Active Sampling for Feature Selection

Sriharsha Veeramachaneni and Paolo Avesani

ITC-IRST,
Via Sommarive 18 - Loc. Pantè, I-38050 Povo, Trento, Italy
{*sriharsha,avesani*}*@irst.itc.it*

Abstract. In many knowledge discovery applications the data mining step is followed by further data acquisition. New data may consist of new instances and/or new features for the old instances. When new features are to be added an acquisition policy can help decide what features have to be acquired based on their predictive capability and the cost of acquisition. This can be posed as a feature selection problem where the feature values are not known in advance. We propose a technique to actively sample the feature values with the ultimate goal of choosing between alternative candidate features with minimum sampling cost. Our algorithm is based on extracting candidate features in a "region" of the instance space where the feature value is likely to alter our knowledge the most. An experimental evaluation on a standard database shows that it is possible outperform a random subsampling policy in terms of the accuracy in feature selection.

1 Introduction

Data mining is mainly concerned with data analysis and the related step of preprocessing. Usually it is assumed that the data are given in advance and their quality and size are parameters beyond the learner's control. The goal of the mining process is the development of an accurate predictive model that will aid in future decision making.

Sometimes the amount and the quality of data are insufficient to perform accurate induction. In these cases the data mining task fails. Taking the perspective of knowledge discovery, the mining process can be conceived not as a linear sequence of steps but as a never ending loop [15, 7, 11]. After an analysis step that produces a rough model a further step of data collection can be arranged to obtain a more accurate model.

Viewing the mining process as an iteration of data collection and data analysis steps, the objective at each step becomes twofold: a descriptive/predictive model and an acquisition policy. A new decision support system has to be developed whose objective is the planning of the data acquisition campaign. The two concerns of a data acquisition plan are

1) what instances to focus on and 2) what features have to be taken into account.

This paper aims to deal with the acquisition policy, restricted to feature extraction (or measurement). This work is motivated by a research project (SMAP) in the domain of agriculture dealing with the Apple Proliferation disease in apple trees[1]. The scenario [3, 9, 8] is the following: biologists monitor a distributed collection of apple trees affected by the disease; the goal is to characterize the conditions for infection spreading. An archive is arranged with a finite set of records each describing a single apple tree. The monitored set contains both infected and not infected trees. All the instances are labeled with respect to this boolean classification. Each summer the archive is updated extending each record with new features. Every year the biologists start by proposing new candidate features that could be extracted (or measured) and at the end of summer a new process of analysis is performed taking into account the past and new data. Since the data collection on the field can be very expensive or time consuming, at the beginning of summer the biologists have to arrange a data acquisition plan by selecting a subset of the candidate features that should be really acquired.

Clearly the usual approach in data mining, that regards feature selection as an a posteriori task performed on a database with a large number of features that are fully extracted on the set of instances, is inappropriate in our case. For our problem the feature selection has to be performed in advance.

We propose a look-ahead strategy for feature selection. Given a sample of labeled instances $S = \{s_i\}_{i=1,\ldots,N}$, described with respect to a set of features $\mathbf{X} = \{\mathbf{X}_j\}_{j=1,\ldots,M}$, the problem is to choose between two alternative candidate features $\mathbf{Y}$ and $\mathbf{Z}$ (In general, there are several candidate features to be ranked in order of relevance). The basic idea is to prescribe a policy that iteratively probes the value of the candidate features on instances $s \in S$. The challenge is to minimize the cost of feature extraction. If all the candidate features have unit cost, this is equivalent to minimizing the sum of sizes of the subsamples $\overline{S}_Y \subseteq S$ and $\overline{S}_Z \subseteq S$ on which the features $\mathbf{Y}$ and $\mathbf{Z}$ are respectively extracted. At the same time the policy should enable an accurate choice of the most

[1] This work is funded by Fondo Progetti PAT, SMAP (Scopazzi del Melo - Apple Proliferation), art. 9, Legge Provinciale 3/2000, DGP n. 1060 dd. 04/05/01.

relevant feature. Notice that when we conduct an exhaustive sampling, i.e., $\overline{S}_Y = \overline{S}_Z = S$, both $\mathbf{Y}$ and $\mathbf{Z}$ are measured on all the instances, we fall into the traditional framework of feature selection [1, 6].

