Feature selection for class probability estimation

Gyozo Gidofalvi
gyozo@cs.ucsd.edu

Bianca Zadrozny
zadrozny@cs.ucsd.edu

Department of Computer Science and Engineering
University of California, San Diego
La Jolla, CA 92093-0114

1 Introduction

Accurate estimates of class membership probabilities are needed in many supervised learning applications when the classification outputs are not used in isolation but are combined with other sources of information for decision-making. For example, such probability estimates are needed when a cost-sensitive decision must be made about examples with example-dependent costs [Zadrozny and Elkan, 2001]. Another application is in speech recognition, where the probabilistic outputs from classifiers are used as input to a hidden Markov model that models sequences of phonemes.

Current feature selection methods choose features that are suitable for 0-1 classification. They do not take into consideration the accuracy of the class probability estimates given by the classifier, which is crucial for many applications. In this paper, we present methods for selecting features for naive Bayesian classifiers with the goal of obtaining accurate class probability estimates.

Naive Bayesian classifiers assign to each example x a score between 0 and 1 that can be interpreted, in principle, as a class membership probability estimate. However, it is well known that these estimates are not accurate when the naive Bayesian assumption of conditional independence of features (given the class label c) is violated. Mathematically, this assumption can be stated as

$$P(x|c) = \prod_{k=1}^{n} P(x_k|c)$$

where x_k is one feature value of x.

Given an example x, suppose that a naive Bayesian classifier computes the score $s(x)$. Because features tend to be correlated, these scores are typically too extreme: for most x, either $s(x)$ is near 0 and then $s(x) < P(c|x)$ or $s(x)$ is near 1 and then $s(x) > P(c|x)$.

Because naive Bayesian classifiers tend to rank examples well (if $s(x) < s(y)$ then $P(c|x) < P(c|y)$), the presence of correlated features does not hurt the 0-1 classification error as much as it hurts class probability estimation. By selecting a set of features that are not as correlated to each other while still being predictive of the class label, we can significantly improved the accuracy of the probability estimates.

Here, we propose two wrapper methods for selecting features in this setting and compare their performances by conducting experiments using the KDD-98 dataset [Bay, 2000].

Wrapper feature selection methods search for a subset of features using the learning method as a black box when evaluating the results [Kohavi and John, 1997].

In Section 2, we describe the KDD-98 dataset, the decision-making task associated with it and why class probability estimates are necessary for this task. In Sections 3, we explain the two proposed wrapper feature selection methods. In Section 4 we report experimental results using the KDD-98 dataset as a test set. Finally, in Section 5 we summarize the main contributions of this paper and suggest directions for future work.

2 A testbed: the KDD-98 dataset

The KDD-98 dataset is a well-studied, large and challenging dataset that was first used in the data mining contest associated with the 1998 KDD conference. This dataset and associated documentation are available in the UCI KDD repository [Bay, 2000]. The dataset contains information about persons who have made donations in the past to a certain charity. The decision-making task is to choose which donors to request a new donation from. This task is completely analogous to typical one-to-one marketing tasks for may organizations, both non-profit and for-profit.

The dataset contains 478 features for each person describing responses to previous donation campaigns and demographic information about the neighborhood where the person lives. The dataset is divided in a fixed, standard way into a training set and a test set. The training set consists of 95412 records for which it is known whether or not the person made a donation (a 0/1 response) and how much the person donated, if a donation was made. The test set consists of 96367 records from the same donation campaign for which similar donation information was not published until after the KDD-98 contest.

Mailing a solicitation to an individual costs the charity \$0.68. The overall percentage of donors among potential recipients is about 5%. The donation amount for persons who respond varies from \$1 to \$200. Given the low response rate and the variation in the value of gifts, it is not easy to achieve a profit that is much higher than that obtained by soliciting all potential donors. The profit obtained by soliciting every individual in the test set is \$10560, while the profit attained by the winner of the KDD'98 competition was \$14712.

The optimal predicted label for example x is the class i that maximizes

$$\sum_j P(j|x)B(i,j,x) \tag{1}$$

where $B(i,j,x)$ is the benefit of predicting class i when the true class is j.

Let the label $j=0$ mean the person x does not donate, and let $j=1$ mean the person does donate. If the person donates, the donation is of a variable amount, say $y(x)$. The cost of mailing a solicitation is \$0.68, so we have the following benefit matrix $B(i,j,x)$:

	actual non-donor	actual donor
predict non-donor	0	0
predict donor (mail)	-0.68	$y(x)-0.68$

Notice that $B(1,1,x)$ is example-dependent and unknown for test examples.

