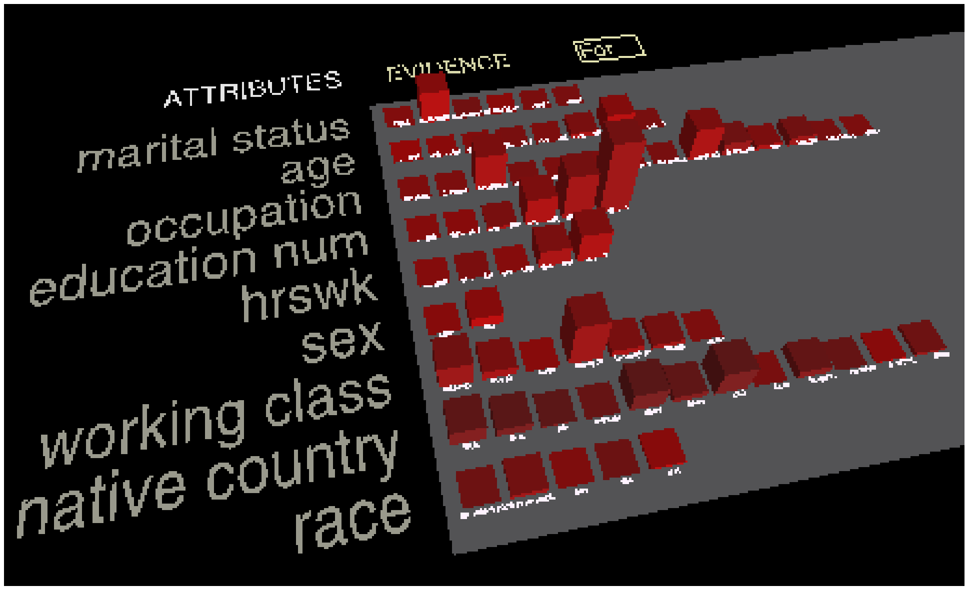
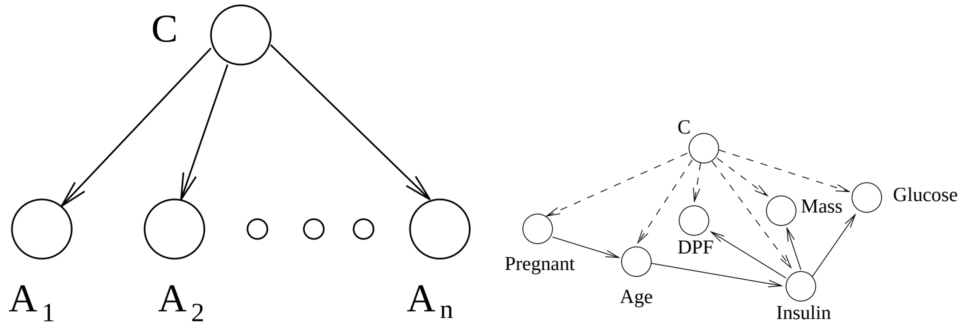


Figure C5.1.5.1: Visualization of Naive Bayes in MineSet™ [link to section D2.2.5], showing US census data for working adults. The attributes are sorted by their discrimination power. For each continuous attribute the range is discretized. For each value (or range), the bar height shows the *evidence* (log of the conditional probability). In this case, the label chosen in the GUI was gross income over $50,000. The high bars indicate that there is most evidence for people to earn over $50,000 when they satisfy one or more of the following criteria: they are married; their age is between 36 and 61; their occupation is executive managerial or professional specialist; their highly educated; they work over 40 hours a week, etc.

ing the probabilities $\Pr(A_i|C)$, when A_i is unknown, one can simply ignore the term, which is equivalent to marginalizing over the attribute, something done in $\mathcal{MLC}$++ [link to section D2.1.2] (Kohavi, Sommerfield & Dougherty 1997). Another alternative is to estimate the probabilities from unknown values in the data. The second alternative works better if there is a special meaning to a missing value (e.g., a blank entry for the army rank of a person usually indicates the person did not serve in the army).

An important advantage of Naive Bayes is that the simple structure lends itself to comprehensible visualizations (Becker, Kohavi & Sommerfield 1997, Kononenko 1993). Figure C5.1.5.1 shows an example visualization used in MineSet (Silicon Graphics 1998, Brunk, Kelly & Kohavi 1997).