The goal of this work is to find a trade-off between the assessment of the feature relevance and the cost incurred in the acquisition of the feature values. The optimum solution should select the same features that a fully informed method would select, but with a small subsample of the feature values.

It is supposed that after the initial data acquisition to determine feature relevance a full acquisition campaign is performed only for the most promising features, while the remaining features are discarded. Therefore the total cost incurred is the sum of the cost of the acquisition driven by the active feature extraction policy and the cost of full acquisition on the chosen features. For simplicity we assume a simple cost model where all the features have equal and unit cost.

After a brief overview of the related literature we sketch an acquisition policy based on the notion of entropy. An empirical evaluation on a standard dataset shows that it is possible to outperform the trivial random subsampling policy.

2 Related Work

Feature selection is a well studied problem in machine learning [1, 6] and in some respects the problem of active acquisition can be considered as a feature selection problem. Nevertheless the common premise of traditional feature selection techniques is the availability of a large amount of known features whose values are available for the entire set of instances. The assessment of a feature relevance is usually performed considering all the values of the given instances.

A recent work [4] proposes a feature selection method based on selective sampling. The idea is to reduce the computational cost of the feature selection by reducing the number of sampled data points. Random sampling is replaced by selective sampling that exploits the data distribution to detect the most informative examples. The detection of closely related examples is performed using a *kd*-tree indexing avoiding the necessity of comparing an example against all others. Although this approach saves computational effort during the relevance assessment, to build the *kd*-tree all the feature values for all the examples need to be known in advance.

Usually the output of a feature selection algorithm is a set of relevant features while a ranking may be more suitable for decision making support. A recent work that takes this perspective proposes a statistical approach that produces a rank built on a probe: the decision of keeping or discarding a given feature is based on the probability that this feature is ranked higher or lower than the probe [5]. Again, even though this approach reduces computational complexity because it does not require the assessment of all the possible rankings, the single step to check the relative order between two features requires the full knowledge of their values.

Our work on the active sampling for feature selection is inspired by earlier work on active learning [2, 14, 10]. Active learning is a framework where the learner has the freedom to select which data points are added to its training set. The acquisition of new examples is driven by the knowledge gained by the past acquisition steps with the aim of accurately learning the concept using as few labeled examples as possible.

We propose an analogous method that selects the next example with the aim of optimizing a target criterion (reduced error rate on feature ranking). In contrast to the conventional active learning paradigm, where the class labels of unlabeled samples are probed to quickly ascertain the best predictor of the class from the features, our problem is to choose from a set of class-labeled samples on which candidate features are to be extracted while still trying to obtain the best predictor for the class.

Although the influence of the cost model on a learning process has been studied (see the works on cost-sensitive learning [12, 13]), we currently ignore this aspect for our active sampling approach.

3 Active Sampling of Feature Values

Consider a set of monitored pattern instances (or subjects) $S = \{s_i\}_{i=1,\ldots,N}$. Let the random variable corresponding to the class label be denoted by $\mathbf{C}$ taking values in $\mathcal{C}$. We assume that the class labels $\{c_i\}_{i=1,\ldots,N}$ for all the instances are known. $\mathbf{X}, \mathbf{Y}_1, \ldots, \mathbf{Y}_M$ are discrete valued features that can be extracted on any pattern taking on values in $\mathcal{X}, \mathcal{Y}_1, \ldots \mathcal{Y}_M$ and respectively. Assume that feature $\mathbf{X}$ is extracted for all the subjects and therefore the feature values $\{x_i\}_{i=1,\ldots,N}$ are known. Therefore the estimated probability distribution $\hat{p}_N(c, x)$ on $\mathcal{C} \times \mathcal{X}$ is assumed be accurate