The expected benefit of not soliciting a person x, i.e. of deciding $i=0$ for x, is

$$P(j=0|x)B(0,0,x)+P(j=1|x)B(0,1,x)=0.$$

The expected benefit of soliciting x is

$$P(j=0|x)B(1,0,x)+P(j=1|x)B(1,1,x)$$

Given $(x_i, y_i), \ldots, (x_n, y_n)$; $x_i \in X, y_i \in \{0,1\}$
Let $F = \emptyset$
For $t = 1, \cdots, T$:

- Train one classifier for each feature j using all features in $F \bigcup \{j\}$.
- Let $s_j(i)$ be the probability estimates output by the classifier trained using $F \bigcup \{j\}$.
- Let $\text{MSE}_j = \frac{1}{n} \sum_i (y(i) - s_j(i))^2 + ((1 - y(i)) - (1 - s(i)))^2$
- Choose feature k such that $\forall j \neq k, \text{MSE}_k < \text{MSE}_j$.
- Let $F = F \bigcup \{k\}$.

Return F.

Table 1: Forward selection using class probability estimates

$$= (1 - P(j = 1|x))(-0.68) + P(j = 1|x)(y(x) - 0.68)$$
$$= P(j = 1|x)y(x) - 0.68.$$

The optimal policy is to solicit exactly those people for whom the expected benefit of mailing is greater than the expected benefit of not mailing: individuals for whom

$$P(j = 1|x)y(x) - 0.68 > 0.$$

In other words, the optimal policy is to mail to people for whom the expected return $P(j = 1|x)y(x)$ is greater than the cost of mailing a solicitation:

$$P(j = 1|x)y(x) > 0.68. \quad (2)$$

In order to apply this policy, we need to estimate the conditional probability of making a donation $P(j = 1|x)$ and the donation amount $y(x)$ for each example x. Since this paper is not concerned with donation amount estimation, we use fixed values for $y(x)$ obtained using a simple linear regression described in [Zadrozny and Elkan, 2001].

3 Feature selection methods

3.1 Forward selection

The first feature selection method we propose is a simple forward selection greedy method [Caruana and Freitag, 1994], in each we evaluate the accuracy of the probability estimates produced by the classifier using the mean squared error (MSE), also known as Brier score [Brier, 1950].

For one example x, the squared error (SE) is defined as $\sum_c (T(c|x) - P(c|x))^2$ where $P(c|x)$ is the probability estimated by the method for example x and class c and $T(c|x)$ is defined to be 1 if the actual label of x is c and 0 otherwise. We calculate the squared error for each example x in the training set to obtain the mean squared error (MSE).

A classifier is said to be calibrated well if the empirical class membership probability $P(c|s(x) = s)$ given that the example has a certain score value s converges to the score value $s(x) = s$, as the number of examples classified goes to infinity.[Murphy and Winkler, 1977]. Intuitively, if we consider all the examples to which a classifier assigns a score $s(x) = 0.8$, then 80% of these examples should be members of the class in question. Calibration is important if we want the scores to be directly interpretable as the chances of membership in the class.

DeGroot and Fienberg [DeGroot and Fienberg, 1982] show that the MSE can be separated into two components, one measuring calibration and the other measuring refinement. If the

Given $(x_i, y_i), \ldots, (x_n, y_n)$; $x_i \in X, y_i \in \{-1, +1\}$
Initialize weights $w_1(i) = 1/n$.
Let $F = \emptyset$
For $t = 1, \cdots, T$:

- Train one hypothesis $h_j : X \rightarrow \Re$ for each feature j using distribution w_i.
- Let $r_j = \sum_i w_t(i) y_i h_j(x_i)$
- Choose $h_t(\cdot) = h_k(\cdot)$ such that $\forall j \neq k, r_k > r_j$.
- Let $F = F \bigcup \{k\}$.
- Let $r_t = r_k$.
- Let $\alpha_t = \frac{1}{2} \ln \left(\frac{1+r_t}{1-r_t} \right)$
- Update:
$$w_{t+1}(i) = w_t(i) \exp(-\alpha_t y_i h_t(x_i))$$
- Normalize $w_{t+1}(i) = \frac{w_{t+1}(i)}{\sum_n w_{t+1}(i)}$ so that $w_{t+1}(i)$ is a distribution.

Return F.