As can be expected from the form of (4), the decision surfaces learned by the Naive Bayesian classifier are of limited form. In particular, if the attributes are binary, then it is easy to show that the decision between any two classes is made by a hyperplane. (Linear decision surface occur also when the attributes are nominal and the conditional distributions are Gaussians.) This fact has been known since the 60's,



(a) (b)

Figure C5.1.5.2: Description of two Bayesian classifiers for diabetes type classification using the probabilistic network representation: (a) the Naive Bayesian classifier, (b) a TAN model learned form data. The dashed lines are those edges required by the naive Bayesian classifier. The solid lines are the dependency edges between attributes that were learned Friedman et al.'s algorithm.

e.g., Duda & Hart (1973), and has been frequently rediscovered. Notice, however, that the decision rule learned by the Naive Bayesian classifier would not, in general, coincide with the ones learned by other linear methods, such as perceptrons.

C5.1.5.4 Alternative Approaches

There are several possible extensions of Bayesian classification beyond the Naive Bayesian classifier. These works fall into several categories.

Work in the first category, such as that of Langley & Sage (1994) and of Kohavi & John (1997), attempted to improve classification accuracy by restricting attention only to a subset of the attributes. This approach can reduce errors due to a strong correlation among attributes by removing one or more of the correlated attributes.

Work in the second category (Ezawa & Schuermann 1995, Friedman, Geiger & Goldszmidt 1997, Kononenko 1991, Pazzani 1995, Sahami 1996) attempts to improve the classification accuracy by removing some of the independence assumptions made in the Naive Bayesian classifier. It turns out that *probabilistic networks* [link to section C5.6] (also known as *Bayesian networks*) provides a useful language to describe such independencies. Friedman et al. (1997) discuss several ways of using these networks for Bayesian classification. Figure C5.1.5.2(a) shows how the Naive Bayesian classifier is represented as a probabilistic network.

For brevity, we will briefly describe one of these approaches that Friedman et al. call *Tree-Augmented Naive* Bayesian classifier, or TAN, is based on ideas that go back to Chow & Liu (1968). In this approach, instead of assuming that each attribute is independent of the rest, we allow each one to depend on at most one other attribute. An example of such a dependency structure, in a probabilistic network notation, is shown in Figure C5.1.5.2(b). The choice of these dependencies implies a different decomposition of the attributes' joint distribution. For example, the decomposition corresponding to the network shown in Figure C5.1.5.2(b) is

$$\Pr(P, A, I, D, M, G|C) = \Pr(P|C)\Pr(A|P, C)\Pr(I|A, C)\Pr(D|I, C)\Pr(M|I, C)\Pr(G|I, C),$$

where we use the obvious abbreviation for each attribute name. In this augmented dependency structure, an edge from A_i to A_j implies that the influence of A_i on the assessment of the class variable also depends on the value of A_j. For example, in Figure C5.1.5.2(b), the influence of the attribute "Glucose" on the class C depends on the value of "Insulin," while in the naive Bayesian classifier the influence of each attribute on the class variable is independent of other attributes. These edges affect the classification process in that a value of "Glucose" that is typically surprising (i.e., $\Pr(g|c)$ is low) may be unsurprising if the value of its correlated attribute, "Insulin," is also unlikely (i.e., $\Pr(g|c, i)$ is high). In this situation, the naive Bayesian classifier will overpenalize the probability of the class variable by considering two unlikely observations, while the augmented network of Figure C5.1.5.2(b) will not.

We are now faced with the question of how to choose the dependency arcs. Friedman et al. describe a procedure that finds the decomposition function that maximizes the *likelihood* [link to section B5] of the data. In addition, this procedure has attractive computational properties, its running time is linear in the number of training instances and quadratic in the number of attributes, k. The TAN method is a compromise between the complexity of the learned model and the generalization ability and computational cost of learning the model. Because only pairwise interactions are modeled directly, the learned model requires only estimates of pairs of attributes, which are relatively robust and efficient to compute. It is clear that in some domains other points on this tradeoffs might be explored. In general, for more complex models, it is NP-hard to find the maximal likelihood structure, and thus we need to resort to some heuristic search. See **?** for some work in these directions.

Finally, in the last category there are approaches that use domain specific models. For example, speech recognition (Rabiner 1990) and protein classification (Durbin, Eddy, Krogh & Mitchison 1998) use specialized Hidden Markov models to learn the distribution of the observed attributes (sound waves frequencies, and amino acids).

Approaches in these categories rely on knowledge of special structure in the domain to construct the density estimates.

References

Becker, B., Kohavi, R. & Sommerfield, D. (1997), Visualizing the simple bayesian classifier, *in* 'KDD Workshop on Issues in the Integration of Data Mining and Data Visualization'.

Bishop, C. M. (1995), *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, U.K.