(the subscript N represents the number of samples used in for estimation). Initially none of the instances have features $\mathbf{Y}_1, \ldots, \mathbf{Y}_M$ extracted. The problem is to rank the candidate features $\mathbf{Y}_1, \ldots, \mathbf{Y}_M$ according to relevance for classification of the subjects given the feature $\mathbf{X}$ minimizing the cost incurred for feature extraction. Although for the following discussion we assume $\mathbf{X}$ is scalar valued, in general it can be a vector valued feature.

Our proposed method assumes that all features are nominal valued and the probability densities in question are multinomial. The feature extraction policy considers each candidate feature separately. Let $\mathbf{Y}$ denote the candidate feature whose relevance we are trying to learn. Whereas a random feature extraction scheme (denoted π_R) chooses an instance randomly from all the instances on which $\mathbf{Y}$ is not extracted, our active sampling strategy (denoted π_A) decides on the most 'profitable' subset (or region) $\acute{S}$ of instances for feature extraction. Then the candidate feature is extracted on a randomly chosen instance from $\acute{S}$.

Let $\overline{S}_k = \{(c_{i_1}, x_{i_1}, y_{i_1}), \ldots, (c_{i_{k-1}}, x_{i_{k-1}}, y_{i_{k-1}})\}$ be the current set of samples with $\mathbf{Y}$ extracted (i.e., with complete descriptions). Then

$$\pi_A(\overline{S}_k, \hat{p}(c, x)) = (c^\star, x^\star) \in \mathcal{C} \times \mathcal{X}$$

is the result of the active policy. The active sampling scheme is an iterative process that proposes at each step the class label and the value taken by the previous feature of the samples on which it is most beneficial to extract the candidate feature.

We define the set of all pattern instances s_i with $(c_i, x_i) = (\alpha, \beta)$ as region $R_{\alpha\beta}$. At every iteration, the active sampling algorithm chooses the subset $\acute{S}$ for the candidate feature extraction as

$$\acute{S} = \{s_i | s_i \in R_{c^\star x^\star}, \mathbf{Y} \text{ not already extracted on } s_i\}$$

The next region for feature extraction is chosen as follows. The intuition behind the algorithm is that the most informative region (information expressed in terms of entropy) at a given stage is where a sample most alters our current knowledge.

At iteration k of the active sampling algorithm we have an estimate (based on $\overline{S}_k$) of the probability distribution $\hat{p}_k(c, x, y) = \hat{p}_k(y|c, x)\hat{p}_N(c, x)$ on $\mathcal{C} \times \mathcal{X} \times \mathcal{Y}$, where θ is the estimator of the conditional probability distribution (i.e., $\hat{p}_k(y|c, x) = \theta(\overline{S}_k; c, x, y)$). The subscript k for the probability estimates makes it explicit that the estimates are based on $\overline{S}_k$.

Given the estimate for the conditional densities we can estimate the entropy in the class given both features which is given by

$$H_k(\mathbf{C}|\mathbf{X},\mathbf{Y}) = -\sum_{\mathcal{C},\mathcal{X},\mathcal{Y}} \hat{p}_k(c,x,y)\log\hat{p}_k(c|x,y) \tag{1}$$

Now for every $(\alpha,\beta) \in \mathcal{C}\times\mathcal{X}$ we compute $\hat{H}_{k+1}(\mathbf{C}|\mathbf{X},\mathbf{Y})$, where

$$\hat{H}_{k+1}(\mathbf{C}|\mathbf{X},\mathbf{Y}) = -\sum_{\gamma\in\mathcal{Y}} \hat{p}_k(\mathbf{Y}=\gamma|c,x)\sum_{\mathcal{C},\mathcal{X},\mathcal{Y}} q_\gamma(c,x,y)\log q_\gamma(c|x,y) \tag{2}$$

where $q_\gamma(c,x,y) = \hat{p}_N(c,x)\theta(\overline{S}_k \cup (\alpha,\beta,\gamma); c,x,y)$. That is q_γ is the estimated probability distribution if we augment the current data with the sample (α,β,γ).