Table 2: Boosted feature selection using confidence estimates

classifier is well-calibrated the first component is zero. For two classifiers that are well-calibrated, the one for which the probability estimates $P(c|x)$ are closer to 0 or 1 is said to be more refined, because it makes predictions that are more confident. Because the second component of the MSE measures refinement, if the two classifiers are well-calibrated, the one with the lowest MSE is more refined, and thus, preferable.

Our version of the forward selection (FS) method starts with the empty set of features and greedily adds features one at a time. At each step, FS adds the feature that, when added to the current set, yields the lowest MSE on the training set. Table 1 shows the pseudo-code for this method. Because naive Bayes does not overfit the training data, its performance on the training set reflects its performance on new examples, so we do not need to hold out data for evaluation.

3.2 Boosted selection

The second method is based on the boosted feature selection method proposed by Viola and Jones [Viola and Jones, 2001]. In their method, at each round one classifier is learned for each feature and the classifier with the lowest error rate is selected. The examples are then reweighted so that misclassified examples get more attention in the next round. The reweighting is performed according to the AdaBoost ensemble learning method [Freund and Schapire, 1996]. The features selected are the ones used by the winning classifiers.

The method originally proposed by Viola and Jones [Viola and Jones, 2001] cannot be applied to class probability estimation directly because it uses the 0-1 classification error as the evaluation measure and to update the weights of the examples at each round.

Schapire and Singer [Schapire and Singer, 1998] propose improvements to AdaBoost in a setting in which classifiers may assign confidences to each of their predictions. In particular they give a definition of a weak learner, an error criterion and weight update rule proposed for this setting. However, the confidence scores used by this methods are not class probability estimates. The confidence score output by the classifier should be a real number that is positive if the example is classified as positive, and negative if the example

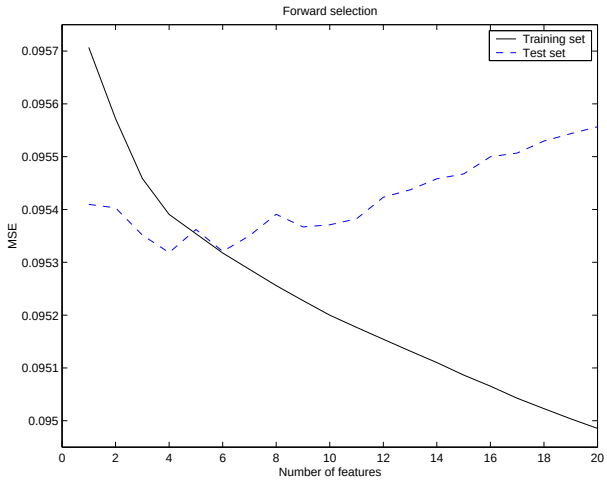

Figure 1: MSE using the forward selection method

is classified as negative. The magnitude of the score should be a measure of confidence in the prediction.

We use this version of AdaBoost for our boosted selection algorithm. In order to use this method, we first convert the probability estimates output by naive Bayes into confidence scores. If $s(x)$ is the probability estimate output by the classifier for example x, we let $h(x) = 2s(x) - 1$ be the confidence estimate for x. This way, we map the scores from the [0,1] interval to the [-1,1] interval, and $p(x) = 0.5$ corresponds to $h(x) = 0$.

Table 2 shows the pseudo-code for this method. The measure $r_j = \sum_i w_t(i) y_i h_j(x_i)$ is used to evaluate the accuracy of each hypothesis j taking into account the confidence estimates $h_j(x_i)$ and the weights $w_t(i)$. Intuitively, of examples that have larger weights are assigned confidence estimates of large magnitude with the correct sign, than r_t is large and the hypothesis is considered accurate.

4 Experimental results

We use each of the feature selection methods to select 20 features from the KDD-98 dataset.

Figure 1 shows how the MSE changes when we apply the forward selection as the number of features selected increases. Notice that although the MSE on the training set steadily decreases as we add more features, the MSE on the test set reaches a minimum for 4 features. Although we do not need a separate hold out set to evaluate each run of naive Bayes, we would need a hold out set to pick the optimal number of features that generalizes well to new data as we add more features.

Figure 2 shows how the MSE changes when we apply the boosted selection method as the number of features selected increases. Surprisingly, the MSE on the training set increases steadily while the MSE on the test set decrease slightly until iteration 5 and then starts increasing. An explanation for the poor performance of this method is that although we measure the accuracy of the probability estimates using the MSE, the boosted feature selection method does not directly optimize MSE, but a related measure that does not take calibration into account.