Brunk, C., Kelly, J. & Kohavi, R. (1997), MineSet: an integrated system for data mining, *in* D. Heckerman, H. Mannila, D. Pregibon & R. Uthurusamy, eds, 'Proceedings of the third international conference on Knowledge Discovery and Data Mining', AAAI Press, pp. 135–138.
`http://mineset.sgi.com`.

Cestnik, B. (1990), Estimating probabilities: A crucial task in machine learning, *in* L. C. Aiello, ed., 'Proceedings of the ninth European Conference on Artificial Intelligence', pp. 147–149.

Chow, C. K. & Liu, C. N. (1968), 'Approximating discrete probability distributions with dependence trees', *IEEE Trans. on Info. Theory* **14**, 462–467.

Clark, P. & Niblett, T. (1989), 'The CN2 induction algorithm', *Machine Learning* **3**(4), 261–283.

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977), 'Maximum likelihood from incomplete data via the EM algorithm', *Journal of the Royal Statistical Society* **B 39**, 1–39.

Domingos, P. & Pazzani, M. (1997), 'Beyond independence: Conditions for the optimality of the simple Bayesian classifier', *Machine Learning* **29**(2/3), 103–130.

Duda, R. & Hart, P. (1973), *Pattern Classification and Scene Analysis*, Wiley.

Durbin, R., Eddy, S., Krogh, A. & Mitchison, G. (1998), *Biological Sequence Analysis : Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press.

Ezawa, K. J. & Schuermann, T. (1995), Fraud/uncollectable debt detection using a Bayesian network based learning system: A rare binary outcome with mixed data structures, *in* P. Besnard & S. Hanks, eds, 'Proc. Eleventh Conference on Uncertainty in Artificial Intelligence (UAI '95)', Morgan Kaufmann, San Francisco, pp. 157–166.

Friedman, J. H. (1997), 'On bias, variance, 0/1-loss, and the curse of dimensionality', *Data Mining and Knowledge Discovery* **1**(1), 55–77. `ftp://playfair.stanford.edu/pub/friedman/curse.ps.Z`.

Friedman, N., Geiger, D. & Goldszmidt, M. (1997), 'Bayesian network classifiers', *Machine Learning* **29**, 131–163.

Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, D. B. (1995), *Bayesian Data Analysis*, Chapman & Hall, London.

Kohavi, R., Becker, B. & Sommerfield, D. (1997), Improving simple bayes, *in* 'The 9th European Conference on Machine Learning, Poster Papers'.

Kohavi, R. & John, G. H. (1997), 'Wrappers for feature subset selection', *Artificial Intelligence* **97**(1-2), 273–324. `http://robotics.stanford.edu/users/ronnyk`.

Kohavi, R. & Sommerfield, D. (1995), Feature subset selection using the wrapper model: Overfitting and dynamic search space topology, *in* 'The First International Conference on Knowledge Discovery and Data Mining', pp. 192–197.

Kohavi, R., Sommerfield, D. & Dougherty, J. (1997), 'Data mining using $\mathcal{MLC}$++: A machine learning library in C++', *International Journal on Artificial Intelligence Tools* **6**(4), 537–566. `http://www.sgi.com/Technology/mlc`.

Kononenko, I. (1991), Semi-naive Bayesian classifier, *in* Y. Kodratoff, ed., 'Proc. Sixth European Working Session on Learning', Springer-Verlag, Berlin, pp. 206–219.

Kononenko, I. (1993), 'Inductive and Bayesian learning in medical diagnosis', *Applied Artificial Intelligence* **7**, 317–337.

Langley, P., Iba, W. & Thompson, K. (1992), An analysis of Bayesian classifiers, *in* 'Proceedings of the tenth national conference on artificial intelligence', AAAI Press and MIT Press, pp. 223–228.

Langley, P. & Sage, S. (1994), Induction of selective Bayesian classifiers, *in* 'Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence', Morgan Kaufmann, Seattle, WA, pp. 399–406.

Pazzani, M. J. (1995), Searching for dependencies in Bayesian classifiers, *in* D. Fisher & H. Lenz, eds, 'Proceedings of the fifth International Workshop on Artificial Intelligence and Statistics', Ft. Lauderdale, FL.

Pearl, J. (1988), *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, San Mateo, CA.

Rabiner, L. R. (1990), 'A tutorial on hidden Markov models and selected applications in speech recognition.', *Proceedings of the IEEE* .

Rubin, D. R. (1976), 'Inference and missing data', *Biometrica* **63**, 581–592.

Sahami, M. (1996), Learning limited dependence Bayesian classifiers, *in* 'KDD-96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining', AAAI Press, Menlo Park, CA, pp. 335–338.

Silicon Graphics (1998), *MineSet User's Guide*, Silicon Graphics, Inc. http://mineset.sgi.com.