Now the expected benefit of sampling in $R_{\alpha\beta}$ is given by the benefit function B defined as

$$B_k(\alpha,\beta) = |\hat{H}_{k+1}(\mathbf{C}|\mathbf{X},\mathbf{Y}) - H_k(\mathbf{C}|\mathbf{X},\mathbf{Y})| \tag{3}$$

After the benefit function is evaluated for every $(\alpha,\beta)\in\mathcal{C}\times\mathcal{X}$, the region with maximum expected benefit is chosen for extracting the next feature. That is,

$$(c^\star,x^\star) = \underset{(c,x)\in\mathcal{C}\times\mathcal{X}}{\operatorname{argmax}} B_k(c,x).$$

Thus the active decision is based upon the absolute change between the current estimate of the entropy in the class given the previous feature and the candidate feature and the expected entropy in the class after the candidate feature is extracted from a sample in the said region. The expectation is over the possible values that the candidate feature can assume and the probabilities used in calculation of the expectation are the current estimates. The algorithm iterates until the candidate feature is extracted on a specified number of instances denoted L ($L = |\overline{S}_Y|$, where $\overline{S}_Y$ is the final subsample). L is problem dependent and is determined by the cost constraints (for feature extraction).

In practice the monitored set is finite sized warranting the search for the most beneficial region with at least one sample on which the candidate feature is not extracted. We use a Bayes estimate for the distribution $p(y|c,x)$ under a Dirichlet prior (independent for each $(c,x)\in\mathcal{C}\times\mathcal{X}$) with all parameters set to unity. Due to the independence in the priors

the estimator θ is decoupled for each (c, x) implying that $q_\gamma(y|c, x) = \hat{p}_k(y|c, x)$ for all $(c, x) \neq (\alpha, \beta)$ where $R_{\alpha\beta}$ is the region whose benefit function is being computed. This fact can be used to show that the following benefit function is equivalent to the one in Equation 3.

$$\begin{aligned}\acute{B}_k(\alpha, \beta) = \quad & |\hat{H}_{k+1}(\mathbf{Y}|\mathbf{C} = \alpha, \mathbf{X} = \beta) - H_k(\mathbf{Y}|\mathbf{C} = \alpha, \mathbf{X} = \beta) \\ & - \hat{H}_{k+1}(\mathbf{Y}|\mathbf{X} = \beta) + H_k(\mathbf{Y}|\mathbf{X} = \beta)|\ p(\mathbf{C} = \alpha, \mathbf{X} = \beta)\end{aligned}$$

This allows for a more efficient implementation of the active learning algorithm. As mentioned earlier our active learning algorithm considers each candidate feature separately. Therefore the candidate features $\mathbf{Y}$ and $\mathbf{Z}$ are not necessarily extracted on the same subsample of the instances.

After the candidate feature is extracted on the specified number (L) of samples or after the cost budget is exhausted, we can construct a Bayes maximum a posteriori classifier using the final estimate $\hat{p}_L(c, x, y) = \hat{p}_L(y|c, x)\hat{p}_N(c, x)$ of the joint distribution. The features are ranked based upon the error rates of these classifiers.

4 Experimental evaluation

To compare the active feature sampling strategy and the random feature extraction scheme for feature evaluation we use two performance measures.

The first is based on the mean square error between the estimated error rate at a particular sample size and the "true" error rate for a given feature. For us the true error rate represents the estimated error rate of the classifier trained after extracting the candidate feature on all N samples in the training set (denoted $p_e^{(N)}$), i,e, the error rate of the classifier designed using $\hat{p}_N(c, x, y)$ (the estimate of the probability densities from all N samples).