In Table 3 we show a summary of the results. Besides showing the results for the two

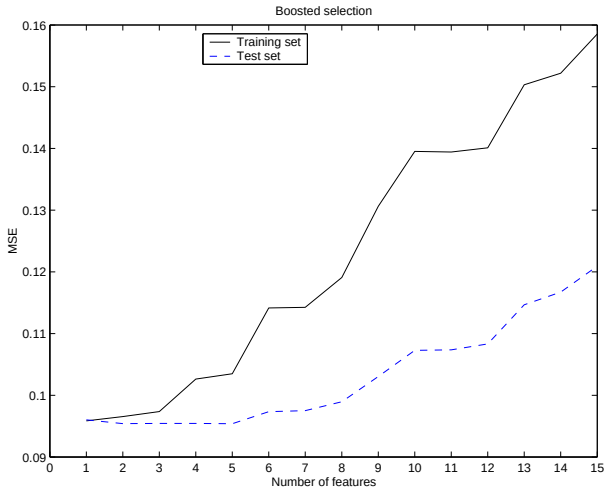

Figure 2: MSE using the boosted selection method

methods presented here, we also show the results obtained using a set of 7 hand-selected features from previous experiments with this dataset [Zadrozny and Elkan, 2001], and the results of using all the 478 features. We use two evaluation measures for comparison: the MSE and the profit obtained when we mail the individuals according to the policy given by Equation 2.

Note that using all features yields a very high MSE, indicating that the class probability estimates produced by naive Bayes are poorly calibrated. By using forward selection we can find a set of 10 features that yield low MSE and high profit on the test set. Such results are comparable to the winner of the KDD-98 competition. Notice that there is a correlation between MSE and profit, but it is not perfect. This is the case because we do not have perfect estimates of the donation amounts.

5 Conclusions

Naive Bayes produces poor class membership probability estimates given many correlated features, and for this reason it is possible to significantly improve the accuracy of those estimates by selecting appropriate features that are not very correlated. We show here that by greedily searching for a set of features that yields low MSE using a simple forward selection method, we can obtain accurate probability estimates for the KDD-98 dataset using naive Bayes. These estimates can be used to obtain a profit comparable to the winner of the KDD-98 competition.

We based our boosted feature selection method on a version of AdaBoost that uses confidence estimates. However these confidence estimates are not calibrated class probability estimates and the evaluation measure used by the method does not take calibration into account. This is one explanation for the failure of this method on the KDD-98 dataset. In future research, it would be worth designing a boosted feature selection method taking into account the calibration of the estimates.

MSE

Method	Training Set	Test Set
7 hand-selected	0.10089	0.10111
5 forward	0.09535	0.09536
10 forward	0.09520	0.09537
20 forward	0.09499	0.09556
5 boosted	0.10349	0.09540
all features	0.64965	0.65750

Profit

Method	Training Set	Test Set
7 hand-selected	$10083	$9531
5 forward	$15484	$14048
10 forward	$16221	$14443
20 forward	$16877	$13366
5 boosted	$7365	$11700
all features	$14355	$10319

Table 3: MSE and profit on the training and test sets

References

[Bay, 2000] Bay, S. D. (2000). UCI KDD archive. Department of Information and Computer Sciences, University of California, Irvine. `http://kdd.ics.uci.edu/`.

[Brier, 1950] Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78:1–3.

[Caruana and Freitag, 1994] Caruana, R. and Freitag, D. (1994). Greedy attribute selection. In *International Conference on Machine Learning*, pages 28–36.

[DeGroot and Fienberg, 1982] DeGroot, M. H. and Fienberg, S. E. (1982). The comparison and evaluation of forecasters. *Statistician*, 32(1):12–22.

[Freund and Schapire, 1996] Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *International Conference on Machine Learning*, pages 148–156.

[Kohavi and John, 1997] Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324.

[Murphy and Winkler, 1977] Murphy, A. and Winkler, R. (1977). Reliability of subjective probability forecasts of precipitation and temperature. *Applied Statistics*, 26(1):41–47.

[Schapire and Singer, 1998] Schapire, R. E. and Singer, Y. (1998). Improved boosting algorithms using confidence-rated predictions. In *Computational Learning Theory*, pages 80–91.

[Viola and Jones, 2001] Viola, P. and Jones, M. (2001). Robust real-time object detection. In *International Workshop on Statistical and Computational Theories of Vision*.

[Zadrozny and Elkan, 2001] Zadrozny, B. and Elkan, C. (2001). Learning and making decisions when costs and probabilities are both unknown. In *Proceedings of the Seventh International Conference on Knowledge Discovery and Data Mining*, pages 204–213. ACM Press.