For each of the schemes, a classifier $\phi_k : \mathcal{X} \times \mathcal{Y} \rightarrow \mathcal{C}$ is constructed after extracting the candidate feature $\mathbf{Y}$ on a given number (L) of samples (i.e, based upon $\hat{p}_L(c, x, y)$). Now the error rate of the classifier is evaluated as

$$p_e^{(L)} = \{1 - \sum_{x,y} \hat{p}_N(x, y|\mathbf{C} = \phi_k(x, y))\} * 100$$

Therefore at every iteration of the learning scheme we can estimate the error rate of the resulting classifier to obtain a ranking. However, in reality we cannot extract the candidate feature for all the samples to estimate the error rates. We can circumvent this problem by obtaining a small random subsample for testing.

The quantity $mse = \mathrm{E}\{(p_e^{(N)} - p_e^{(L)})^2\}$ is a measure of the correctness in the estimated error rate for a feature after the given number of samples were extracted. We compute the error rate of the classifier for several runs of the learning scheme for the given sample size to compute the mean square error. For a particular learning scheme and for each feature the mean square error can be plotted against the sample size.

The second performance measure which is based on the Spearman rank-order correlation more directly indicates the efficacy of a sampling scheme for feature ranking and therefore for feature selection. The Spearman rank-order correlation coefficient r between two vectors of scores for M variables is given by

$$r = 1 - \frac{6 \sum d^2}{M(M^2 - 1)}$$

where d is the difference in the ranks of corresponding variables.

For each sample size we compute the error rate (as described above) for every candidate feature which are then ranked accordingly. The rank-order correlation between this ranking and the "true" ranking (based on $p_e^{(N)}$ for all candidates) is computed and its average value over many iterations of the learning scheme is plotted against the sample size.

We chose the "mushroom" database from the UCI machine learning repository for experimentation because it contains a large number of instances with several nominal features whose relevance to the class varies widely. The 8124 instances are almost evenly distributed between 2 classes ("edible" and "poisonous") and there are 22 features extracted of which feature 11 ("stalk-root") has several missing values and was therefore deleted from the database leaving 21 features for each instance. Feature 5 ("odor") is the most relevant and feature[2] 15 ("veil-type") is the least relevant for classification. We only use $N = 4000$ randomly chosen samples for experimentation. We vary the number of samples extracted L from 0 through 100 for the plots.

[2] Feature 16 before deleting the "stalk-root" feature

To compare the performance of the active and the random learning schemes given previous features we partitioned the 21 features into seven sets of three. This was done instead of experimenting with all possible $\binom{21}{3}$ combinations for simplicity. Each of these sets was fixed as the previous feature set $\mathbf{X}$ and the remaining $M = 18$ features were evaluated as candidates. The error rates for the classifier trained on each of previous feature set (i.e, before the candidate feature is extracted on any sample) is given in Table 1.

Table 1. Error rates in % when the classifier is trained on each set of previous features $\mathbf{X}$.

$\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3)$	(1, 2, 3)	(4, 5, 6)	(7, 8, 9)	(10, 11, 12)	(13, 14, 15)	(16, 17, 18)	(19, 20, 21)
Error rate	30.9	2.3	12.8	17.5	24.7	21.3	11.1

5 Discussion of results

Figure 1 shows the behaviour of the rank-order correlation coefficient (y-axis) between the vector of error rates estimated after a specific number of samples (x-axis) are extracted and the true error rates. As mentioned earlier, true error rate represents the estimated error rate of the classifier trained on all the samples in the database. The different plots correspond to different sets of previous feature vectors $\mathbf{X}$ indicated on the top of each subplot. Our active feature extraction strategy converges more quickly to the correct ranking of the features than the random scheme. When $\mathbf{X} = (4, 5, 6)$ the difference is not significant. This can be attributed to the fact that feature 5 individually leads to a very low error rate and therefore the candidate features have to be extracted on a large number of samples to be confidently ranked.

For each set of previous features we plotted the mean square difference between the true error rate and the estimated error rate (y-axis) after the candidate feature is extracted on a specific number of samples (x-axis). Figure 2 shows the plot for feature where the active policy proffered the most advantage over the random scheme. The plots indicate that the active policy can be used to lower the cost for feature evaluation.

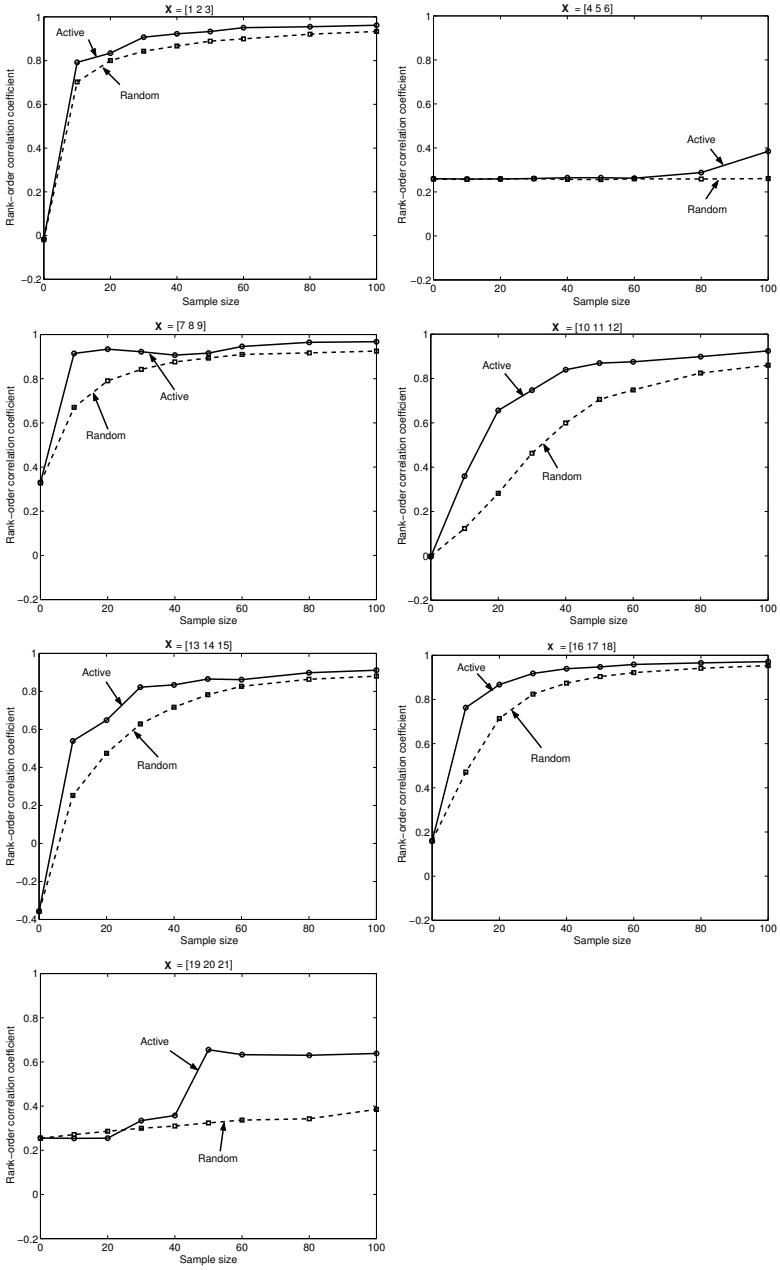

Fig. 1. The plot against sample size of the Spearman rank-order correlation between estimated error rates and the true error rates. For each sample size the rank-order correlation coefficient is averaged over 500 runs of the experiment. For each set of previous features only the feature for which the active policy was most beneficial over the random scheme is shown.

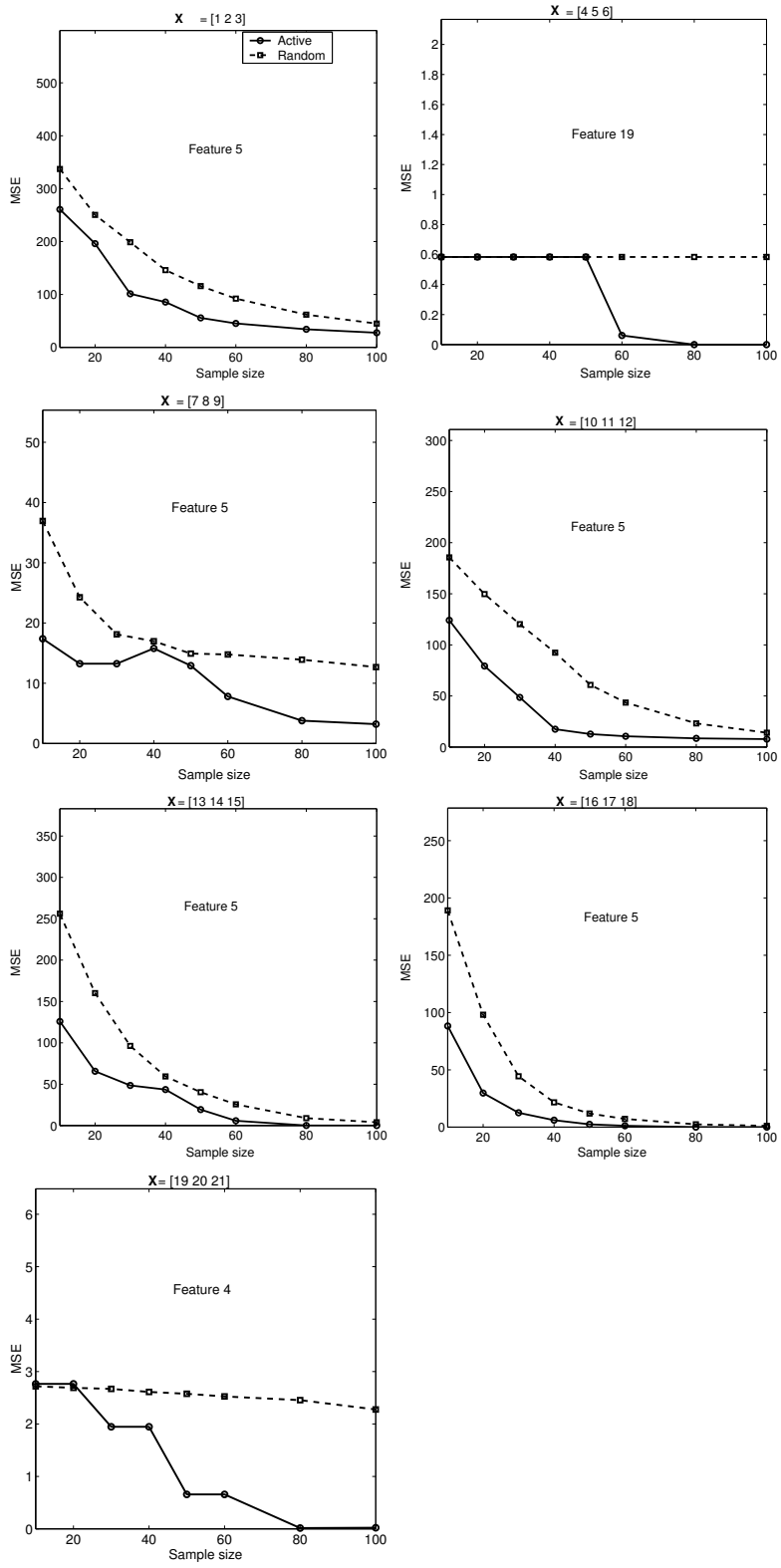

Fig. 2. The plot of the mean square difference (computed over 500 runs of the experiment) between estimated error rate and the true error rate. For each set of previous features $\mathbf{X}$ only the feature for which the active policy is most advantageous (based on the area between the curves) over the random policy is shown.

6 Conclusions and Future Work

In this paper we have dealt with the problem of cost-constrained feature selection in a knowledge discovery process. An active strategy that selects the instances for feature extraction is proposed to aid in choosing the most relevant features among a set of candidate features. The choice is based upon the absolute change between the current estimate of the entropy in the class before and the predicted entropy after the candidate feature value acquisition. A ranking is produced over the candidate features based on the estimate of the error rate of a classifier trained on a subsample of the feature values.

We provided empirical evidence on a standard dataset for the dominance of the active sampling scheme over a random policy for subsample selection for feature evaluation. A deeper analysis should be performed to derive an active policy with a non-trivial feature acquisition cost model.

The main purpose of this work was to investigate the possibility of exploiting the active learning approach for feature sampling. The promising results, although restricted to a specific case study, encourage more study taking scalability factors into account. Our current method does not scale to large number of previous features because of the necessity to estimate full class-conditional distributions (we do not assume any feature independence). Moreover the current design of solution is strongly related to the specific classifier, i.e. the Bayes classifier with class-conditional multinomial feature distributions. A general solution should be independent of the specific classifier and suitable for both nominal and real valued features.

References

1. A.Blum and P.Langley. Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence*, 97(1-2):245–271, 1997.
2. D.A.Cohn, L.Atlas, and R.E.Ladner. Improving Generalization with Active Learning. *Machine Learning*, 15(2):201–221, 1994.
3. G.Hughes. Sampling for Decision Making in Crop Loss Assessment and Pest Management: Introduction. In *Symposium on Sampling for Decision Making in Crop Loss Assessment and Pest Management*, pages 1080–1083, 1999.
4. H.Liu, H.Motoda, and L.Yu. Feature Selection with Selective Sampling. In *International Joint Conference on Machine Learning*, pages 395–402, 2002.
5. H.Stoppiglia, G.Dreyfus, R.Dubois, and Y.Oussar. Ranking a Random Feature for Variable and Feature Selection. *Journal of Machine Learning Research*, 3:1399–1414, 2003.

6. I.Guyon and A.Elisseefi. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.
7. E.Frank I.H.Witten. *Data Mining*. Morgan Kaufmann Publishers, 1999.
8. J.P.Nyrop, M.R.Binns, and W.van der Werf. Sampling for IPM Decision Making: Where Should We Invest Time and Resources. In *Symposium on Sampling for Decision Making in Crop Loss Assessment and Pest Management*, pages 1104–1111, 1999.
9. L.V.Madden and G.Hughes. Sampling for Plant Disease Incidence. In *Symposium on Sampling for Decision Making in Crop Loss Assessment and Pest Management*, pages 1088–1103, 1999.
10. M.Saar-Tsechansky and F.Provost. Active Sampling for Class Probability Estimation and Ranking. In *Proc. 7th International Joint Conference on Artificial Intelligence*, pages 911–920, 2001.
11. A.Susi P.Avesani, E.Olivetti. Feeding data mining. Technical report, ITC-irst, 2002.
12. P.Domingos. MetaCost: A General Method for Making Classifiers Cost-Sensitive. In *Knowledge Discovery and Data Mining*, pages 155–164, 1999.
13. P.D.Turney. Cost-sensitive Classification: Empirical Evaluation of a Hybrid Genetic Decision Tree Induction Algorithm. *Journal of Artificial Intelligence Research*, 2:369–409, 1995.
14. Nicholas Roy and Andrew McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proc. 18th International Conf. on Machine Learning*, pages 441–448. Morgan Kaufmann, San Francisco, CA, 2001.
15. J.Zyt W.Klosgen, J.M.Zytkow, editor. *Handbook of Data Mining and Knowledge Discovery*. Oxford University Press, 2002.